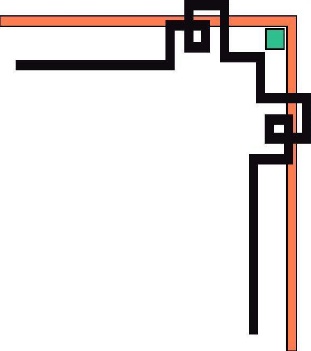
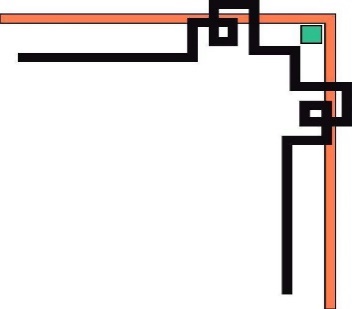
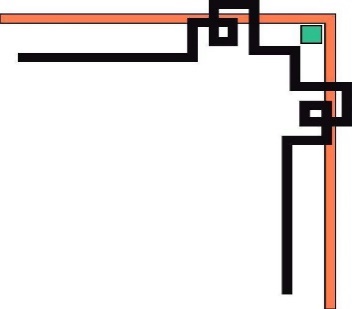
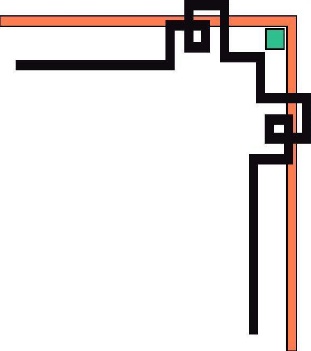
**BỘ GIÁO DỤC VÀ ĐÀO TẠO**



**TRƯỜNG ĐẠI HỌC CẦN THƠ**

**KHOA CÔNG NGHỆ THÔNG TIN & TRUYỀN THÔNG**

**BÁO CÁO NIÊN LUẬN NGÀNH HỆ THỐNG THÔNG TIN**

**ĐỀ TÀI: HỆ THỐNG QUẢN LÝ NHÀ HÀNG**

****

**GIÁO VIÊN HƯỚNG DẪN: TS *Phạm Thị Ngọc Diễm***

**SINH VIÊN THỰC HIỆN: NGUYỄN MINH Ý**

**MSSV: B1411457**

**CHUYÊN NGÀNH: Hệ Thống Thông Tin**

**Khóa 40 - MÃ LỚP: 1495A2**

**Cần Thơ, 05/ 2018**

# MỤC LỤC

# DANH MỤC HÌNH

# DANH MỤC BẢNG

# CHƯƠNG 1: TỔNG QUAN

## ĐẶT VẤN ĐỀ:

Hiện nay phần lớn các nhà hàng giúp khách hàng đặt tiệc một cách thuận lợi. Trong thời đại mới, chỉ cần có internet là khách hàng có thể tìm cho mình một chỗ ăn uống tuyệt vời cùng bạn bè và người thân.

Dù luôn tạo môi trường tốt nhất cho khách hàng nhưng vẫn gặp rất nhiều khó khăn cần giải quyết. Khó khăn đầu tiên là tốn kém cho phí, hiện tại muốn quản lý tốt, phục vụ khách hàng tận tình thì số lượng nhân viên phải nhiều, ước tính hơn cả trăm người cho một nhà hàng. Khó khăn thứ hai, gây lãng phí rất nhiều thời gian và công sức. Đầu tiên khách hàng đặt tiệc, nhân viên phải tiêm kiếm thời gian trống để giúp khách hàng đặt cho phù hợp. Khó khăn thứ ba, sổ sách ghi chú danh thu dễ bị thất lạc, rất dễ hỏng hóc, rách nát gây khó khăn trong công việc bảo quản, tìm kiếm, thống kê.

Từ những khó khăn trên, em quyết định đề tài “quản lý nhà hàng” để làm đề tài niên luận ngành . Hệ thống sẽ xây dựng một quy trình quản lý giúp tiết kiệm công sức, chi phí và thời gian, đơn giản trong khâu lưu trữ và tìm kiếm.

Hệ thống được phát triển bằng cách sử dụng ngôn ngữ lập trình PHP, cơ sở dữ liệu MySql, và các công nghệ web bao gồm HTML, CSS, JavaScript, Ajax cũng được áp dụng trong dự án. Ngoài ra hệ thống dựa trên cơ sở tích hợp đầy đủ các quy trình nghiệp vụ đánh giá và thực hiện các công việc theo quy trình phát triển phần mềm bao gồm quá trình phân tích các yêu cầu hệ thống, thiết kế giao diện người dùng, lập trình và kiểm thử hệ thống.

## MỤC TIÊU:

Đề tài xây dựng “Hệ thống quản lý nhà hàng” được xây dựng xuất phát từ nhu cầu thực tế, xử lý được nhiều nghiệp vụ, giảm bớt các công việc thủ công, giúp cho việc quản lý kinh doanh tại nhà hàng hiệu quả, nhanh chóng, chính xác hơn. Nhân viên sẽ dễ dàng nhập hóa đơn thanh toán theo bàn ở từng sảnh/khu vực. Hệ thống hỗ trợ người quản lý thống kê những món ăn bán chạy nhất theo tháng. Hệ thống đáp ứng cá yêu cầu về tốc độ, giao diện, độ chính xác cao. Mô hình hóa được thực tế vào máy tính để người dùng tiện lợi, thao tác dễ dàng.

## PHẠM VI VÀ KẾT QUẢ CẦN ĐẠT ĐƯỢC

* 1. Phạm vi nghiên cứu:
* Nghiên cứu xây dựng hệ thống quản lý nhà hàng. Tin học hóa việc quản lý nhà hàng. Nghiên cứu các nghiệp vụ của một nhà hàng trên thực tế.
* Nghiên cứu Laravel Framework.
* Hệ quản trị cơ sở dữ liệu MySQL.
  1. Kết quả cần đạt được:

Đề tài xây dựng hệ thống quản lý nhà hàng cần đảm bảo các chức năng sau:

* Đối tượng quản lý nhà hàng:
* Quản lý danh mục.
* Quản lý khách hàng.
* Quản lý món ăn.
* Quản lý thu, chi.
* Quản lý khu vực.
* Quản lý hóa đơn.
* Quản lý đặt tiệc.
* Thống kê.
* Đối tượng khách hàng:
* Đăng kí.
* Đặt bàn.
* Xem menu.

## KẾ HOẠCH THỰC HIỆN

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **TT** | **CÔNG VIỆC** | **TUẦN** | | | | | | | | | | | | | | | | |
| **1** | **2** | **3** | **4** | **5** | **6** | **7** | **8** | **9** | **10** | **11** | **12** | **13** | **14** | **15** | **16** | **17** |
| 1 | Khảo sát hệ thống |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| 2 | Phân tích yêu cầu |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| 3 | Thiết kê mô hình |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| 4 | Cài đặt chương trình |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| 5 | Nhập liệu và kiểm thử |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| 6 | Sửa lỗi chương trình |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| 7 | Viết báo cáo |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |

# CHƯƠNG 2: CƠ SỞ LÝ THUYẾT

1. **HỆ QUẢN TRỊ CƠ SỞ DỮ LIỆU MYSQL:**
2. **MySQL là gì?**

MySQL là hệ quản trị cơ sở dữ liệu tự do nguồn mở phổ biến nhất thế giới và được các nhà phát triển rất ưa chuộng trong quá trình phát triển ứng dụng. Vì MySQL là cơ sở dữ liệu tốc độ cao, ổn định và dễ sử dụng, có tính khả chuyển, hoạt động trên nhiều hệ điều hành cung cấp một hệ thống lớn các hàm tiện ích rất mạnh. Với tốc độ và tính bảo mật cao, MySQL rất thích hợp cho các ứng dụng có truy cập CSDL trên internet. MySQL miễn phí hoàn toàn cho nên bạn có thể tải về MySQL từ trang chủ. Nó có nhiều phiên bản cho các hệ điều hành khác nhau: phiên bản Win32 cho các hệ điều hành dòng Windows, Linux, Mac OS X, Unix,FreeBSD, NetBSD, Novell NetWare, SGI Irix, Solaris, SunOS,…

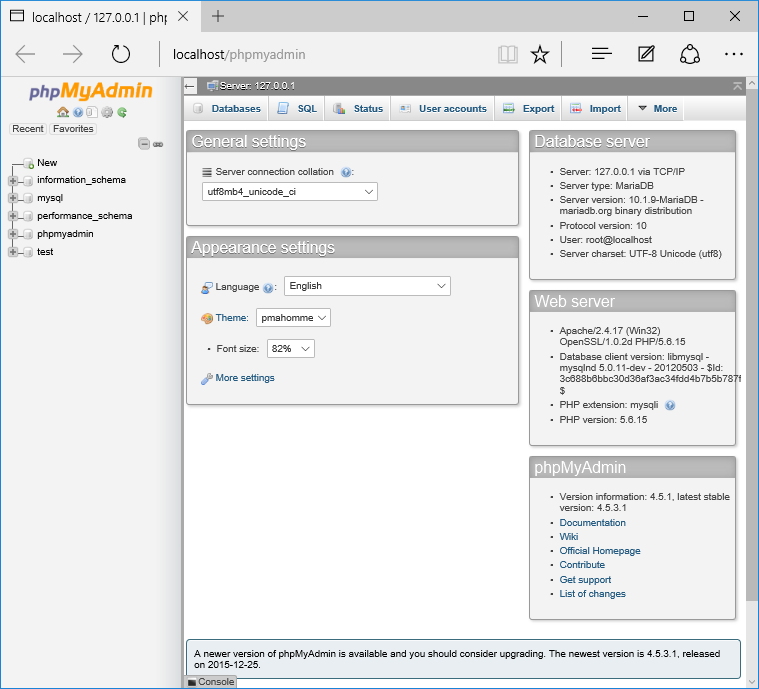
MySQL là một trong những ví dụ rất cơ bản về Hệ Quản trị Cơ sở dữ liệu quan hệ sử dụng Ngôn ngữ truy vấn có cấu trúc (SQL).

MySQL được sử dụng cho việc bổ trợ PHP, Perl, và nhiều ngôn ngữ khác, nó làm nơi lưu trữ những thông tin trên các trang web viết bằng PHP hay Perl,…

1. **Tìm hiểu về phpMyAdmin:**

**phpMyAdmin** là phần mềm mã nguồn mở được viết bằng ngôn ngữ PHP nhằm giúp người dùng (thường là các nhà quản trị cơ sở dữ liệu hay database administrator) có thể quản lý cơ sở dữ liệu MySQL thông qua giao diện web thay vì sử dụng giao diện cửa sổ dòng lệnh (command line interface).

Sử dụng **phpMyadmin** người dùng có thể thực hiện được nhiều tác vụ khác nhau như khi sử dụng cửa sổ dòng lệnh. Các tác vụ này bao gồm việc tạo, cập nhật và xoá các cơ sở dữ liệu, các bảng, các trường, dữ liệu trên bảng, phân quyền và quản lý người dùng....



1. **NGÔN NGỮ LẬP TRÌNH PHP:**
2. **Tổng quan về PHP:**

**PHP** - viết tắt hồi quy của **"Hypertext Preprocessor"**, là một ngôn ngữ lập trình kịch bản được chạy ở phía server nhằm sinh ra mã html trên client. PHP đã trải qua rất nhiều phiên bản và được tối ưu hóa cho các ứng dụng web, với cách viết mã rõ rãng, tốc độ nhanh, dễ học nên PHP đã trở thành một ngôn ngữ [lập trình web](http://hoclaptrinhweb.org/hoc-thiet-ke-web/lap-trinh-web.html) rất phổ biến và được ưa chuộng.

**PHP** chạy trên môi trường Webserver và lưu trữ dữ liệu thông qua hệ quản trị cơ sở dữ liệu nên PHP thường đi kèm với Apache, MySQL và hệ điều hành Linux (LAMP).

* Apache là một phần mềm web server có nhiệm vụ tiếp nhận request từ trình duyệt người dùng sau đó chuyển giao cho PHP xử lý và gửi trả lại cho trình duyệt.
* MySQL cũng tương tự như các hệ quản trị cơ sở dữ liệu khác (Postgress, Oracle, SQL server...) đóng vai trò là nơi lưu trữ và truy vấn dữ liệu.
* Linux: Hệ điều hành mã nguồn mở được sử dụng rất rộng rãi cho các webserver. Thông thường các phiên bản được sử dụng nhiều nhất là RedHat Enterprise Linux, Ubuntu...

