|  |  |
| --- | --- |
|  | BỘ GIÁO DỤC VÀ ĐÀO TẠO |

**ĐỒ ÁN MÔN HỌC: CÔNG NGHỆ PHẦN MỀM**

**XÂY DỰNG TRANG WEBSITE BÁN HOA TƯƠI**

Ngành: **CÔNG NGHỆ THÔNG TIN**

Lớp:

**Giảng viên hướng dẫn:**

Sinh viên thực hiện: Mã SV:

*Tp.HCM, ngày 9 tháng 1 năm 2024*

***Lời nói đầu***

*Hiện nay trên thế giới thương mại điện tử đang phát triển rất mạnh mẽ. Kỹ thuật số giúp chúng ta tiết kiệm đáng kể các chi phí nhờ chi phí vận chuyển trung gian, chi phí giao dịch và đặc biệt là giúp tiết kiệm thời gian để con người đầu tư vào các hoạt động khác. Hơn nữa thương mại điện tử còn giúp con người có thể tìm kiếm tự động theo nhiều mục đích khác nhau, tự động cung cấp thông tin theo nhu cầu và sở thích của con người. Giờ đây, con người có thể ngồi tại nhà để mua sắm mọi thứ theo ý muốn và các website bán hàng trên mạng sẽ giúp ta làm được điều đó. Chính vì vậy trong đồ án này em chọn đề tài về:* ***“Xây dựng Website bán hoa”****.*

**Danh sách Hình vẽ**

[Hình 2-1: Các thành phần chính của mô hình MVC 14](#_Toc63437)

[Hình 2-2: Kiến trúc của linq trong .NET Framework 3.5 20](#_Toc63438)

[Hình 2-3: Kiến trúc Dataset 21](#_Toc63439)

[Hình 2-4: Kiến trúc của Entity Framework 22](#_Toc63440)

[Hình 3-1: Admin,Owner Roles 14](#_Toc63437)

[Hình 3-2:Users Roles 20](#_Toc63438)

[Hình 3-3: Bảng User 21](#_Toc63439)

[Hình 3-4: Bảng Products 22](#_Toc63440)

[Hình 3-5: Bảng Orders 14](#_Toc63437)

[Hình 3-6: Bảng Order\_Items 20](#_Toc63438)

[Hình 3-7: Bảng Category\_Requests 21](#_Toc63439)

[Hình 3-8: Bảng Categories 22](#_Toc63440)

[Hình 3-9: Giao diện đăng nhập 14](#_Toc63437)

[Hình 3-10: Giao diện đăng nhập 20](#_Toc63438)

[Hình 3-11: Giao diện Đăng ký tài khoản 21](#_Toc63439)

[Hình 3-12: Giao diện Trang chủ 22](#_Toc63440)

[Hình 3-13: Giao diện Sản phẩm 14](#_Toc63437)

[Hình 3-14: Giao diện Mô tả 20](#_Toc63438)

[Hình 3-15: Giao diện Giỏ hàng 21](#_Toc63439)

[Hình 3-16: Giao diện Thông tin người dùng 22](#_Toc63440)

[Hình 3-17: Giao diện Đơn hàng đã đặt 14](#_Toc63437)

[Hình 3-18: Giao diện Thêm, sửa, xóa SP 20](#_Toc63438)

[Hình 3-19: Giao diện Quản lý người dùng 21](#_Toc63439)

[Hình 3-20: Giao diện Quản lý đơn hàng 22](#_Toc63440)

[Hình 3-21: Giao diện chỉnh sửa sản phầm 14](#_Toc63437)

[Hình 3-22: Giao diện Mua hàng thành công 20](#_Toc63438)

**Mục lục**

Lời mở đầu................................................................................................................... 2

Danh sách hình.............................................................................................................3

Danh sách các từ viết tắt...............................................................................................6

Chương 1:Tổng quan....................................................................................................7

1.1 Giới thiệu về đề tài.............................................................................................7

1.2 Lý do chọn đề tài................................................................................................7

1.3 Mục đích phát triển đề tài...................................................................................7

1.4 Ý nghĩa thực tiễn............................................................................................... 7

1.5 Đặt bài toán........................................................................................................7

1.5.1 Chức năng...................................................................................................7

1.5.2 Yêu cầu đặt ra.............................................................................................8

1.6 Các giai đoạn triển khai dự án...........................................................................9

1.7 Vấn đề - Giải pháp.............................................................................................9

Chương 2: Tìm hiểu 1 số công nghệ xây dựng website..............................................10

1.8 ASP.NET và MVC...........................................................................................10

1.8.1 ASP.NET...................................................................................................10

1.8.2 MVC..........................................................................................................11

1.9 Javascript..........................................................................................................14

1.10 Razor...............................................................................................................16

1.11 Bootstrap.........................................................................................................16

1.12 Hệ cơ sở dữ liệu SQL server..........................................................................16

1.13 Truy vấn Linq.................................................................................................17

1.13.1 Các thành phần của LINQ.......................................................................17

1.13.2 LINQ vs ADO.NET.................................................................................21

Chương 3: Phân tích thiết kế hệ thống........................................................................22

1.14 Tổng quan thiết kế.......................................................................................... 22

1.14.1 Diagram..............................................................................................22

1.14.2 Cơ sở dữ liệu.......................................................................................23

1.15Mô tả sử dụng các giao diện của web SunFlower.........................................25

Chương 4: Kết luận...................................................................................................36

Tài liệu tham khảo.....................................................................................................37