1. **Lịch sử phát triển của PHP:**

**PHP** mà chúng ta biết đến ngày nay là sự kế thừa của một sản phẩm có tên là PHP/FI. Được viết bởi Rasmus Lerdorf, bản PHP đầu tiên là một tập hợp đơn giản các Common Gateway Interface (CGI) – một chuẩn giao tiếp giữa client và server – nhị phân viết bằng ngôn ngữ C. Tháng 9/1995, Rasmus mở rộng PHP, thêm vào công cụ có tên là FI (Forms Interpreter), và một số chức năng như chúng ta biết ngày nay. Nó có các biến kiểu Perl và cú pháp cho phép nhúng mã HTML. Cú pháp này, tương tự Perl, nhưng hạn chế hơn, đơn giản và có phần không phù hợp. Trong thực tế, để nhúng mã lệnh PHP/FI vào một tập tin HTML, các lập trình viên phải sử dụng các comment HTML. Và mặc dù phương pháp này không được hoàn toàn đón nhận, FI vẫn tiếp tục phát triển và đem về lợi nhuận thương mại.

**PHP 3.0:** Được Andi Gutmans và Zeev Suraski tạo ra năm 1997 sau khi viết lại hoàn toàn bộ mã nguồn trước đó. Lý do chính mà họ đã tạo ra phiên bản này là do họ nhận thấy PHP/FI 2.0 hết sức yếu kém trong việc phát triển các ứng dụng thương mại điện tử. PHP 3.0 như là phiên bản thế hệ kế tiếp của PHP/FI 2.0, và chấm dứt phát triển PHP/FI 2.0.PHP 3.0 cung cấp cho người dùng cuối một cơ sở hạ tầng chặt chẽ dùng cho nhiều cơ sở dữ liệu, giao thức và API khác nhau.Cho phép người dùng có thể mở rộng theo modul.  
=> Chính điều này làm cho PHP3 thành công so với PHP2. Lúc này họ chính thức đặt tên ngắn gọn là 'PHP' ( Hypertext Preprocessor ). PHP 3.0 đã chính thức được công bố vào tháng 6 năm 1998.

**PHP4:** Được công bố năm 2000 tốc độ xử lý được cải thiện rất nhiều, PHP 4.0 đem đến các tính năng chủ yếu khác gồm có sự hỗ trợ nhiều máy chủ Web hơn, hỗ trợ phiên làm việc HTTP, tạo bộ đệm thông tin đầu ra, nhiều cách xử lý thông tin người sử dụng nhập vào bảo mật hơn và cung cấp một vài các cấu trúc ngôn ngữ mới. Với PHP 4, số nhà phát triển dùng PHP đã lên đến hàng trăm nghìn và hàng triệu site đã công bố cài đặt PHP, chiếm khoảng 20% số tên miền trên mạng Internet. Nhóm phát triển PHP cũng đã lên tới con số hàng nghìn người và nhiều nghìn người khác tham gia vào các dự án có liên quan đến PHP như PEAR, PECL và tài liệu kĩ thuật cho PHP.

**PHP5:** Bản chính thức đã ra mắt ngày 13 tháng 7 năm 2004 sau một chuỗi khá dài các bản kiểm tra thử bao gồm Beta 4, RC 1, RC2, RC3. Mặc dù coi đây là phiên bản sản xuất đầu tiên nhưng PHP 5.0 vẫn còn một số lỗi trong đó đáng kể là lỗi xác thực HTTP.

**PHP6:** Hiện nay phiên bản tiếp theo của PHP đang được phát triển, PHP 6 bản sử dụng thử đã có thể được download tại địa chỉ snaps.php.net. Phiên bản PHP 6 được kỳ vọng sẽ lấp đầy những khiếm khuyết của PHP ở phiên bản hiện tại.

1. **KHÁI QUÁT VỀ PHP FRAMEWORK:**

Để phát triển một website hệu quả, thuận lợi và không tốn công xây dựng một hệ thống từ đầu thì các lập trình viên thường sử dụng PHP Framewwork. PHP Framewwork là một thư viện các hàm xây dựng sẵn, mục địch chính là hỗ trợ cho việc phát triển website bằng ngôn ngữ PHP. Khi sử dụng Framewwork, website sẽ có cấu trúc hệ thống rõ ràng và chặt chẽ, thời gian để phát triển website nhanh hơn và giúp hệ thống bảo mật hơn. Một PHP Framework thường được xây dựng trên kiến trúc MVC.

Ngoài ra, khi phát triển các website động dựa vào công nghệ PHP – MySQL, các công ty phát triển phần mềm đều sử dụng các PHP Framework như: Laravel, Zend, CakePHP, Symfony,…

1. **LARAVEL FRAMEWORK**
2. **Giới thiệu về Laravel Framwork:**

**Laravel** là một PHP framework  mã nguồn mở và miễn phí, được phát triển bởi Taylor Otwell và nhắm vào mục tiêu hỗ trợ phát triển các ứng dụng web theo kiếm trúc model-view-controller (MVC). Những tính năng nổi bật của Laravel bao gồm cú pháp dễ hiểu – rõ ràng , một hệ thống đóng gói modular và quản lý gói phụ thuộc, nhiều cách khác nhau để truy cập vào các cơ sở dữ liệu quan hệ, nhiều tiện ích khác nhau hỗ trợ việc triển khai vào bảo trì ứng dụng.

Vào khoảng Tháng 3 năm 2015, các lập trình viên đã có một cuộc bình chọn PHP framework phổ biến nhất, Laravel đã giành vị trí quán quân cho PHP framework phổ biến nhất năm 2015, theo sau lần lượt là Symfony2, Nette, CodeIgniter, Yii2 vào một số khác. Trước đó, Tháng 8 2014, Laravel đã trở thành project PHP phổ biến nhất và được theo dõi nhiều nhất trên Github.

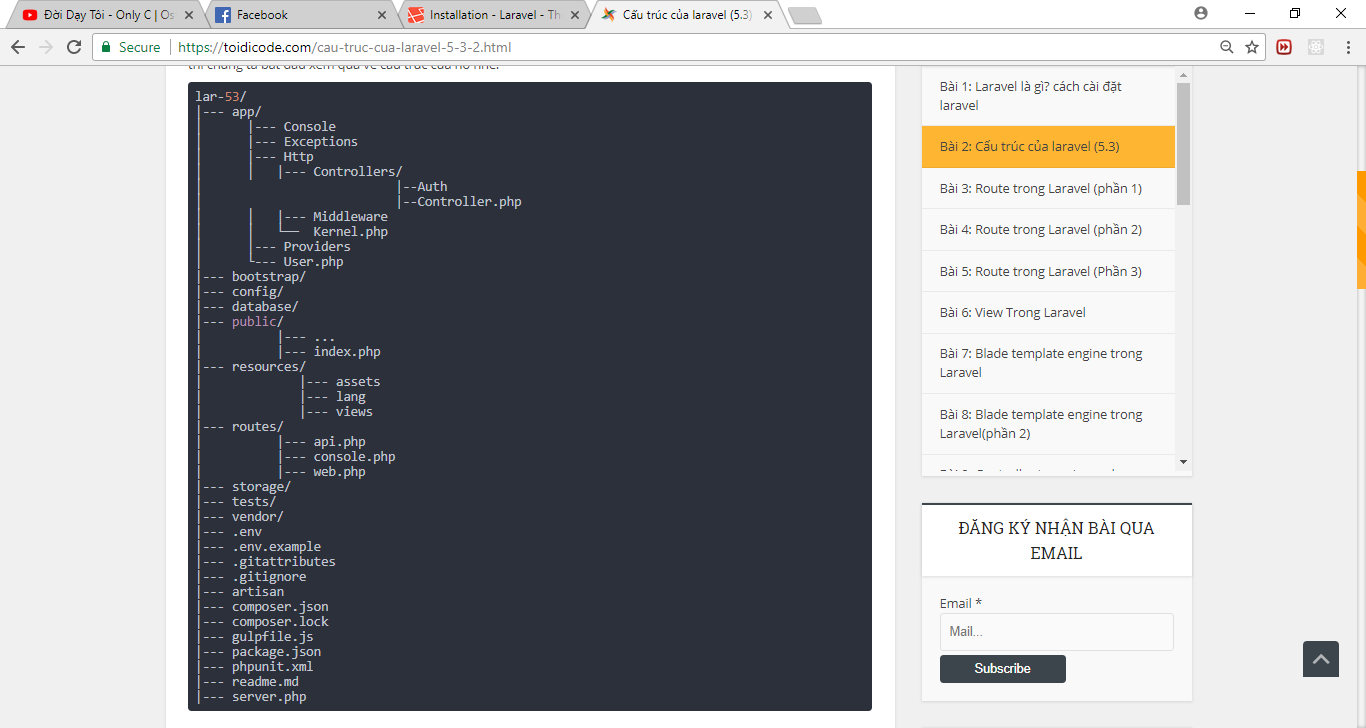
Laravel được phát hành theo giấy phép MIT, với source code được lưu trữ tại [Github](https://github.com/laravel/laravel).

1. **Lịch sử phát triển của Laravel:**

**Laravel** được Taylor Otwell tạo ra như một giải pháp thay thế cho CodeIgniter, cung cấp nhiều tính năng quan trọng hơn  như xác thực và phân quyền. Tôi cũng không chắc về điều này, nhưng có thể Taylor vốn là một .NET developer khi bắt đầu có nhu cầu làm việc với PHP khoảng vào những năm 2010-2011, đã chọn CodeIgniter khi đó đang là một ngôi sao mới nổi, thậm chí lấn át cả Symfony gạo cội. Và Taylor nhanh chóng nhận ra những điểm khiếm khuyết ở CodeIgniter, với tài năng và kiến thức xuất sắc về design-pattern của mình, Taylor quyết định tự mình tạo ra một framework sao cho thật đơn giản, dễ hiểu, hỗ trợ lập trình viên hiện thực ý tưởng một cách nhanh nhất bằng nhiều tính năng hỗ trợ như Eloquent ORM mạnh mẽ, xác thực đơn giản, phân trang hiệu quả, và hơn thế nữa.

* **Bản Laravel beta đầu tiên** được phát hành vào ngày 9/6/2011, tiếp đó là **Laravel 1** phát hành trong cùng tháng. Laravel 1 bao gồm các tính năng như xác thực, bản địa hóa, model, view, session, định tuyến và các cơ cấu khác, nhưng vẫn còn thiếu controller, điều này làm nó chưa thật sự là một MVC framework đúng nghĩa.
* **Laravel 2** được phát hành vào tháng 9 năm 2011, mang đến nhiều cài tiến từ tác giả và cộng đồng. Tính năng đáng kể bao gồm hỗ trợ controller, điều này thực sự biến Laravel 2 thành một MVC framework hoàn chỉnh, hỗ trợ Inversion of Control ([IoC](https://en.wikipedia.org/wiki/Inversion_of_control)), hệ thống template Blade. Bên cạnh đó, có một nhược điểm là hỗ trợ cho các gói của nhà phát triển bên thứ 3 bị gỡ bỏ.
* **Laravel 3** được phát hành vào tháng 2 năm 2012, với một tấn tính năng mới bao gồm giao diện dòng lệnh (CLI) tên “Artisan”, hỗ trợ nhiều hơn cho hệ thống quản trị cơ sở dữ liệu, chức năng ánh xạ cơ sở dữ liệu Migration, hỗ trợ “bắt sự kiện” trong ứng dụng,  và hệ thống quản lý gói gọi là “Bundles”. Lượng người dùng và sự phổ biến tăng trưởng mạnh kể từ phiên bản Laravel 3.
* **Laravel 4**, tên mã “Illuminate”, được phát hành vào tháng 5 năm 2013. Lần này thực sự là sự lột xác của Laravel framework, di chuyển và tái cấu trúc các gói hỗ trợ vào một tập được phân phối thông qua Composer, một chương trình quản lý gói thư viện phụ thuộc độc lập của PHP. Bố trí mới như vậy giúp khả năng mở rộng của Laravel 4 tốt hơn nhiều so với các phiên bản trước. Ra mắt lịch phát hành chính thức mỗi sáu tháng một phiên bản nâng cấp nhỏ. các tính năng khác trong Laravel 4 bao gồm tạo và thêm dữ liệu mẫu (database seeding), hỗ trợ hàng đợi, các kiểu gởi mail, và hỗ trợ “xóa mềm”  (soft-delete: record bị lọc khỏi các truy vấn từ Eloquent mà không thực sự xóa hẳn khỏi DB).
* **Laravel 5** được phát hành trong tháng 2 năm 2015, như một kết quả thay đổi đáng kể cho việc kết thúc vòng đời nâng cấp Laravel lên 4.3. Bên cạnh một loạt tính năng mới và các cải tiến như hiện tại, Laravel 5 cũng giới thiệu cấu trúc cây thư mục nội bộ cho phát triển ứng dụng mới. Những tính năng mới của Laravel 5 bao gồm hộ trợ lập lịch định kỳ thực hiện nhiệm vụ thông qua một gói tên là “Scheduler”, một lớp trừu tượng gọi là “Flysystem” cho phép điều khiển việc lưu trữ từ xa đơn giản như lưu trữ trên máy local – dễ thấy nhất là mặc định hỗ trợ dịch vụ Amazone S3, cải tiến quản lý assets thông qua “Elixir”, cũng như đơn giản hóa quản lý xác thực với các dịch vụ bên ngoài bằng gói “Socialite”.
* **Laravel 5.1** phát hành vào tháng 6 năm 2015, là bản phát hành đầu tiên nhận được hỗ trợ dài hạng (LTS) với một kết hoạch fix bug lên tới 2 năm vào hỗ trợ vá lỗi bảo mật lên tới 3 năm. Các bản phát hành LTS của Laravel được lên kế hoạch theo mỗi 2 năm.

1. **Cấu trúc lưu trữ của Laravel:**



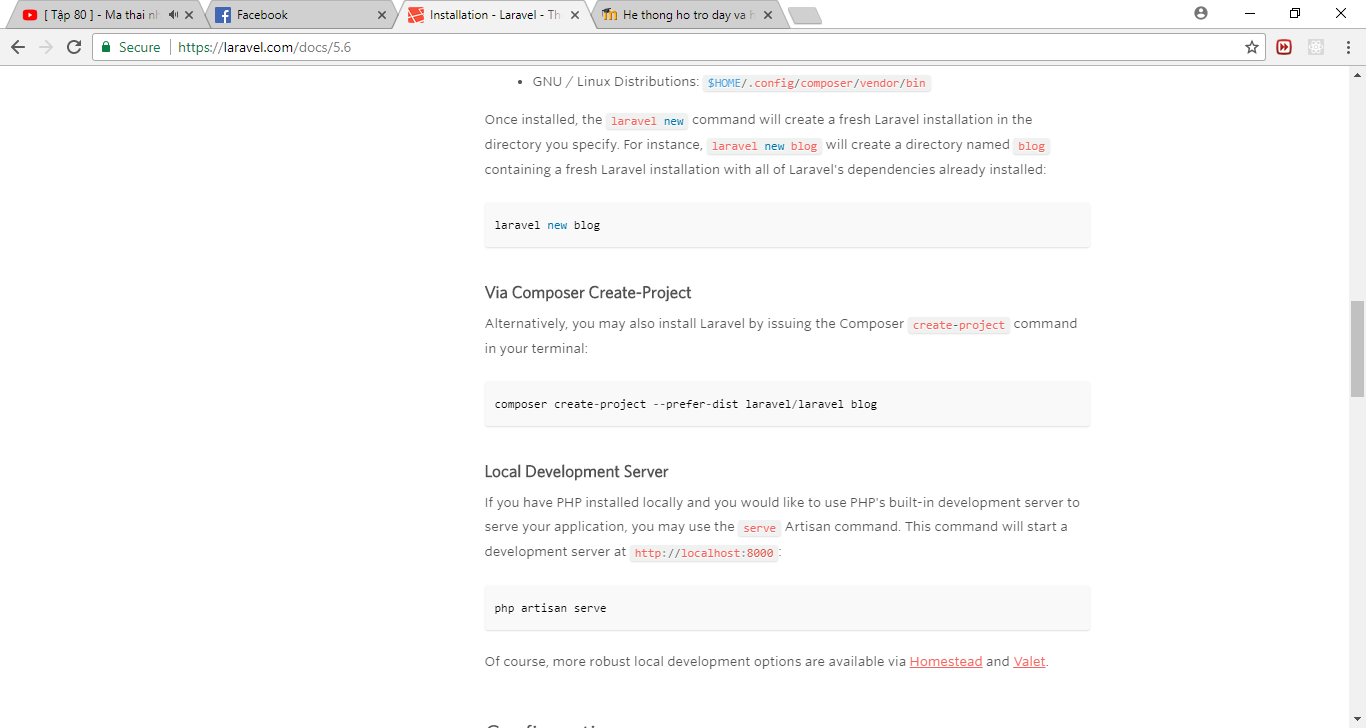
* **app:** Chứa các thư mục, các tập tin php, thư viện, models.
  + **Console:** Chứa các tập tin định nghĩa các câu lệnh trên artisan.
  + **Excerption:**  Chứa các tập tin quản lý, điều hướng lỗi.
  + **Http**
    - **Controllers** : Chứa các controllers của project.
    - **Middleware:**Chứa các tập tin lọc và ngăn chặn các requests.
    - **Kernel.php:**Cấu hình, định nghĩa Middleware.
  + **Providers:**Chứ các providers thực hiện việc **binding vào service container**(ở phần nâng cao mình sẽ nói sau).
  + **User.php:**Là model User mà Laravel tự địn sẵn cho chúng ta.
* **bootstrap:**Chứa tập tin điều hướng hệ thống.
* **config:** Chứa mọi tập tin cấu hình của Laravel.
* **database:** Chứa các thư mục tập tin vềdatabase.
  + **migrations:** Chứa các tập tin định nghĩa,khởi tạo và sửa bảng.
  + **seeds:** Chứa các tập tin định nghĩa dữ liệu insert(thêm) vào trong database.
  + **factories:** Chứa các tập tin định nghĩa các cột bảng dữ liệu để tạo ra các dữ liệu ảo.
* **public:** Chứa các tập tin css, js, image.
  + **index.php:**Đây là tệp tin root của Laraver
* **resources:** Chứa các views, ngôn ngữ(language) của project.
* **routes:** Chứa các tập tin định nghĩa các router, xử lý điều hướng router bao gồm: **web, api và console**(Mình sẽ nói thêm ở phần sau).
* **storage:** Chứa các tập tin hệ thống cache, session, ...
* **tests: ...**
* **vendor:** Chứa các thư viện của composer.
* **.env:** Là tập tin cấu hình chính của laravel như key app,database.
* **.env.example:**Tệp tin cấu hình mẫu của laraver.
* **composer.json:** tập tin của composer.
* **composer.lock:** tập tin của composer.
* **package.js:** Tập tin cấu hình của [nodejs](http://toidicode.com/nodejs-la-gi-26.html) (chứa các package cần dùng cho projects).
* **gulpfile.js:** Là tập tin gulp builder.
* **phpunit.xml:** Là tập tin xml của phpunit dùng để testing project.
* **server.php:** Là tập tin để artisan trỏ đến tạo server khi gõ lệnh php artisan serve.
* **artisan:** Tập tin thực thi lệnh của Laravel.

1. **Những ưu điểm của Laravel so với các PHP Framework khác:**

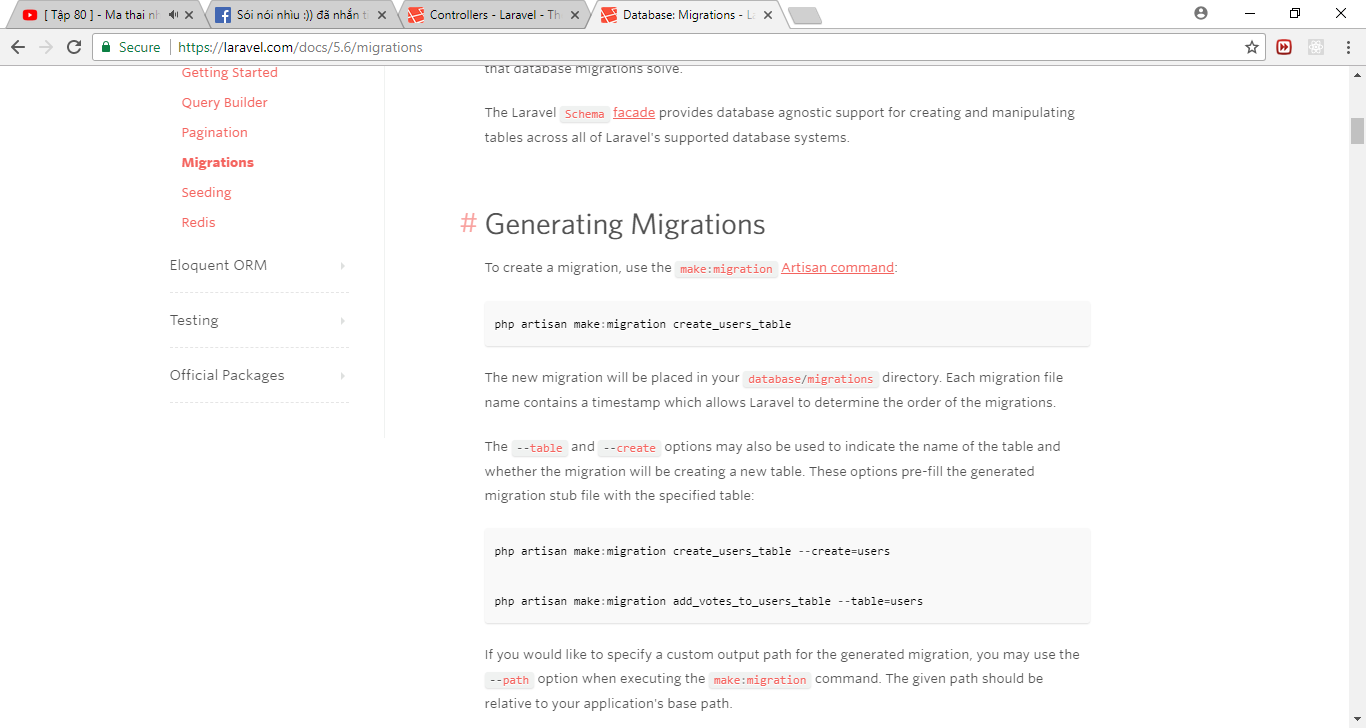
* Thừa hưởng những thế mạnh và khắc phục các nhược điểm của các Framework trước đó.
* Mức độ cạnh tranh cao vì được cập nhật liên tục nên các lỗi được khắc phục nhanh.
* Tài liệu hướng dẫn được thiết kế đầy đủ và rõ ràng
* Dễ dàng được hỗ trợ bởi cộng đồng vì hiện tại Laravel có số lượng người dùng nhiều nhất.
* Dễ dàng quản lý và tích hợp các thư viện vào dự án dựa vào Composer.
* Tự động nạp thư viện mã lệnh (autoload) theo không gian tên (namespace).
* Tính năng Migration giúp quản lý cơ sở dữ liệu hiệu quả hơn khi làm việc nhóm.
* Các lớp Eloquent giúp đơn giản hóa các thao tác xử lý cơ sở dữ liệu đối với quan hệ 1 – N và quan hệ N – N.
* Eloquent ORM (Object Relational Mapping): đây là một tính năng tuyệt vời, hỗ trợ việc quản lý và truy vết các thao tác truy vấn cơ sở dữ liệu. Tương thích với hầu hết các hệ quản trị cơ sở dữ liệu như MySQL, SQL Server, SQLite,…
* Cơ chế Route trong Laravel thật sự khác biệt và đầy mạnh mẽ so với các Framework khác.
* Việc quản lý bố cục rất đơn giản dựa vào Blade Template

1. **Các lệnh cơ bản của Laravel:**

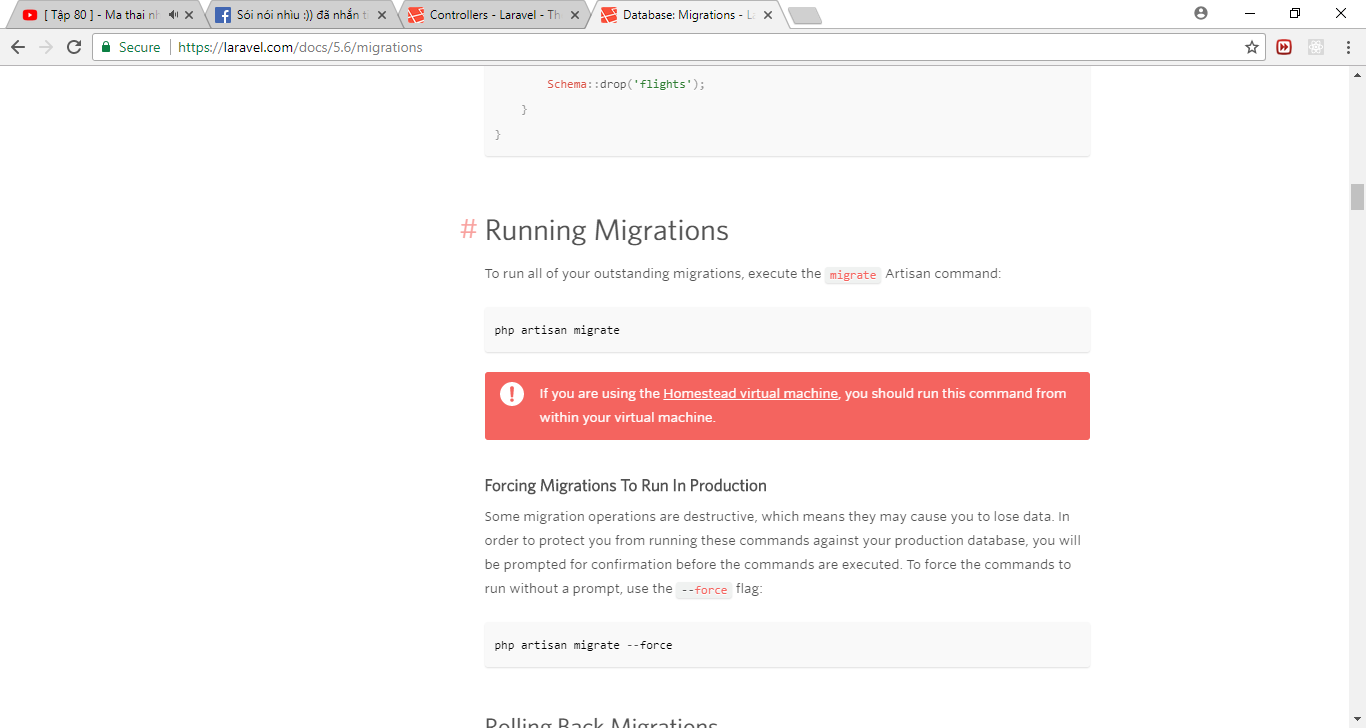
Tạo Project:



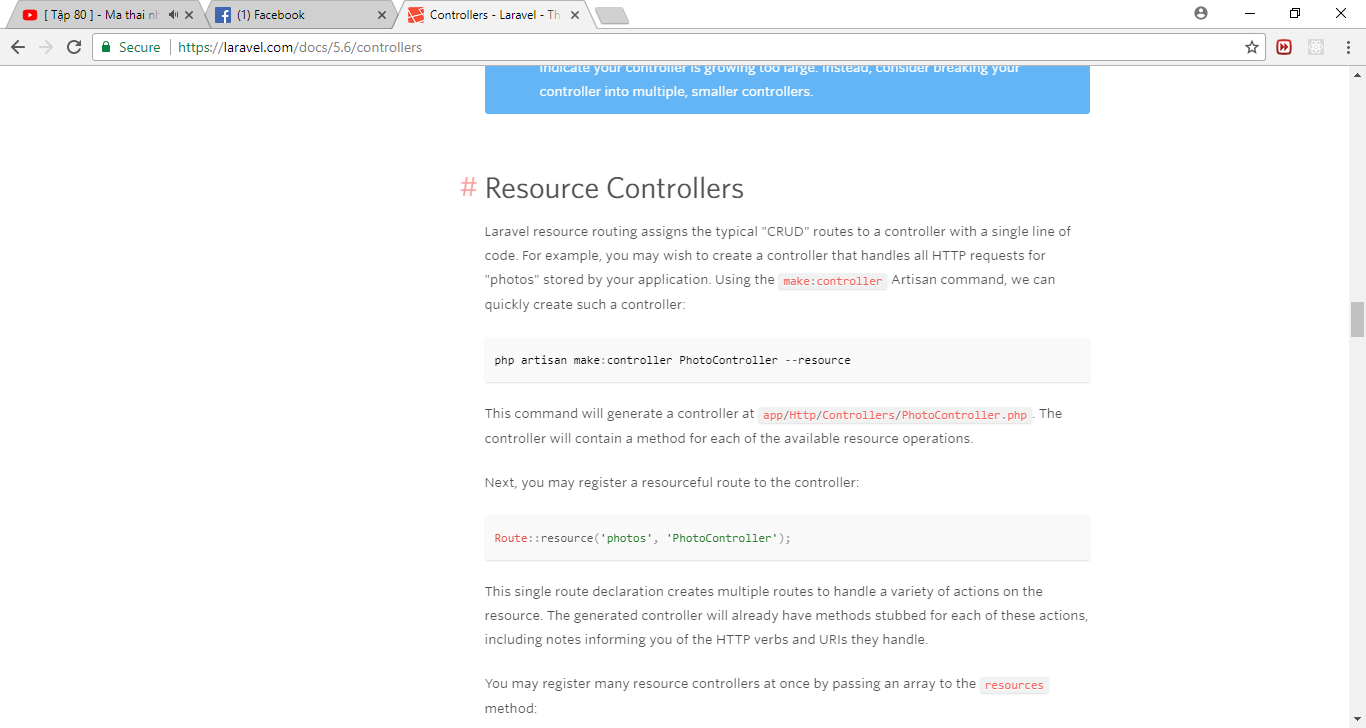
Tạo Migration:



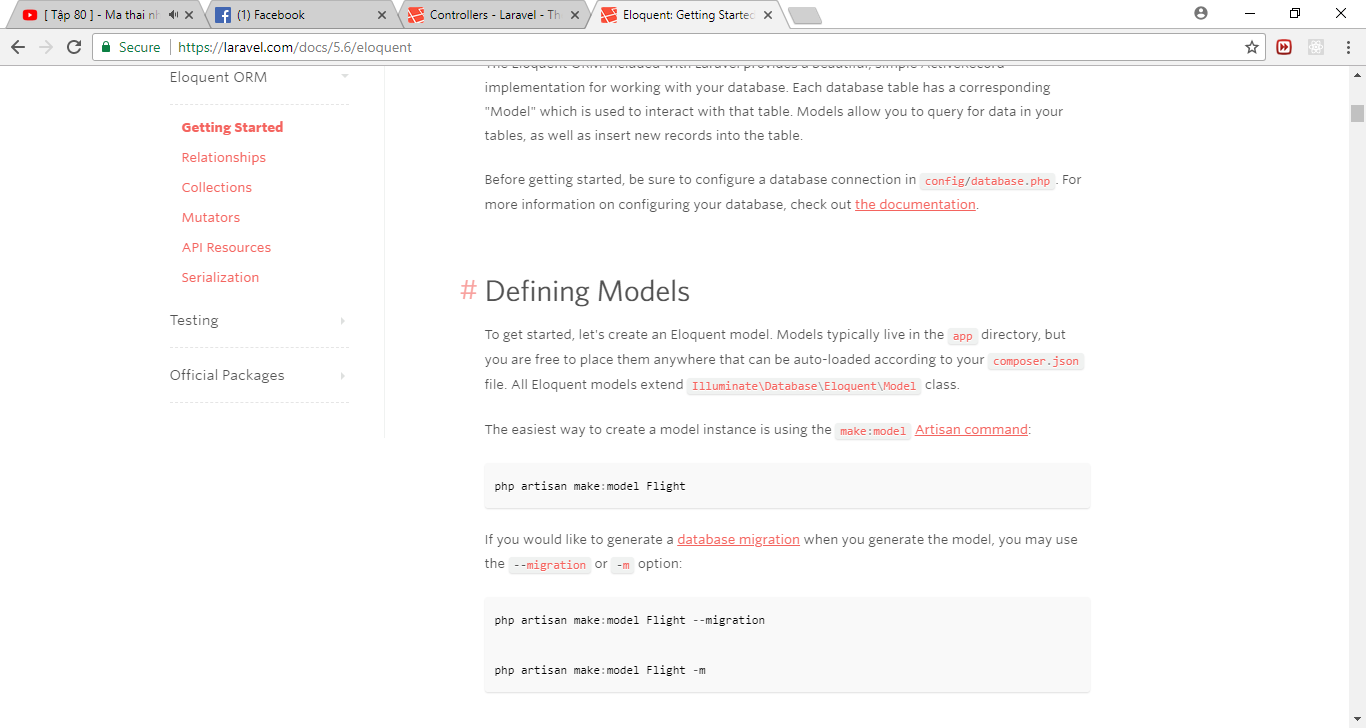
Migrate dữ liệu:



Tạo Controllers:



Tạo Model:



1. **Mô hình MVC:**

# CHƯƠNG 3: NỘI DUNG VÀ KẾT QUẢ THỰC HIỆN

## MÔ TẢ HỆ THỐNG

### Mô tả yêu cầu:

Hiện nay việc ứng dụng công nghệ thông tin vào các hệ thống quản lý ngày càng phổ biến, trong đó có các nhà hàng để giúp tối ưu hóa công tác quản lý, từ đó dần nâng cao chất lượng phục vụ. Hệ thống quản lý nhà hàng MONVIET phục vụ nhu cần quản lý của nhà hàng bao gồm quản lý nhà hàng, nhân viên thu ngân và khách hàng.

Nhà hàng có nhiều sảnh/khu vực, các sảnh có nhiều bàn, nhân viên thu ngân nhận ca có thể thanh toán ở bất cứ sảnh/khu vực hoặc bàn.

Khi khách đến nhà hàng nhân viên phục vụ sẽ ghi order của khách trên giấy và đem vào cho nhân viên thu ngân nhập order vào hệ thống. Order có thể thay đổi món. Nhân viên thu ngân chọn sảnh, chọn bàn mà khách ngồi và nhập vào order của khách từ thực đơn của nhà hàng. Khi khách đi nhóm nhiều người, thu ngân có thể chọn ghép nhiều bàn với nhau theo qui trình: chọn bàn đầu tiên muốn ghép, sau đó chọn cái bàn tiếp theo, order sẽ đc nhập vào bàn a+b+... Khi khách hàng muốn thanh toán nhân viên thu ngân sẽ tiến hành thanh toán và xuất hóa đơn sau đó giao nhân viên phục vụ đem đến khách và xác nhận thanh toán của khách hàng, trả tiền thừa.

Ngoài ra nhà hàng còn có website riêng, ở đây sẽ giới thiệu các món ăn trong thực đơn của nhà hàng. Khách hàng khi truy cập website còn có thể đăng ký thành viên hoặc đặt tiệc.

Để có thể đặt tiệc, khách hàng cần đăng ký tài khoản thành viên trên trang web, các món ăn sẽ được hiển thị trên website, khách hàng chọn món ăn mình muốn đặt, sau khi chọn món xong, khách hàng tiếp tục chọn thời gian và số lượng khách, số lượng bàn muốn đặt. Món ăn tại mỗi bàn sẽ giống nhau. Sao khi hoàn tất đặt thì nhân viên sẽ liên hệ với khách để xác nhận.

Món ăn của nhà hàng bao gồm các loại: món khai vị, món gỏi-salad, món lẩu, món nướng, cơm, thức uống,...mỗi loại sẽ có nhiều món ăn. Thực đơn bao gồm: Thực đơn ăn chay, thực đơn tiệc buffet, thực đơn gia đình, thực đơn thanh đạm,...Quản lý nhà hàng có thể cập nhật (tạo mới, sửa, xóa, tìm kiếm) loại món ăn, món ăn, thực đơn theo tình hình thực tế của nhà hàng. Hệ thống sẽ giúp thống kê món ăn bán chạy, doanh thu của nhà hàng theo ngày, tháng và in báo cáo từ đó giúp nhà hàng có thể điều chỉnh kế hoạch kinh doanh phù hợp nhất. Khi khách hàng đặt tiệc trên website nhà hàng đơn đặt tiệc sẽ được lưu vào hệ thống với trạng thái chờ, quản lý sẽ dựa vào thông tin khách hàng trong đơn và liên hệ với khách hàng.

### Các biểu mẫu liên quan

## THIẾT KẾ THÀNH PHẦN DỮ LIỆU

### Mô hình dữ liệu mức quan niệm



### Mô hình dữ liệu mức vật lý

**2.2.1. Bảng:**

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Tên Trường** | **Kiểu Dữ Liệu** | **Kích Thước** | **Giá Trị Mặc Định** | **Khóa Chính** | **Duy Nhất** | **Not Null** | **Khóa Ngoại** | **Diễn Giải** |
| NV\_MA | Integer |  |  | X | X | X |  | Mã nhân viên |
| NV\_TAIKHOAN | Nvachar | 32 |  |  | X | X |  | Tài khoản nhân viên |
| NV\_MATKHAU | Nvachar | 25 |  |  |  | X |  | Mật khẩu nhân viên |
| NV\_HOTEN | Nvachar | 100 |  |  |  | X |  | Họ tên nhân viên |
| NV\_GIOITINH | Int | 3 |  |  |  | X |  | Giới tính nhân viên |
| NV\_EMAIL | Nvachar | 200 |  |  | X | X |  | Email nhân viên |
| NV\_NGAYSINH | DateTime | 10 |  |  |  | X |  | Ngày sinh nhân viên |
| NV\_DIACHI | Nvachar | 225 |  |  |  | X |  | Địa chỉ nhân viên |
| NV\_DIENTHOAI | Int | 12 |  |  |  | X |  | Điện thoại nhân viên |
| NV\_TAOMOI | DateTime |  |  |  |  | X |  | Tạo mới nhân viên |
| NV\_CAPNHAT | DateTime |  |  |  |  | X |  | Cập nhật nhân viên |
| NV\_TRANGTHAI | Tinyint |  | 2 |  |  | X |  | Trạng thái nhân viên.(1. Khóa, 2. Khả dụng) |

**NHANVIEN(NV\_MA,NV\_TAIKHOAN,NV\_MATKHAU, NV\_HOTEN, NV\_GIOITINH, NV\_EMAIL, NV\_NGAYSINH, NV\_DIACHI, NV\_DIENTHOAI, NV\_TAOMOI, NV\_CAPNHAT, NV\_TRANGTHAI)**