**Danh sách các từ viết tắt**

**#**

**Chữ viết tắt**

**Gải thích**

1

SRS

Software Requirement Specification

2

BR

Business Rule

3

SC

Screen

4

UC

Use Case

5

CRUD

Create/Read/Update/Delete a record in database

6

CSDL

Cơ sở dữ liệu

7

OOP

Object Oriented Programming

8

JS

J

avaScript

9

HTML

10

PHP

11

SEO

**Chương 1:Tổng quan**

**1.1 Giới thiệu về đề tài.**

Trong bối cảnh kinh kế thị trường hiện nay, việc mỗi công ty, cửa hàng hay thậm chí cá nhân có một hay nhiều website là một việc bình thường, xong việc xây dựng cho mình một trang web bán hoa là một điều còn khá mới mẻ với các cửa hàng báng hoa. Vấn đề còn bỏ ngỏ, nhận thấy cơ hội, khả năng phát triển, em nhận đề tài “Xây dưng website bán hoa” nhằm mục đích đưa các mồ hình bán hàng thông qua các website tới tay người dùng và các công ty, người kinh doanh shop hoa lớn nhỏ nói chung

**1.2 Lý do chọn đề tài**

Khi internet ra đời, không ai nghĩ rằng nó sẽ phát triển đến ngày hôm nay, nó len lỏi đến mọi ngó ngánh trong cuộc sống, trở thành 1 phần không thể thiếu. Chỉ cần ngồi nhà và cả thế giới trong tầm tay bạn. Cũng như vậy, khi internet đã trở nên phổ biến, việc mua bán qua mạng internet phát triển cũng phát triển theo đó. Xong bên cạnh đó, việc sở hữu một trang web đối với những cửa hàng nhỏ thì thực sự chưa dễ. Vì thế nhóm em quyết định tạo ra trang web này để cho người dùng và các doanh nghiệp dễ dàng tiếp cận hơn .

Bên cạnh đó, việc phát triển website đi kèm những tính năng giúp tối ưu hóa việc trải nghiệm website của người dùng cũng là cách giúp website phát triển 1 cách bền vững và có hiệu quả.

**1.3 Mục đích phát triển đề tài**

Đề tài được phát triển nhằm mục đích thương mại điện tử, quản lý việc mua bán thông qua mạng internet, nâng cao trải nghiệm sử dụng website của người dùng qua đó đưa thương hiệu đến rông rãi với người dùng hơn.

**1.4 Ý nghĩa thực tiễn**

Phát triển các sản phẩm công nghệ phù hợp với nhu cầu của xã hội, qua đó rèn luyện , trau đồi, phát triển kiến thức của bản thân nhằm đáp ứng nhu cầu của các nhà tuyển dụng sau khi ra trường, tạo ra sản phẩm thực tế có giá trị sử dụng trong cuộc sống.

**1.5 Đặt bài toán**

Bài toán đặt ra: Xây dựng website bán hoa gồm bao gồm các chức năng của 1 website bán hoa đơn thuần.

Qua quá trình khảo sát các nguồn thông tin từ internet, bạn bè, trải nghiệm thực tế trên một số website bán hoa có thương hiệu như <https://hoathuongyeu.com>, , <http://24hflowers.com>, em phân tích bài toán:

**1.5.1 Chức năng**

Đây là website bán hoa và giới thiệu sản phẩm của cửa hàng đến người tiêu dùng với các thông tin chi tiết mặt hàng, hình ảnh mô tả sản phẩm, giá cả, khuyến mãi, ưu đãi với người mua hàng. Website cần bao gồm các chức năng sau:

* Cho phép cập nhập hàng hóa vào CSDL
* Hiển thị danh sách các mặt hàng hóa
* Hiển thị các mặt hàng người dùng đã chọn sau quá trình tìm hiểu sản phẩm
* Hiển thị thông tin khách hàng
* Quản lý đơn đặt hàng
* Cập nhập hàng hóa, ...
* Xử lý đơn hàng
* Thống kê các khách hàng mua hàng trong khoảng thời gian nhất định

**1.5.2 Yêu cầu đặt ra**

Hệ thống có 2 phần:

a.Phần thứ nhất: Khách hàng

Khách hàng có nhu cầu mua hàng, họ sẽ tìm kiếm mặt hàng trên website, khi quyết định mua sẽ đặt hàng thông qua giao diện website. Đơn hàng sẽ được chuyển về cho cửa hàng, công ty. Vì vậy giao diện khách hàng cần có các chức năng sau:

Giao diện người dùng thân thiện, có thể thay đổi phù hợp với những ngày lễ, những đợt giảm giá khuyến mãi, ... nhằm đánh mạnh vào nhu cầu, tâm lý mua sắm của người dùng. Việc người dùng sử dụng website của bạn thêm một giây là gia tăng thêm tỷ lệ mua hàng của khách hàng.

Hiển thị các danh sách các mặt hàng để khách hàng xem, lựa chọn và mua

Sau khi khách hàng đã lựa chọn xong và muốn đặt hàng, cần có giao diện đặt hàng cho người dùng bao gồm các thông tin về hàng hóa đã chọn, số lượng, số tiền, tổng tiền đơn hàng, các thông tin cá nhân của khách hàng để phục vụ việc liên lạc lại với khách hàng.

b.Phần thứ hai: Giao diện quản trị.

Người làm chủ ứng dụng được cấp tài khoản riêng để có thể kiểm soát toàn các hoạt động của ứng dụng. Về phần này cần có các chức năng chính:

* Chức năng thêm, sửa, xóa các loại mặt hàng, nhà sản xuất, …
* Tiếp nhận đơn hàng của khách hàng và sửa lý đơn hàng
* Một số chức năng nâng cao theo yêu cầu như tạo giao diện menu động, tùy chỉnh theo ý người quản trị website, …

Giao diện quản trị nên làm đơn giản, tập chung vào các chức năng, đơn giản, dễ sử dụng

**1.6 Các giai đoạn triển khai dự án.**

* Giai đoạn 1:

Xây dựng 1 website đầy đủ các chức năng cơ bản, sử dụng cơ dở dữ liệu SQL và xây dựng website theo mô hình 3 lớp kết hợp MVC.

Giai đoạn 2:

Xây dựng 1 số tính năng mới để tăng trải nghiệm của người dùng với website, qua đó hỗ trợ người dùng 1 cách tối ưu.