**2.2.2. Bảng:**

**KHACHHANG(KH\_MA, KH \_HOTEN, KH \_GIOITINH, KH \_EMAIL, KH \_NGAYSINH, KH \_DIACHI, KH \_DIENTHOAI, KH \_TAOMOI, KH \_CAPNHAT, KH \_TRANGTHAI, KH\_TAIKHOAN, KH\_MATKHAU )**

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Tên Trường** | **Kiểu Dữ Liệu** | **Kích Thước** | **Giá Trị Mặc Định** | **Khóa Chính** | **Duy Nhất** | **Not Null** | **Khóa Ngoại** | **Diễn Giải** |
| KH\_MA | Integer |  |  | X | X | X |  | Mã khách hàng |
| KH\_TAIKHOAN | Nvachar | 32 |  |  | X | X |  | Tài khoản khách hàng |
| KH\_MATKHAU | Nvachar | 25 |  |  |  | X |  | Mật khẩu khách hàng |
| KH\_HOTEN | Nvachar | 100 |  |  |  | X |  | Họ tên khách hàng |
| KH\_GIOITINH | Int | 3 |  |  |  | X |  | Giới tính khách hàng |
| KH\_EMAIL | Nvachar | 200 |  |  | X | X |  | Email khách hàng |
| KH\_NGAYSINH | DateTime | 10 |  |  |  | X |  | Ngày sinh khách hàng |
| KH\_DIACHI | Nvachar | 225 |  |  |  | X |  | Địa chỉ khách hàng |
| KH\_DIENTHOAI | Int | 12 |  |  |  | X |  | Điện thoại khách hàng |
| KH\_TAOMOI | DateTime |  | Current\_timestamp |  |  | X |  | Tạo mới khách hàng |
| KH\_CAPNHAT | DateTime |  | Current\_timestamp |  |  | X |  | Cập nhật khách hàng |
| KH\_TRANGTHAI | Tinyint |  | 2 |  |  | X |  | Trạng thái khách hàng(1. Khóa, 2. Khả dụng) |

**2.2.3. Bảng:**

**LOAI(L\_MA, L \_TEN, L\_TAOMOI, L\_CAPNHAT, L\_TRANGTHAI)**

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Tên Trường** | **Kiểu Dữ Liệu** | **Kích Thước** | **Giá Trị Mặc Định** | **Khóa Chính** | **Duy Nhất** | **Not Null** | **Khóa Ngọai** | **Diễn Giải** |
| L\_MA | Integer |  |  | X | X | X |  | Mã loại món ăn |
| L\_TEN | Nvachar | 100 |  |  |  | X |  | Tên loại món ăn |
| L\_TAOMOI | DateTime |  | Current\_timestamp |  |  | X |  | Tạo mới loại món ăn |
| L\_CAPNHAT | DateTime |  | Current\_timestamp |  |  | X |  | Cập nhật loại món ăn |
| L\_TRANGTHAI | Tinyint |  | 2 |  |  | X |  | Trạng thái loại(1. Khóa, 2. Khả dụng) |

**2.2.4. Bảng:**

**MONAN(MA\_MA, L\_MA, MA\_TEN, MA\_GIABAN, MA\_HINH, MA\_TAOMOI, MA\_CAPNHAT, MA\_TRANGTHAI, MA\_THONGTIN)**

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Tên Trường** | **Kiểu Dữ Liệu** | **Kích Thước** | **Giá Trị Mặc Định** | **Khóa Chính** | **Duy Nhất** | **Not Null** | **Khóa Ngoài** | **Giá Trị Mặc Định** |
| MA\_MA | Integer |  |  | X | X | X |  | Mã món ăn |
| L\_MA | Integer |  |  |  | X | X | X | Mã loại |
| MA\_TEN | Nvachar | 100 |  |  |  | X |  | Tên món ăn |
| MA\_GIABAN | Int | 10 |  |  |  | X |  | Giá bán món ăn |
| MA\_HINH | Vachar | 10 |  |  |  | X |  | Hình món ăn |
| MA\_TAOMOI | DateTime |  | Current\_timestamp |  |  | X |  | Tạo mới món ăn |
| MA\_CAPNHAT | DateTime |  | Current\_timestamp |  |  | X |  | Cập nhật món ăn |
| MA\_TRANGTHAI | Tinyint |  | 2 |  |  | X |  | Trạng thái món ăn |
| MA\_THONGTIN | Nvachar | 100 |  |  |  | X |  | Thông tin món ăn |

**2.2.5. Bảng:**

**KHUVUC(KV\_MA, KV\_TEN, KV\_TAOMOI, KV\_CAPNHAT, KV\_TRANGTHAI)**

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Tên Trường** | **Kiểu Dữ Liệu** | **Kích Thước** | **Giá Trị Mặc Định** | **Khóa Chính** | **Duy Nhất** | **Not Null** | **Khóa Ngoài** | **Giá Trị Mặc Định** |
| KV\_MA | Integer |  |  | X | X | X |  | Mã khu vực |
| KV\_TEN | Nvachar | 100 |  |  |  | X |  | Tên khu vục |
| KV\_TAOMOI | DateTime |  | Current\_timestamp |  |  | X |  | Tạo mới khu vực |
| KV\_CAPNHAT | DateTime |  | Current\_timestamp |  |  | X |  | Cập nhật khu vực |
| KV\_TRANGTHAI | Tinyint |  | 2 |  |  | X |  | Trạng thái khu vực |

**2.2.6. Bảng:**

**BAN(B\_MA, KV\_MA, B\_TEN, B\_TAOMOI, B\_CAPNHAT, B\_TRANGTHAI)**

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Tên Trường** | **Kiểu Dữ Liệu** | **Kích Thước** | **Giá Trị Mặc Định** | **Khóa Chính** | **Duy Nhất** | **Not Null** | **Khóa Ngoài** | **Diễn Giải** |
| B\_MA | Integer |  |  | X | X | X |  | Mã bàn |
| KV\_MA | Integer |  |  |  |  | X | X | Mã khu vực |
| B\_TEN | Nvachar | 200 |  |  |  | X |  | Tên bàn |
| B\_TAOMOI | DateTime |  | Current\_timestamp |  |  | X |  | Ngày tạo mới bàn |
| B\_CAPNHAT | DateTime |  | Current\_timestamp |  |  | X |  | Ngày cập nhật bàn |
| B\_TRANGTHAI | Tinyint |  | 2 |  |  | X |  | Trạng thái bàn(1. Khóa, 2. Khả dụng) |

**2.2.7. Bảng:**

**COMBO(CB\_MA,CB \_TEN, CB\_MOTA, CB\_BATDAU, CB\_KETTHUC, CB\_DONGIA, CB\_TAOMOI, CB\_CAPNHAT, CB\_TRANGTHAI)**

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Tên Trường** | **Kiểu Dữ Liệu** | **Kích Thước** | **Giá Trị Mặc Định** | **Khóa Chính** | **Duy Nhất** | **Not Null** | **Khóa Ngoại** | **Diễn Giải** |
| CB\_MA | Integer |  |  | X |  | X |  | Mã combo |
| CB\_TEN | Nvachar | 100 |  |  |  | X |  | Tên combo |
| CB\_MOTA | Nvachar | 100 |  |  |  |  |  | Mô tả combo |
| CB\_BATDAU | DateTime |  | Current\_timestamp |  |  | X |  | Bắt đầu combo |
| CB\_KETTHUC | DateTime |  | Current\_timestamp |  |  | X |  | Kết thúc combo |
| CB\_DONGIA | Integer | 20 |  |  |  | X |  | Đơn giá combo |
| CB\_TAOMOI | DateTime |  | Current\_timestamp |  |  | X |  | Tạo mới combo |
| CB\_CAPNHAT | DateTime |  | Current\_timestamp |  |  | X |  | Cập nhật combo |
| CB\_TRANGTHAI | Tinyint |  | 2 |  |  | X |  | Trạng thái combo (1. Khóa, 2. Khả dụng) |

**2.2.8. Bảng:**

**CHITIETCOMBO(CB\_MA, MA\_MA,CTCB\_SOLUONG)**

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Tên Trường** | **Kiểu Dữ Liệu** | **Kích Thước** | **Giá Trị Mặc Định** | **Khóa Chính** | **Duy Nhất** | **Not Null** | **Khóa Ngoài** | **Giá Trị Mặc Định** |
| CB\_MA | Integer |  |  | X | X | X | X | Mã combo |
| MA\_MA | Integer |  |  | X | X | X | X | Mã món ăn |
| CTCB\_SOLUONG | Integer | 10 |  |  |  | X |  | Số lượng chi tiết combo |

**2.2.9. Bảng:**

**HOADONDATTIEC(PDT\_MA, HDT\_MA)**

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Tên Trường** | **Kiểu Dữ Liệu** | **Kích Thước** | **Giá Trị Mặc Định** | **Khóa Chính** | **Duy Nhất** | **Not Null** | **Khóa Ngoài** | **Giá Trị Mặc Định** |
| PDT\_MA | Integer |  |  |  | X | X | X | Mã phiếu đặt tiệc |
| HDT\_MA | Integer |  |  |  |  | X |  | Mã hóa đơn tiệc |
| HDT\_TONGTIEN | Integer |  |  |  |  |  |  | Tổng tiền |
| HDT\_GHICHU | Text |  |  |  |  |  |  | Ghi chú |

**2.2.10. Bảng:**

**PHIEUDATTIEC(PDT\_MA, KH\_MA, PDT\_NGAYTAO)**

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Tên Trường** | **Kiểu Dữ Liệu** | **Kích Thước** | **Giá Trị Mặc Định** | **Khóa Chính** | **Duy Nhất** | **Not Null** | **Khóa Ngoại** | **Giá Trị Mặc Định** |
| PDT\_MA | Integer |  |  | X | X | X |  | Mã phiếu đặt tiệc |
| B\_MA | Integer |  |  |  |  | X | X | Mã bàn |
| KH\_MA | Integer |  |  |  |  | X | X | Mã khách hàng |
| PDT\_NGAYTAO | DateTime |  | Current\_timestamp |  |  | X |  | Ngày tạo phiếu đặt tiệc |

**2.2.11 Bảng:**

**CHITIETPHIEUDAT(PDT\_MA, MA\_MA, CTPD\_SOLUONG)**

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Tên Trường** | **Kiểu Dữ Liệu** | **Kích Thước** | **Giá Trị Mặc Định** | **Khóa Chính** | **Duy Nhất** | **Not Null** | **Khóa Ngoại** | **Giá Trị Mặc Định** |
| PDT\_MA | Integer |  |  | X | X | X | X | Mã phiếu đặt tiệc |
| MA\_MA | Integer |  |  | X | X | X | X | Mã món ăn |
| CTPD\_SOLUONG | Int |  |  |  |  | X |  | Số lượng chi tiết phiếu đặt |

**2.2.12. Bảng:**

**PHIEUDATTIEC\_COMBO(CB\_MA, PDT\_MA, PDTCB\_SOLUONG, PDT\_CB\_DONGIA)**

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Tên Trường** | **Kiểu Dữ Liệu** | **Kích Thước** | **Giá Trị Mặc Định** | **Khóa Chính** | **Duy Nhất** | **Not Null** | **Khóa Ngoại** | **Giá Trị Mặc Định** |
| PDT\_MA | Integer |  |  | X | X | X | X | Mã phiếu đặt tiệc |
| CB\_MA | Integer |  |  | X | X | X | X | Mã combo |
| PDTCB\_SOLUONG | Int |  |  |  |  | X |  | Số lượng combo |

**2.2.13. Bảng:**

**HOADON(HD\_MA, B\_MA, KH\_MA, NV\_MA, HD\_TONGTIEN, HD\_THOIGIANLAP, HD\_TAOMOI, HD\_CAPNHAT, HD\_TRANGTHAI)**

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Tên Trường** | **Kiểu Dữ Liệu** | **Kích Thước** | **Giá Trị Mặc Định** | **Khóa Chính** | **Duy Nhất** | **Not Null** | **Khóa Ngoại** | **Diễn Giải** |
| HD\_MA | Integer |  |  | X | X | X |  | Mã hóa đơn |
| B\_MA | Integer |  |  |  |  | X | X | Mã bàn |
| KH\_MA | Integer |  |  |  |  | X | X | Mã khách hàng |
| NV\_MA | Integer |  |  |  |  | X | X | Mã nhân viên |
| HD\_TONGTIEN | Int | 20 |  |  |  |  |  | Tổng tiền hóa đơn |
| HD\_THOIGIANLAP | DateTime |  | Current\_timestamp |  |  |  |  | Thời gian lập hóa đơn |
| HD\_TAOMOI | DateTime |  | Current\_timestamp |  |  | X |  | Tạo mới hóa đơn |
| HD\_CAPNHAT | DateTime |  | Current\_timestamp |  |  | X |  | Cập nhật hóa đơn |
| HD\_TRANGTHAI | Tinyint |  | 2 |  |  | X |  | Trạng thái hóa đơn(1. Đã Thah Toán, 2. Chưa Thanh Toán) |