**1.7 Vấn đề - Giải pháp**

* Khó khă

|  |
| --- |
| - |

Kiến thức thực tế còn hạn chế

|  |
| --- |
| - |

Công nghệ biến đổi theo từng đơn vị thời gian nên cần có chiến lược phát triền, thời gian và công sức tìm hiểu.

* Giải pháp:

|  |
| --- |
| - |

Tự tìm dành thời gian tìm hiểu những công nghệ mới, giao lưu, học hỏi từ những người có kinh nghiệm trong ngành công nghệ thông tin và giáo viên hướng dẫn

**Chương 2: Tìm hiểu 1 số công nghệ xây dựng website**

Việc xây dựng và phát triển website hiện nay có rất nhiều ngôn ngữ như sử dụng javascript có MeotorJS, Angularjs, C# , Java, PHP ... Trong quá trình tìm hiểu và xin ý kiến của những người đi trước và kiến thức bản thân trong quá trình rèn luyện học tập, em phát triển website theo ASP.NET sử dụng mô hình MVC. Về thiết kế giao diện em sử dụng framework Bootstrap 5 giúp giảm thời gian phát triển giao diện và kế thừa những ưu điểm của bootstrap. Bên cạnh đó sử dụng javascript, jquery, razor … nhằm tối ưu hoá sử lý website, sử dụng SQL làm cơ sở dữ liệu, enity framework để kết nối với cơ sở dữ liệu, sử dụng linq thay câu truy vấn SQL.

Chương này bao gồm lý thuyết về các công nghệ làm website được áp dụng.

**1.8 ASP.NET và MVC**

**1.8.1 ASP.NET**

ASP.NET là Active Server Pages .NET (.NET là .NET framework). Nói đơn giản và ngắn gọn thì ASP.NET là một công nghệ có tính cách mạng dùng để phát triển các ứng dụng về mạng hiện nay cũng như trong tương lai (ASP.NET is a revolutionary technology for developing web applications). ASP.NET là một phương pháp tổ chức hay khung tổ chức (framework) để thiết lập các ứng dụng hết sức hùng mạnh cho mạng dựa trên CLR (Common Language Runtime) chứ không phải là một ngôn ngữ lập trình.

ASP.NET là một cơ cấu trong các cơ cấu của hệ điều hành Windows dưới dạng nền hay khung .NET. Như vậy ASP.NET không những có thể dùng các object của các ứng dụng cũ mà còn có thể sử dụng tất cả mọi tài nguyên Windows có một cách dễ dàng.

Tập tin của ASP.NET (ASP.NET file) có extension là .ASPX, còn tập tin của ASP là .ASP

Tập tin của ASP.NET (ASP.NET file) được phân tích ngữ pháp bởi XSPISAPI.DLL. Còn tập tin của ASP được phân tích bởi i ASP.DLL.

ASP.NET là kiểu mẫu lập trình phát động bằng sự kiện (event driven), còn các trang ASP được thi hành theo tuần tự từ trên xuống dưới.

ASP.NET sử dụng trình biên dịch (compiled code) nên rất nhanh, còn ASP dùng trình thông dịch (interpreted code) do đó hiệu suất và tốc độ cũng thua kém hẳn.

ASP.NET hỗ trợ gần 25 ngôn ngữ lập trình mới với .NET và chạy trong môi trường biên dịch (compiled environment), còn ASP chỉ chấp nhận VBScript và JavaScript nên ASP chỉ là một scripted language trong môi trường thông dịch (in the interpreter environment). Không những vậy, ASP.NET còn kết hợp nhuần nhuyễn với XML (Extensible Markup Language) để vận chuyển các thông tin qua mạng.

ASP.NET hỗ trợ tất cả các browser và quan trọng hơn nữa là hỗ trợ các thiết bị lưu động (mobile devices). Chính các thiết bị lưu động, mà mỗi ngày càng phổ biến, đã khiến việc dùng ASP trong phát triển mạng nhằm vươn tới thị trường mới đó trở nên vô cùng khó khăn.

**1.8.2 MVC**

MVC là viết tắt của Model – View – Controller. Là một trong những design pattern. Được vận hành để tách mã lệnh thành 3 phần riêng biệt. Ở mỗi phần MVC sẽ có những chức năng đặc thù. Để xử lý các tác vụ mà request gởi tới. MVC làm cho mã lệnh trở nên trong sáng, dễ phát triển và dễ nâng cấp theo thời gian.

Để làm việc tốt đối với MVC, chúng ta cần nắm thật vững kiến thức OOP. Bản chất của các framework khác cũng được hình thành trên lý thuyết MVC. Do vậy, nếu chúng ta nắm tốt MVC, thì ở những framework khác chắc chắn sẽ không cảm thấy khó hiểu.

* Model: Là thành phần chịu trách nhiệm xử lý các thao tác trên database. Và gửi trả kết quả thông qua view.
* View: Là phần hiển thị thông tin trên website, sau khi đi qua controller và nhận kết quả từ phía model thì view là bước cuối cùng để chuyển thông tin tới người dùng.
* Controller: Là phần điều hướng các request tới những tác vụ tương ứng. Controller là một phần không thể thiếu ở bất cứ framework nào. Vì nó có trách nhiệm gởi và nhận request từ hệ thống tới người sử dụng.

**Xuất xứ**

Tất cả bắt đầu vào những năm 70 của thế kỷ 20, tại phòng thí nghiệm Xerox

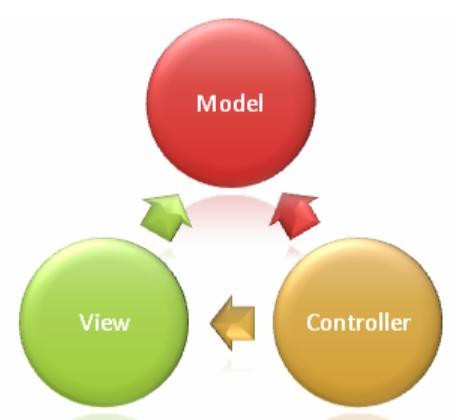
PARC ở Palo Alto. Sự ra đời của giao diện đồ họa (Graphical User Interface) và lập trình hướng đối tượng (Object Oriented Programming) cho phép lập trình viên làm việc với những thành phần đồ họa như những đối tượng đồ họa có thuộc tính và phương thức riêng của nó. Không dừng lại ở đó, những nhà nghiên cứu ở Xerox PARC còn đi xa hơn khi cho ra đời cái gọi là kiến trúc MVC (viết tắt của Model – View – Controller).

MVC được phát minh tại Xerox Parc vào những năm 70, bởi

TrygveReenskaug. MVC lần đầu tiên xuất hiện công khai là trong Smalltalk-80. Sau đó trong một thời gian dài hầu như không có thông tin nào về MVC, ngay cả trong tài liệu 80 Smalltalk. Các giấy tờ quan trọng đầu tiên đƣợc công bố trên MVC là “A Cookbook for Using the Model-View-Controller User Interface Paradigm in Smalltalk – 80”, bởi Glenn Krasner và Stephen Pope, xuất bản trong tháng 8 / tháng 9 năm 1988.

**Kiến trúc mô hình MVC**

Trong kiến trúc MVC, một đối tượng đồ họa người dùng (GUI Component) bao gồm 3 thành phần cơ bản: Model, View, và Controller. Model có trách nhiệm đối với toàn bộ dữ liệu cũng như trạng thái của đối tượng đồ họa. View chính là thể hiện trực quan của Model, hay nói cách khác chính là giao diện của đối tượng đồ họa. Và Controller điều khiển việc tương tác giữa đối tượng đồ họa với người sử dụng cũng như những đối tượng khác.



# Hình 2-1:Các thành phần chính của mô hình MVC

Khi người sử dụng hoặc những đối tượng khác cần thay đổi trạng thái của đối tượng đồ họa, nó sẽ tương tác thông qua Controller của đối tượng đồ họa. Controller sẽ thực hiện việc thay đổi trên Model. Khi có bất kỳ sự thay đổi nào xảy ra ở Model, nó sẽ phát thông điệp (broadcast message) thông báo cho View và Controller biết. Nhận được thông điệp từ Model, View sẽ cập nhật lại thể hiện của mình, đảm bảo rằng nó luôn là thể hiện trực quan chính xác của Model. Còn Controller, khi nhận được thông điệp từ Model, sẽ có những tương tác cần thiết phản hồi lại người sử dụng hoặc các đối tượng khác

**Đặc điểm của mô hình MVC**

Cái lợi ích quan trọng nhất của mô hình MVC là nó giúp cho ứng dụng dễ bảo trì, module hóa các chức năng, và được xây dựng nhanh chóng. MVC tách các tác vụ của ứng dụng thành các phần riêng lẻ model, view, controller giúp cho việc xây dựng ứng dụng nhẹ nhàng hơn. Dễ dàng thêm các tính năng mới, và các tính năng cũ có thể dễ dàng thay đổi. MVC cho phép các nhà phát triển và các nhà thiết kế có thể làm việc đồng thời với nhau. MVC cho phép thay đổi trong 1 phần của ứng dụng mà không ảnh hưởng đến các phần khác

Sở dĩ như vậy vì kiến trúc MVC đã tách biệt (decoupling) sự phụ thuộc giữa các thành phần trong một đối tượng đồ họa, làm tăng tính linh động (flexibility) và tính tái sử dụng (reusebility) của đối tượng đồ họa đó. Một đối tượng đồ họa bấy giờ có thể dễ dàng thay đổi giao diện bằng cách thay đổi thành phần View của nó trong khi cách thức lưu trữ (Model) cũng như xử lý (Controller) không hề thay đổi. Tương tự, ta có thể thay đổi cách thức lưu trữ (Model) hoặc xử lý (Controller) của đối tượng đồ họa mà những thành phần còn lại vẫn giữ nguyên.

Chính vì vậy mà kiến trúc MVC đã được ứng dụng để xây dựng rất nhiều framework và thư viện đồ họa khác nhau. Tiêu biểu là bộ thư viện đồ họa của ngôn ngữ lập trình hướng đối tượng SmallTalk (cũng do Xerox PARC nghiên cứu và phát triển vào thập niên 70 của thế kỷ 20). Các Swing Components của Java cũng được xây dựng dựa trên kiến trúc MVC. Đặc biệt là nền tảng ASP.NET MVC Frame work.

**Tiềm hiểu về Controler**

Controller có trách nhiệm chính là điều hướng các yêu cầu của người sử dụng. Như vậy trên toàn ứng dụng của ta, tất cả các request đều sẽ phải đi tới controller. Và tại đây, ứng với các tham số người sử dụng truyền mà ta đưa họ đến một tác vụ nào đó trên ứng dụng.

Tại các tác vụ này, chúng sẽ thông qua lớp model để làm việc và trả kết quả trở về controller. Cuối cùng controller sẽ đẩy dữ liệu thao tác tới view. View là thành phần cuối cùng mà người sử dụng nhận được khi họ giở request tới ứng dụng. Có thể hiểu controller, giống với kỹ thuật đa cấp với các tác vụ chạy ứng dụng phân cấp theo từng nhánh riêng biệt như: Module, action,….

**Tìm hiểu Model**

Model là thành phần chủ yếu được sử dụng để thao tác xử lý dữ liệu. Trong các framework, Model vẫn thường sử dụng theo phương thức Active Record. Một trong những design pattern. Chúng có tác dụng rút ngắn thời gian viết câu truy vấn cho người sử dụng. Biến những câu truy vấn phức tạp trở nên gần gũi và thân thiện với người sử dụng thông qua các thư viện được định nghĩa sẵn.

Model thường sẽ là các phương thức có trách nhiệm xử lý các tác vụ như:

select, insert, update, delete các record trong database. Ứng với các lấy dữ liệu, model thường sử dụng mảng để gởi trả kết quả về. Vì mảng có thể cho phép model lưu trữ nhiều thông tin hơn, nên thường các record khi bóc tách chúng sẽ mang các dữ liệu của database một cách chi tiết.