**2.2.14. Bảng:**

**HOADON\_COMBO(HD\_MA, CB\_MA, HDCB\_SOLUONG)**

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Tên Trường** | **Kiểu Dữ Liệu** | **Kích Thước** | **Giá Trị Mặc Định** | **Khóa Chính** | **Duy Nhất** | **Not Null** | **Khóa Ngoại** | **Giá Trị Mặc Định** |
| HD\_MA | Integer |  |  | X | X | X | X | Mã hóa đơn |
| CB\_MA | Integer |  |  | X | X | X | X | Mã combo |
| HDCB\_SOLUONG | Int |  |  |  |  | X |  | Số lượng combo |

**2.2.15. Bảng:**

**CHITIETHOADON(HD\_MA, MA\_MA,CTHD\_SOLUONG, CTHD\_DONGIA)**

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Tên Trường** | **Kiểu Dữ Liệu** | **Kích Thước** | **Giá Trị Mặc Định** | **Khóa Chính** | **Duy Nhất** | **Not Null** | **Khóa Ngoài** | **Giá Trị Mặc Định** |
| HD\_MA | Integer |  |  | X | X | X | X | Mã hóa đơn |
| MA\_MA | Integer |  |  | X | X | X | X | Mã món ăn |
| CTHD\_SOLUONG | Int | 10 |  |  |  | X |  | Số lượng món ăn |
| CTHD\_DONGIA | Int | 20 |  |  |  | X |  | Đơn giá món ăn |

**2.2.16. Bảng:**

**HINHANH(HA\_STT, MA\_MA**,**HA\_TEN)**

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Tên Trường** | **Kiểu Dữ Liệu** | **Kích Thước** | **Giá Trị Mặc Định** | **Khóa Chính** | **Duy Nhất** | **Not Null** | **Khóa Ngoại** | **Giá Trị Mặc Định** |
| HA\_STT | Integer |  |  | X | X | X |  | Số thứ tự |
| MA\_MA | Integer |  |  |  | X | X | X | Mã món ăn |
| HA\_TEN | Nvachar | 200 |  |  |  | X |  | Tên hình ảnh |

### Mô hình dữ liệu mức luận lý

Các ràng buộc tham chiếu:

MONAN(L\_MA) 🡺 LOAI(L\_MA)

HINHANH(MA\_MA) **🡺** MONAN(MA\_MA)

BAN(KV\_MA) 🡺 KHUVUC(KV\_MA)

HOADON(NV\_MA) **🡺** NHANVIEN(NV\_MA)

HOADON(B\_MA) **🡺** BAN(B\_MA)

CHITIETHOADON(MA\_MA) **🡺** MONAN(MA\_MA)

CHITIETHOADON(HD\_MA) **🡺** MONAN(HD\_MA)

HOADON\_COMBO(HD\_MA) **🡺** HOADON(HD\_MA)

HOADON\_COMBO(CB\_MA) **🡺** COMBO(CB\_MA)

CHITIETCOMBO(CB\_MA) **🡺** COMBO(CB\_MA)

CHITIETCOMBO(MA\_MA) **🡺** MONAN(MA\_MA)

PHIEUDATTIEC(KH\_MA) **🡺** KHACHHANG(KH\_MA)

PHIEUDATTIEC(B\_MA) **🡺** BAN(B\_MA)

PHIEUDATTIEC\_COMBO(CB\_MA) **🡺** COMBO(CB\_MA)

PHIEUDATTIEC\_COMBO(PDT\_MA) **🡺** PHIEUDATTIEC(PDT\_MA)

### Sơ đồ usecase:

* + 1. Mô hình Use case cho quản lý:



* + 1. Mô hình Use case cho nhân viên thu ngân:



* + 1. Mô hình Use case cho khách hàng:



## THIẾT KẾ THÀNH PHẦN XỬ LÝ

### Lưu đồ dòng dữ liệu

### Mô hình chức năng

Hệ thống quản lý nhà hàng

Quản lý danh mục

Cập nhật danh mục

Quản lý hóa đơn

Quản lý bình luận

Quản lý người dùng

Quản lý menu

Thống kê

Cập nhật hóa đơn

Xem danh mục

Thống kê hóa đơn

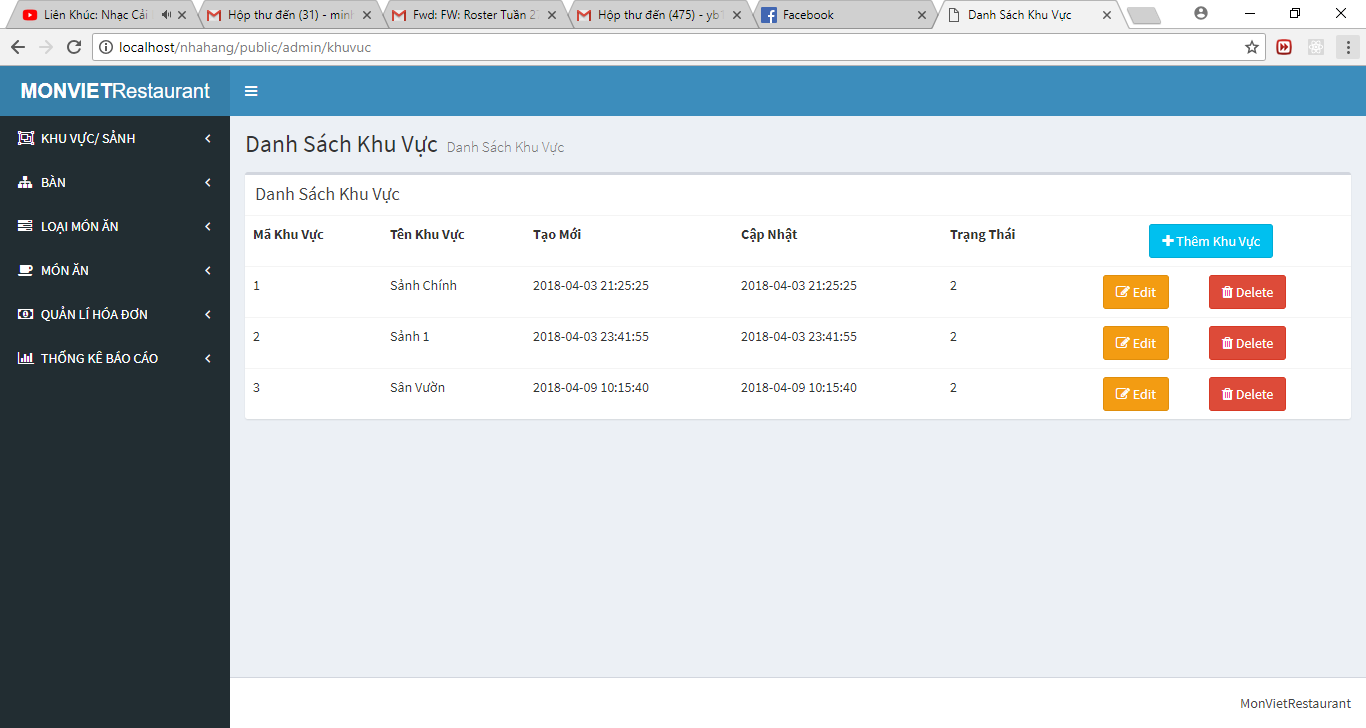
Cấp quyền

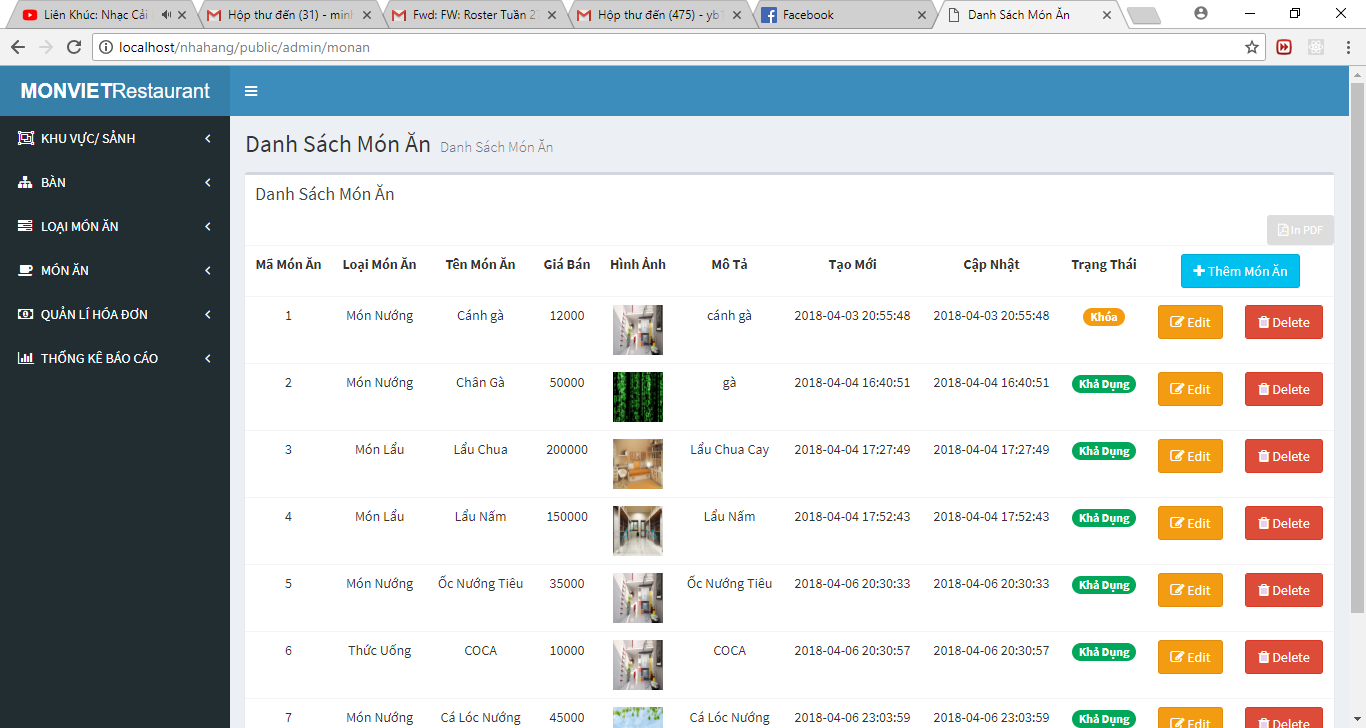
Cập nhật món ăn

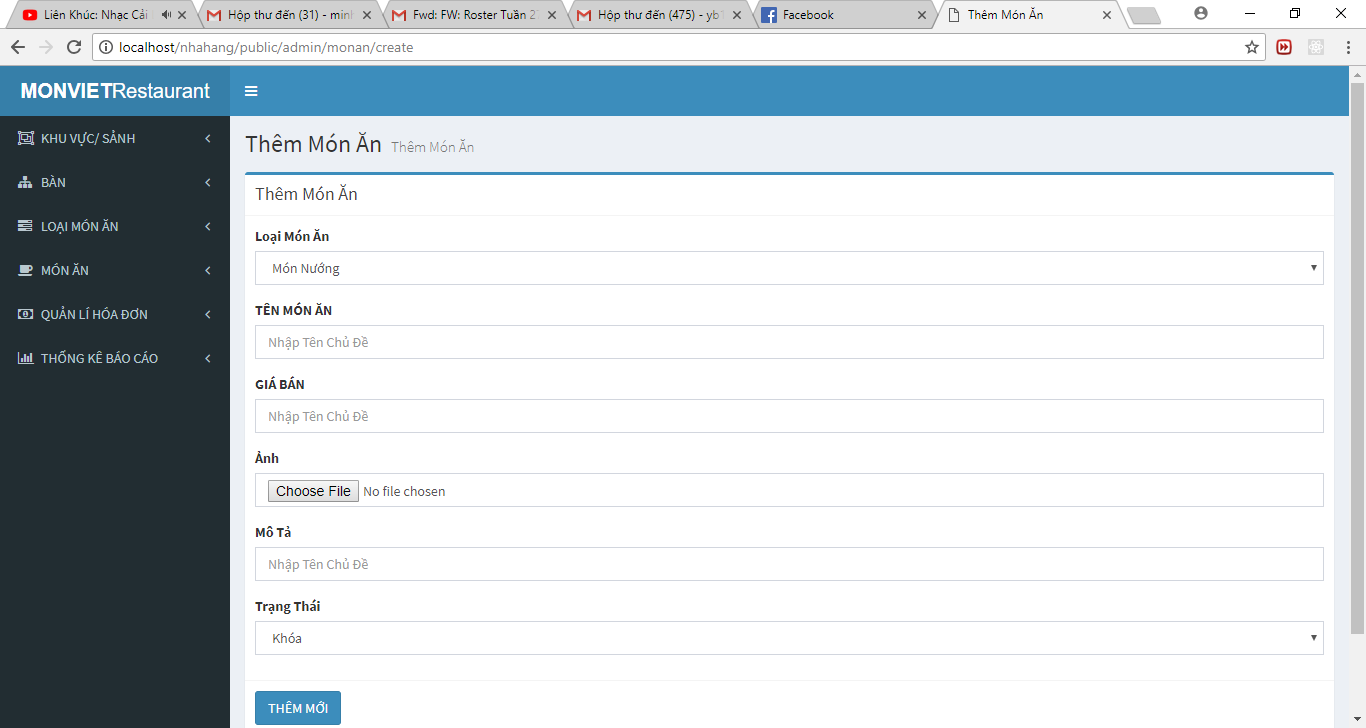
Xem món ăn

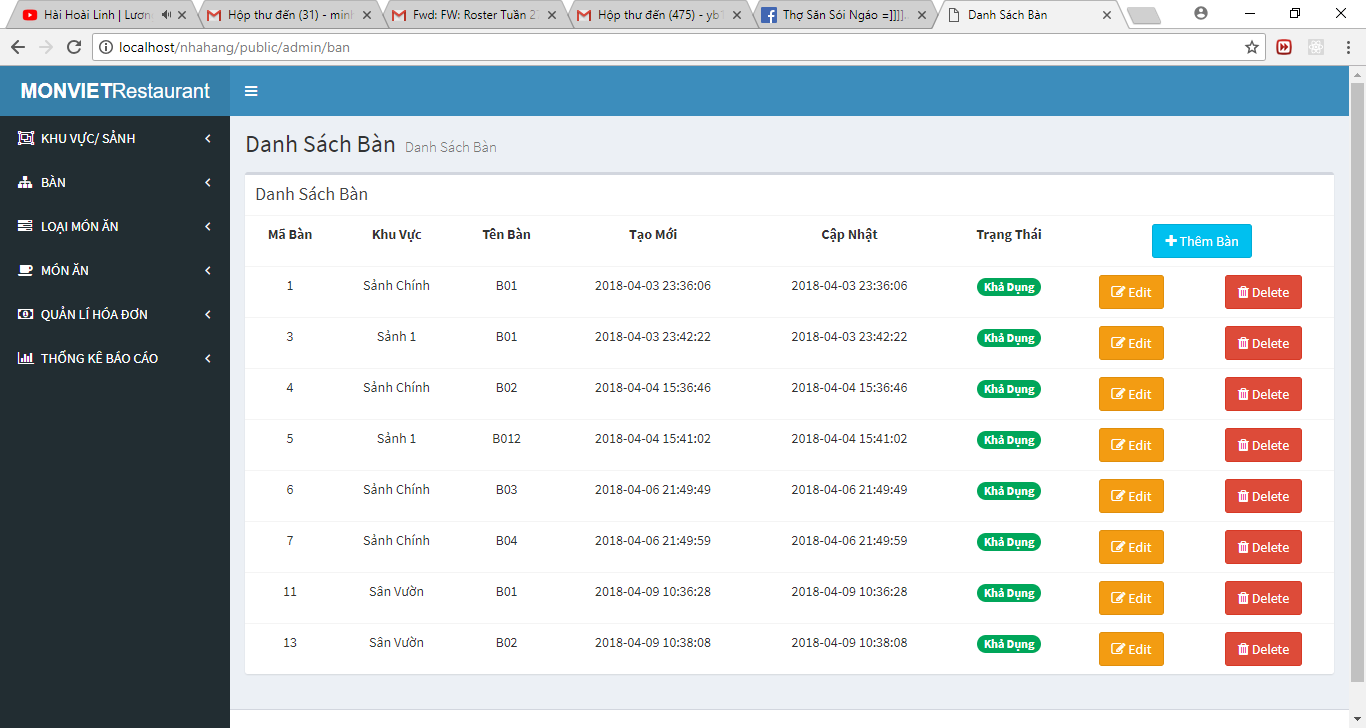
### Các lưu đồ - giải thuật

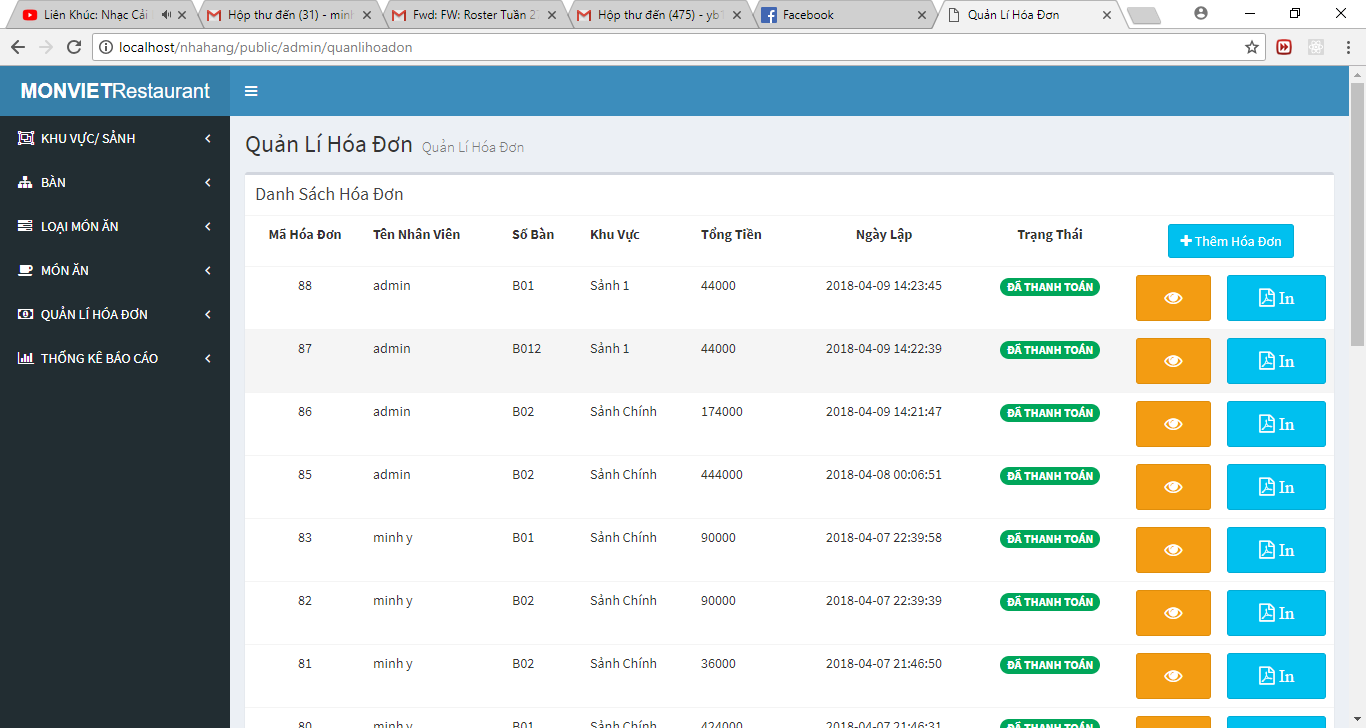
## THIẾT KẾ GIAO DIỆN

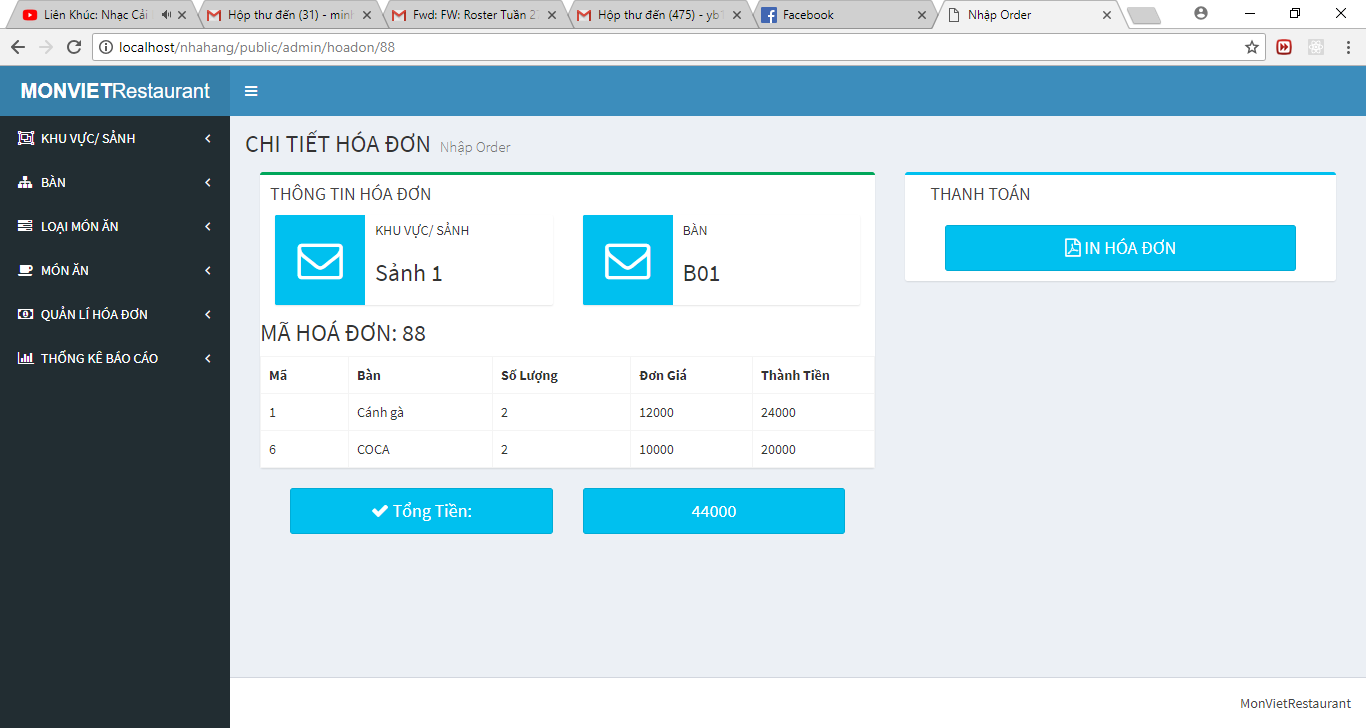




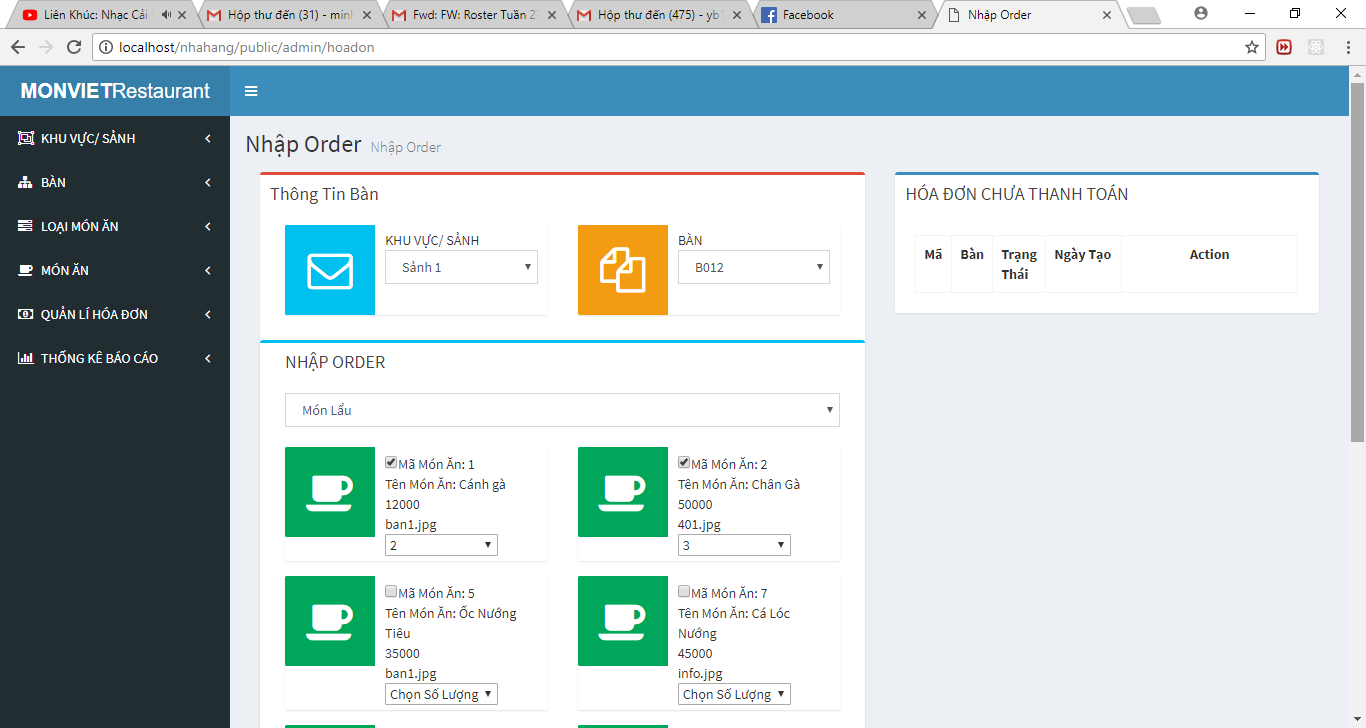


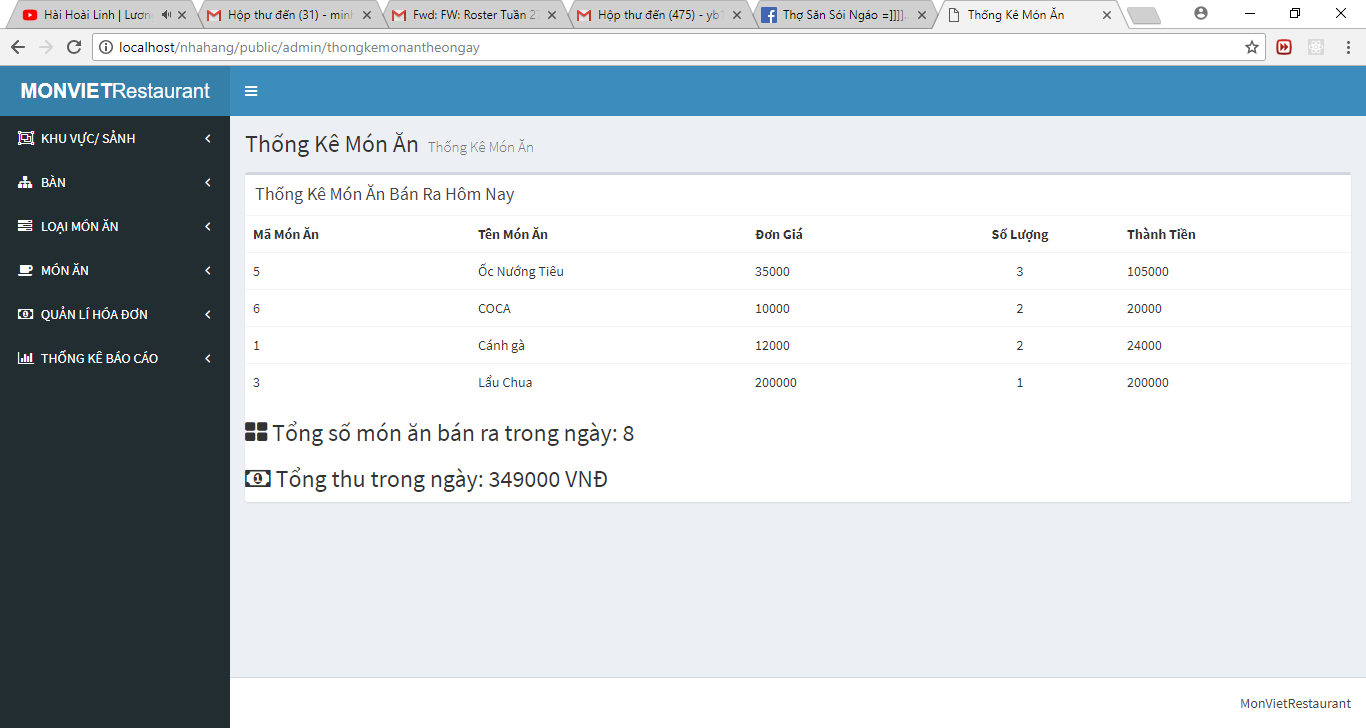


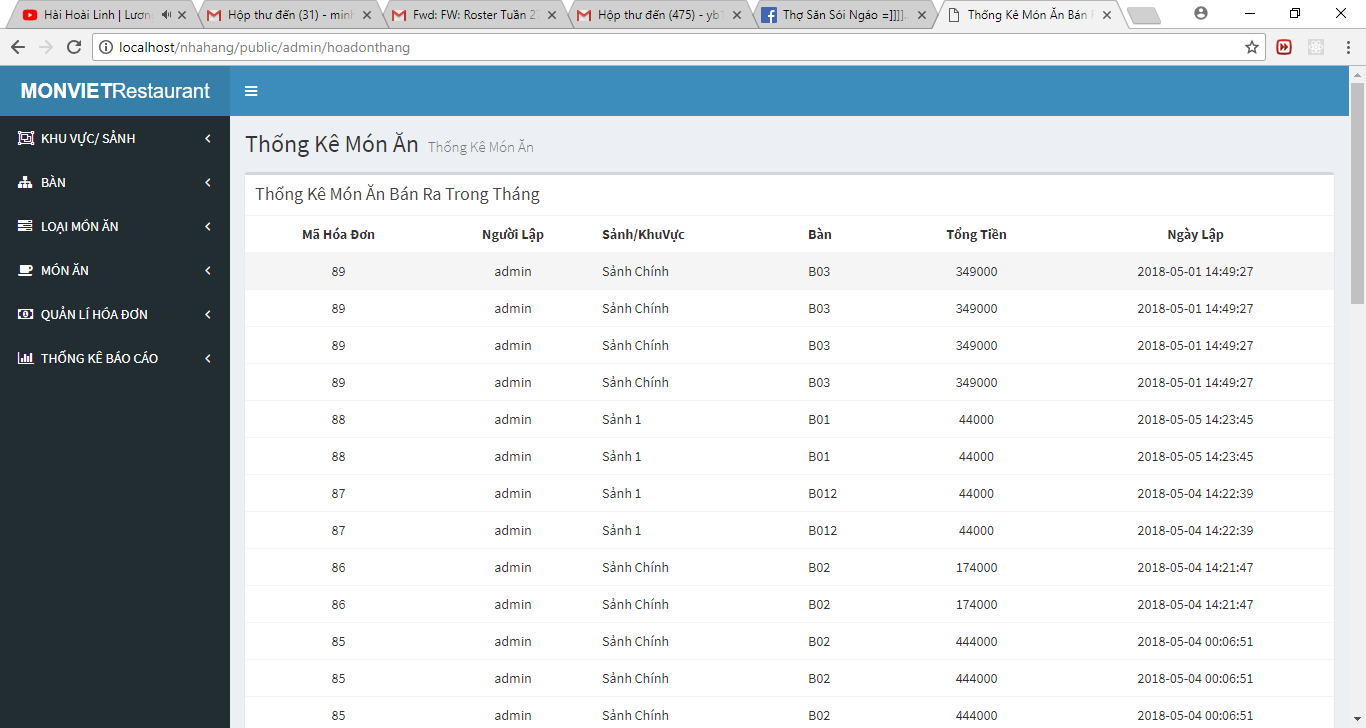


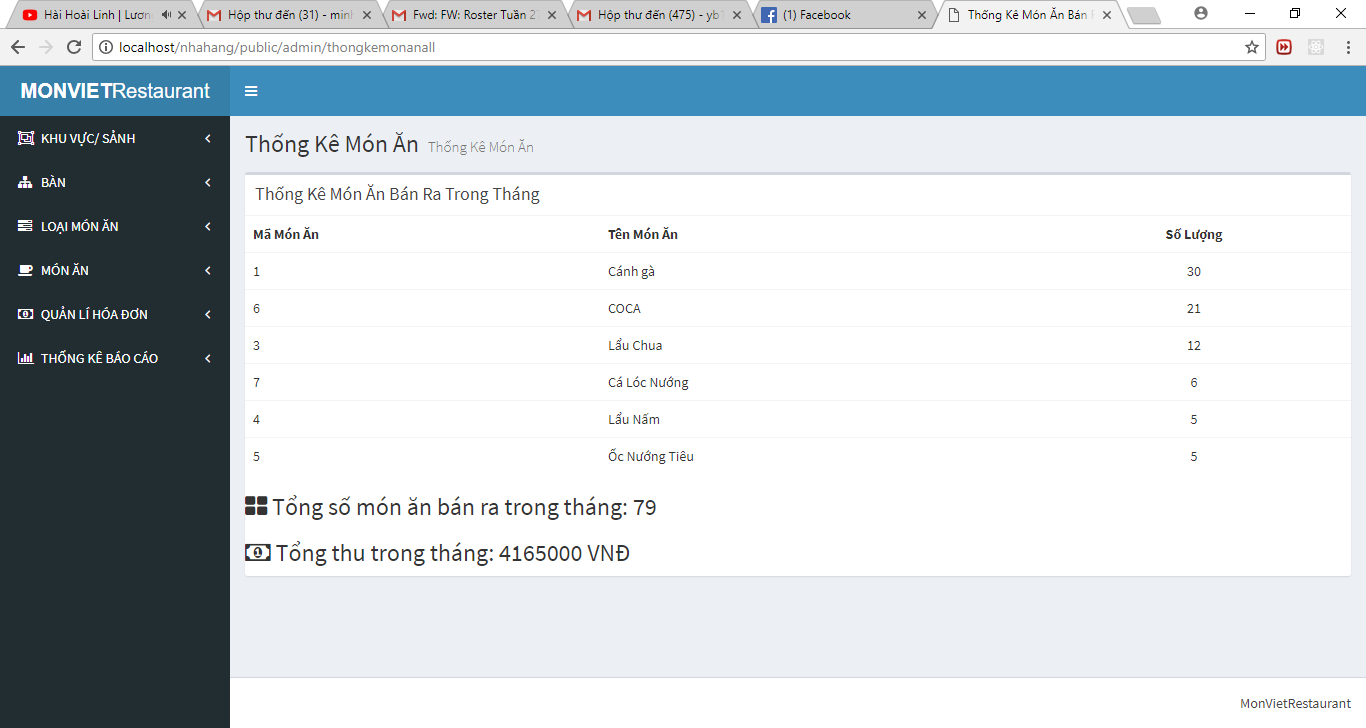












## THẢO LUẬN

# 

# CHƯƠNG 4: KẾT LUẬN VÀ HƯỚNG PHÁT TRIỂN

## KẾT LUẬN

* 1. **Kết quả đạt được:**

Hệ thống quản lý nhà trọ sinh viên tại thành phố Cần Thơ với các chức năng cơ bản đáp ứng hầu hết các yêu cầu đặt ra. Hệ thống phân quyền cho ba nhóm người dùng chính: chủ nhà trọ/ người cần cho thuê, sinh viên/ người cần thuê nhà, Admin.

* Chủ nhà trọ/ người cần cho thuê: có các chức năng cơ bản sau
* Đăng kí nhà trọ.
* Đăng tin cho thuê.
* Quản lý nhà trọ đã đăng kí.
* Quản lý bài đăng.
* Quản lý đặt phòng.
* Sinh viên/ người cần thuê nhà:
* Tìm kiếm nhà trọ theo nhiều tiêu chí.
* Bình luận nhà trọ.