Khi sử dụng models, ta cũng cần tuân theo nguyên tắc chính của chúng là không xuất giá trị trực tiếp trong model. Mà tất cả những dữ liệu ấy, phải đưa vào mảng và trả về theo phương thức. Và tiếp tục ở view ta sẽ sử dụng nó để lấy dữ liệu ra.

**Tìm hiều về View**

View là phần hiển thị thông tin tương phản khi gởi và nhận request. Trước đây, khi người lập trình chưa nghĩ tới view. Họ thường thao tác xử lý dữ liệu ngay trực tiếp trên ứng dụng và đổ cả dữ liệu ngay trên file PHP đó. Điều này làm cho ứng dụng trở nên cồng kềnh, và đặc biệt rất khó cho việc bảo trì nâng cấp sau này. Nhất là đối với designer, việc thay đổi giao diện của một website luôn làm cho họ cảm thấy đau đầu vì phải chỉnh sửa thẳng vào core.

Trước đây, để giải quyết tình huống này. Người ta thường sử dụng template để phân tách website thành 2 mảng riêng biệt. Một là giao diện và một là core. Việc chỉnh sửa giao diện trở nên đơn giản hơn đối với họ so với cách viết thập cẩm kia. Tuy nhiên, các thư viện này thực chất sẽ làm cho ứng dụng của chúng ta trở nên chậm chạp hơn bao giờ hết, bởi chúng phải phiên dịch nhiều lần các kịch bản.

 ***Ưu Điểm:***

|  |  |
| --- | --- |
| - | MVC làm cho ứng dụng trở nên trong sáng, giúp lập trình viên phân tách |

ứng dụng thành ba lớp một cách rõ ràng. Điều này sẽ rất giúp ích cho việc phát triển những ứng dụng xét về mặt lâu dài cho việc bảo trì và nâng cấp hệ thống.

|  |  |
| --- | --- |
| - | MVC hiện đang là mô hình lập trình tiên tiến bậc nhất hiện nay, điều mà |

các framework vẫn đang nổ lực để hướng tới sự đơn giản và yếu tố lâu dài cho người sử dụng.

 ***Khuyết Điểm:***

|  |  |
| --- | --- |
| - | Mặc dù, MVC tỏ ra lợi thế hơn nhiều so với cách lập trình thông thường. |

Nhưng MVC luôn phải nạp, load những thư viện đồ sộ để xử lý dữ liệu. Chính điều này làm cho mô hình trở nên chậm chạp hơn nhiều so với việc code tay thuần túy.

|  |  |
| --- | --- |
| - | MVC đòi hỏi người tiếp cận phải biết qua OOP, có kinh nghiệm tương |

đối cho việc thiết lập và xây dựng một ứng dụng hoàn chỉnh. Sẽ rất khó khăn nếu OOP của người sử dụng còn yếu.

|  |  |
| --- | --- |
| - | MVC tận dụng mảng là thành phần chính cho việc truy xuất dữ liệu. |

Nhất là với việc sử dụng active record để viết ứng dụng. Chúng luôn cần người viết phải nắm vứng mô hình mảng đa chiều.

**1.9 Javascript**

JavaScript được phát minh bởi Brendan Eich của Netscape (với trình duyệt Navigator 2.0) vào năm 1995, và xuất hiện ở hầu hết các công cụ trình duyệt của Netscape và Microsoft kể từ năm 1996.

Ngày nay, gần như các trình duyệt đình đám đều có hỗ trợ Javascript, và chính Javascript là nhân tố gần đây đã châm ngòi cho cuộc chiến các trình duyệt.

Javascript đã ra đời từ khá lâu, nhưng cho đến ngày nay, nó vẫn mang 1 tầm quan trọng rất lớn, vì vậy học Javascript là điều cần thiết.

**Javascript là gì?**

Về phương diện thực thi ngôn ngữ

Hiện nay khi phát triển các trang web động, có hai phương thức rất phổ biến đó là *Server-Side* và *Client-Side*.

***Client-Side***: là phương thức sẽ xử lý các mã lệnh bởi mỗi browser riêng biệt tại máy người dùng, sau đó gửi kết quả lên cho Server.

***Server-Side***: thì lại gửi các mã lệnh cho server xử lý trước, sau đó server gửi kết quả lại cho browser.

JavaScript là một ngôn ngữ lập trình kịch bản Client-Side (Client-Side Script

Language), và dĩ nhiên nó được thực thi tại máy người dùng. Và vì javascript là một Client-Side Script Language nên nó không có khả năng kết nối và thao tác với CSDL trên Server.

Về phương diện phong cách ngôn ngữ lập trình

Các tài liệu cũ trước đây nói Javascript là ngôn ngữ lập trình dựa vào đối tượng (*Object-based language*). Nhưng từ lúc Ajax ra đời, Javascript trở nên hướng đối tượng hơn rất nhiều. Vì vậy, chúng ta có thể coi Javascript là một ngôn ngữ lập trình hướng đối tượng (*Object-Oriented programming*). Về cú pháp, Javascript cũng tương tự như C, Perl và Java, …ví dụ mệnh đề lặp if, while, for, tương tự nhau.

Nói tóm lại:

**Javascript là một ngôn ngữ lập trình**:

* Kịch bản Client-Side (Client-Side Script Language).  Hướng đối tượng (*Object-Oriented programming*).

**Javascript được sử dụng để làm gì?**

* JavaScript được sử dụng nhằm bổ sung sự tương tác cho các trang HTML.
* JavaScript có thể đáp ứng các sự kiện như tải hay loại bỏ các form. Khả năng này cho phép JavaScript trở thành một ngôn ngữ script động.
* JavaScript có thể được sử dụng để xác nhận dữ liệu người dùng nhập vào trước khi nó được chuyển đến server.
* Sử dụng Javascript có thể giúp website của bạn tương tác với người dùng một cách uyển chuyển hơn.  Tùy biến trình duyệt.

***\*Ưu điểm của Javascript:***

* Hoàn toàn miễn phí và dễ học.
* JavaScript được thiết kế độc lập với hệ điều hành. Nó có thể chạy trên bất kỳ hệ điều hành nào có trình duyệt hỗ trợ JavaScript.
* Dễ dàng tương tác, điều khiển và tránh bớt việc xử lý từ phía server
* Nắm vững kiến thức JavaScript bây giờ rất hữu dụng cho các bạn sau này để có thê tiếp thu những công nghệ mới mà nó được gói gọn vào những ngôn ngữ như : Ajax, Atlas …. ***\*Nhược điểm của Javascript***
* Javascript không có trình biên dịch riêng mà được diễn dịch và chạy bởi trình duyệt hỗ trợ nó. Chính vì thế, nếu trình duyệt không hỗ trợ, hoặc không bật JS, nó sẽ không chạy được.
* JS có thể làm ứng dụng web của bạn trở nên nặng nề hơn.  Bảo mật kém. Không có khả năng giấu mã. **Nguyên tắc lập trình trong JavaScript**
* Bất cứ ngôn ngữ lập trình nào cũng đều phải có nguyên tắc của nó, và Javascript cũng ko ngoại lệ. Để bắt đầu học lập trình 1 ngôn ngữ nào, trước tiên chúng ta cần phải biết quy tắc của nó.
* Một số quy tắc quan trọng trong JS mà chúng ta cần phải nhớ như sau:
  + - Lệnh Javascript phân biệt chữ in hoa và chữ thường - Mội câu lệnh Javascript đều kết thúc bằng dấu chấm phẩy “;”
    - Các điều kiện phải được khai báo trong cặp dấu ngoặc đơn ().
    - Khi sử dụng lệnh điều khiển, nếu sử dụng nhiều hơn 1 lệnh, bạn phải sử dụng cặp dấu ngoặc nhọn {}
    - Javascript sử dụng dấu chấm “.” để tham chiếu đến 1 phương thức hay thuộc tính của đối tượng
  1. **Razor**

Razor là một ngôn ngữ template engine được sử dụng trong các dự án của Microsoft, đặc biệt là trong framework ASP.NET và ASP.NET MVC. Ngôn ngữ này giúp kết hợp mã C# (hoặc VB.NET) với HTML để tạo ra các trang web động một cách dễ đọc và dễ bảo trì. Razor cung cấp cú pháp ngắn gọn và linh hoạt cho việc nhúng mã code C# vào trong các file HTML, mà không làm mất đi sự rõ ràng của cấu trúc HTML. Ví dụ, bạn có thể sử dụng cú pháp như @if, @foreach, @model, và nhiều lệnh khác để tích hợp logic C# vào trong mã HTML một cách dễ hiểu.

* 1. **Bootstrap**

Bootstrap là một **Font-end Framework** được viết bằng [SASS](http://freetuts.net/tag/sass-css) và biên dịch thành CSS, nó là một bộ thư viện mạnh mẽ được tích hợp nhiều CSS có sẵn giúp cho việc lập trình HTML & CSS trở nên đơn giản và dễ dàng hơn. Thông thường khi chúng ta lập trình bằng CSS thuần thì công việc kiểm tra tính tương thích trên các trình duyệt khác nhau và trên các thiết bị di động khác nhau rất là khó khăn, nhưng với Bootstrap thì lập trình viên không cần phải tốn nhiều công sức nữa vì mọi thứ đã có người tạo nên Bootstrap lo.

Bootstrap chia layout của một trang web ra thành 960 Grid và gồm 12 cột, mỗi cột 80Grid và đây chính lả chuẩn thiết kế HTML & CSS của Bootstrap. Nó có hỗ trợ hầu hết các module của một trang web như menu, tabs, tooltip, popup, ... Ngoài ra nó còn sử dụng thêm Javascript để xử lý các hiệu ứng cấp cao, [javascript](http://freetuts.net/tag/javascript-toan-tap) của Bootstrap sử dụng [jquery](http://freetuts.net/tag/jquery-toan-tap) nên để sử dụng được bắt buộc bạn phải bổ sung thêm thư viện jQuery nữa.

Các lợi ích khi sử dung bootstrap

* Phát triển giao diện nhanh chóng.
* Dễ học, dễ sử dụng
* Hỗ trợ javascript ,jquery
  1. **Hệ cơ sở dữ liệu SQL server**

SQL là một công cụ quản lý dữ liệu được sử dụng phổ biến ở nhiều lĩnh vực. Đầu tiên SQL được sử dụng trong các ngôn ngữ quản lý CSDL và chạy trên máy đơn lẻ. Do sự phát triển nhanh chóng của nhu cầu xây dựng những CSDL lớn theo mô hình khách/chủ (Client/Server), nhiều phần mềm sử dụng ngôn ngữ SQL đã ra đời mà điển hình là MS SQL Server, Oracle, Sybase... Trong mô hình khách/chủ, toàn bộ CSDL được tập trung lưu trữ trên máy chủ (Server), mọi thao tác xử lý dữ liệu được thực hiện trên máy chủ bằng các lệnh SQL. Máy trạm (Client) chỉ dùng để cập nhật dữ liệu cho máy chủ hoặc lấy thông tin từ máy chủ. Trên lĩnh vực đang phát triển hiện nay là Internet, ngôn ngữ SQL lại càng đóng vai trò quan trọng hơn. Nó được sử dụng để nhanh chóng tạo ra các trang Web động (Dynamic Web Page). Trang Web động thường có nội dung được lấy ra từ CSDL. SQL có thể được sử dụng như một chất keo kết dính giữa CSDLvà trang Web. Khi người dùng yêu cầu, SQL sẽ thực hiện việc truy cập thông tin trong CSDL trên máy chủ và hiển thị kết quả trên trang Web. Và SQL cũng là công cụ để cập nhật thông tin cho CSDLđó.