* Xem hướng dẫ chỉ đường đến nhà trọ.
* Đặt xem phòng
* Admin:
* Quản lý danh mục: loại nhà trọ, loại bài đăng, tiện ích,…
* Quản lý trường đại học.
* Quản lý nhà trọ.
* Quản lý tài khoản
* Quản lý bài đăng.
* Quản lý bình luận.

Tính chính xác khá cao vì tận dụng tối đa công nghệ Google Map Api để định vị các vị trí của người dùng.

Tính khả dụng cao vì có thể chạy ổn định trên nhiều thiết bị.

Dễ dàng nâng cấp, có khả năng phát triển với mức độ cao hơn.

* 1. **Hạn chế:**

Hệ thống quản lý nhà hàng cơ bản đã đáp ứng được những chức năng cơ bản, song để hoàn thiện hơn hệ thống cần phát triển thêm một số chức năng.

* Chưa có chức năng đặt tiệc online.
* Chưa tối ưu trong việc nhập hóa đơn cho thu ngân.
* Giao diện chưa co dãn tốt theo nhiều kích thước màn hình.

## HƯỚNG PHÁT TRIỂN

* Triển khai hệ thống đến với các nhà hàng vừa và nhỏ.
* Tối ưu hóa cơ sở dữ liệu.
* Tối ưu hóa chức năng tìm kiếm, nhằm hỗ trợ tối đa cho nhân viên trong việc quản lý nhà hàng.
* Thiết kế giao diện chuyên nghiệp hơn.
* Cập nhật những công nghệ mới nhất để hệ thống chạy mượt mà hơn.
* Xây dựng app trên thiết bị di động, hỗ trợ order tại bàn.

# TÀI LIỆU THAM KHẢO

1. Giáo Trình Cơ Sở Dữ Liệu – Đại Học Cần Thơ – 2014.
2. Giáo trình phân tích thiết kế hệ thống thông tin
3. Website:

[www.laravel.com](http://www.laravel.com)

[www.viblo.asia](http://www.viblo.asia)

[www.stackoverflow.com](http://www.stackoverflow.com)