* 1. **Truy vấn Linq**

LINQ là gì ?

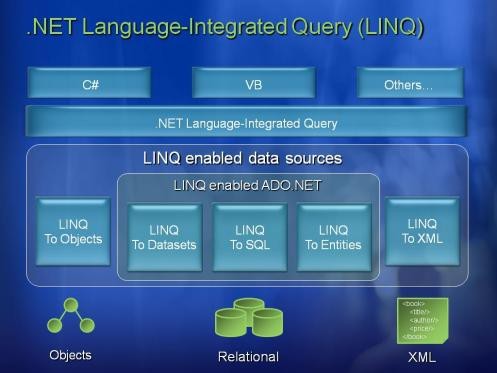
Để giảm gánh nặng thao tác trên nhiều ngôn ngữ khác nhau và cải thiện năng suất lập trình, Microsoft đã phát triển giải pháp tích hợp dữ liệu cho .NET Framework có tên gọi là LINQ (Language Integrated Query), đây là thư viện mở rộng cho các ngôn ngữ lập trình C# và Visual Basic.NET (có thể mở rộng cho các ngôn ngữ khác) cung cấp khả năng truy vấn trực tiếp dữ liệu Object, cơ sở dữ liệu và XML.

Điểm mạnh của LINQ là “viết truy vấn cho rất nhiều các đối tượng dữ liệu”. Từ cơ sở dữ liệu, XML, Data Object … thậm chí là viết truy vấn cho một biến mảng đã tạo ra trước đó. Vì thế ta có các khái niệm như là LinQ to SQL, LinQ to

XML,….

**1.13.1**

**Các thành phần của LINQ**



# Hình 2-2:Kiến trúc của linq trong .NET Framework 3.5

*1.13.1.1LINQ to Objects*

“LINQ to Objects” ở đây có nghĩa là nói đến cách sử dụn LINQ đối với các đối tượng Collection mà đã được thực thi giao diện IEnumerable hoặc

IEnumerable<T> tức những collection có thể “liệt kê” ra được. Đây là trường hợp sử dụng đơn giản nhất của LINQ khi làm việc với dữ liệu.

*1.13.1.2LINQ to SQL*

LINQ to SQL là một phiên bản hiện thực hóa của O/RM (object relational mapping) có bên trong .NET Framework 3.5, nó cho phép bạn mô hình hóa một cơ sở dữ liệu dùng các lớp .NET. Sau đó bạn có thể truy vấn cơ sở dữ liệu dùng LINQ, cũng như cập nhật/thêm/xóa dữ liệu từ đó.

LINQ to SQL hỗ trợ đầy đủ transaction, view và các stored procedure (SP). Nó cũng cung cấp một cách dễ dàng để thêm khả năng kiểm tra tính hợp lệ của dữ liệu và các quy tắc vào trong mô hình dữ liệu của bạn.

*1.13.1.3LINQ to XML*

Sử dụng LINQ với mục đích truy vấn file XML và truy vấn tiện lợi hơn nhiều so với việc dùng XmlDocument, Xpath và Xquery như trước kia.

*1.13.1.4LINQ to Datasets*

DataSet trong ADO.NET là một bước phát triển lớn trong việc phát triển ứng dụng cơ sở dữ liệu đa hệ. Khi lấy và chỉnh sửa dữ liệu, duy trì liên tục kết nối tới Data Source trong khi chờ user yêu cầu thì rõ ràng là tốn tài nguyên máy rất nhiều.

DataSet giúp ích ở đây rất lớn. Vì DataSet cho phép lưu trữ dữ liệu và chỉnh sửa tại ‘local cache’, hay gọi là offline mode. Có thể xem xét và xử lý thông tin trong khi ngắt kết nối. Sau khi chỉnh sửa và xem xong thì tạo một kết nối và update dữ liệu từ local vào Data Source.

Dữ liệu trong DataSet được lưu trữ dưới dạng một Collection các Tables và bạn cần phải xử lý thông qua các lớp DataTable (DataRow và DataColumn).

A diagram of a data set

Description automatically generated with medium confidence

# Hình 2-3:Kiến trúc Dataset

**LINQ to DataSet** cho phép người lập trình sử dụng DataSets như một nguồn dữ liệu bình thường bằng các cú pháp truy vấn căn bản của LINQ

*1.13.1.5LINQ to Entities*

**Khái Niệm về Entity Framework**

Entity Framework như giản đồ của các dữ liệu được lưu trữ trong cơ sở dữ liệu và trình bày giản đồ khái niệm ứng dụng của nó. Giản đồ này được ánh xạ từ cơ sỡ dữ liệu.

Entity Framework là cơ sở dữ liệu độc lập và xây dựng dựa trên một mô hình chuẩn ADO.NET cho phép truy cập vào cơ sở dữ liệu của bên thứ ba thông qua việc sử dụng ADO.NET.

A screenshot of a phone

Description automatically generated

# Hình 2-4:Kiến trúc của Entity Framework

Kiến trúc của Entity Framework

LINQ to Entities cho phép chúng ta có thể truy vấn các thực thể bên trong Entity Framework.

**Các Operator chuẩn trongLINQ**

1.Select / SelectMany

2.Where

3.Sum / Min / Max / Average /

Aggregate

4.Join / GroupJoin

5.Take / TakeWhile

6.Skip / SkipWhile

7.OfType

8.Concat

9.OrderBy / ThenBy

10.Reverse

11.GroupBy

12.Distinct

13.Union / Intersect / Except

14.EqualAll

15.First / FirstOrDefault / Last /

LastOrDefault

16.Single

17.ElementAt

18.Any / All / Contains

19.Count

**1.13.2** **LINQ vs ADO.NET**

Nhều người vẫn hay thắc mắc về mối quan hệ giữa LINQ vs ADO.NET, liệu chúng có đối lập nhau hay hỗ trợ lẫn nhau ?

LINQ là tập mở rộng cho phép viết các câu truy vấn ngay trong các ngôn ngữ lập trình. Nó cho phép bạn làm việc với các kiểu tập hợp dữ liệu, như XML, collection, array,… và cả CSDL.

ADO.NET là công nghệ cho phép các ứng dụng có thể kết nối và làm việc với các loại CSDL khác nhau (truy vấn, cập nhật, thêm, xóa, gọi thủ tục…).

Bản thân LINQ không phải là một công nghệ được tạo ra để thay thế ADO.NET, bạn có thể làm việc với LINQ mà không cần đến cơ sở dữ liệu. Tuy nhiên, LINQ to SQL, là một phần mở rộng của LINQ, cho phép bạn có thể làm việc được với CSDL SQL Server, trong trường hợp này thì khi viết bạn có thể bỏ qua các câu lệnhADO.NET mà chỉ quan tâm tới cú pháp mà LINQ cung cấp.

**Chương 3: Phân tích thiết kế hệ thống**

Chương này sẽ đi sâu vào phân tích thiết kế hệ thống, bao gồm các UseCase, phân tích các UseCase.

**1.14 Tổng quan thiết kế**

Chủ cửa hàng, công ty là người có quyền cao nhất: có thể xem thông tin về tình trạng, hoạt động cũng như mọi thay đổi thên website.

Nếu chủ của hàng cũng là người quản trị website thì mọi thông tin trên website web là do chủ cửa hàng quản lý, nếu thuê nhân viên quản trị thì chủ cửa hàng cung cấp tài khoản và quyền hạn cho nhân viên đó truy cập vào hệ thống.

Nhân viên là người giả quyết các nhiệm vụ được giao như các nhận đơn hàng, tiếp thu ý kiến phản hồi của người sửa dụng….

Quyền cao nhất là của chủ cửa hàng, có thể xem đầy đủ mọi thông tin trên website, còn nhân viên và khách hàng chỉ được xem những thông tin được cho phép

Một số trang bán hàng cao cấp có tích hợp nhiều phương thức thanh toán, xong do điều kiện thực tế, phần này khó áp dụng với website có quy mô nhỏ.

**1.14.1 Diagram**

**A diagram of a company

Description automatically generated**

Hình 3-1:Admin,Owner Roles

**A diagram of a diagram

Description automatically generated**

Hình 3-2:User Roles

**1.14.2 Cơ sở dữ liệu**

Cơ sở dữ liệu SQL cho một trang web bán hoa chủ yếu dùng để lưu trữ và quản lý thông tin về sản phẩm, đơn hàng, người dùng và các thông tin liên quan. Dưới đây là một mô tả tổng quan về cấu trúc cơ sở dữ liệu SQL phổ biến cho trang web bán hoa:

* Users

A screenshot of a data

Description automatically generated

Hình 3-3:Bảng User

* Products

A screenshot of a computer

Description automatically generated

Hình 3-4:Bảng Products

* Orders

A screenshot of a data

Description automatically generated

Hình 3-5:Bảng Oders

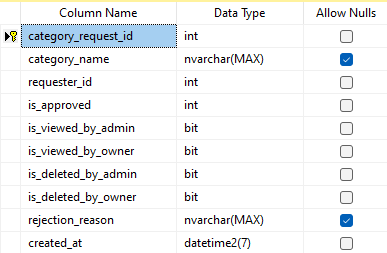
* Order\_items

A screenshot of a data

Description automatically generated

Hình 3-6: Bảng Order\_Items

* Category\_Requests



Hình 3-7: Bảng Category\_Requests

* Categories

A close up of a grid

Description automatically generated

Hình 3-8: Bảng Categories

**1.15 Mô tả sử dụng các giao diện của web SunFlower**

- **Giao diện đăng nhập/đăng ký**

**A screenshot of a computer

Description automatically generated**

**Hình 3-9 : Giao diện Đăng Nhập**

**A screenshot of a computer

Description automatically generated**

**Hình 3-10 : Giao diện Đăng ký tài khoản**

* Form Đăng Nhập: Giao diện đăng nhập cung cấp một form nhập liệu đơn giản với các trường như tên đăng nhập hoặc địa chỉ email và mật khẩu.
* Ghi Nhớ Đăng Nhập: Tính năng cho phép người dùng ghi nhớ thông tin đăng nhập để tiện lợi cho lần đăng nhập sau.
* Quên Mật Khẩu: Cung cấp liên kết hoặc nút để khôi phục mật khẩu nếu người dùng quên.
* Form Đăng Ký: Giao diện đăng ký sử dụng một form chi tiết yêu cầu thông tin như tên, địa chỉ email, mật khẩu, hình ảnh,và thông tin liên hệ khác.
* Xác Nhận Mật Khẩu: Yêu cầu người dùng xác nhận lại mật khẩu để đảm bảo tính chính xác.
* Nút Đăng Nhập/Đăng Ký: Sử dụng nút lớn và rõ ràng để thực hiện các hành động đăng nhập hoặc đăng ký.
* Thông Báo Lỗi: Hiển thị thông báo lỗi một cách rõ ràng khi người dùng nhập sai thông tin hoặc có vấn đề khác liên quan đến đăng nhập/đăng ký002E
* Xác Nhận Đăng Nhập/Đăng Ký Thành Công: Thông báo một cách tích cực khi người dùng đã đăng nhập/đăng ký thành công.
* Đăng Nhập/Đăng Ký Nhanh Chóng: Cung cấp liên kết hoặc nút cho phép người dùng đăng nhập/đăng ký nhanh chóng bằng tài khoản Google, Facebook, hoặc các tài khoản mạng xã hội khác.

**A person holding a bouquet of flowers

Description automatically generated** - **Giao diện Trang Chủ bán hoa**

**Hình 3-11: Giao diện trang chủ bán hoa**

Giao diện trang chủ của trang web bán hoa được thiết kế để mang lại trải nghiệm tuyệt vời cho người sử dụng, tạo cảm giác tươi mới và gần gũi với thế giới của hoa tươi. Dưới đây là mô tả chi tiết về cách sử dụng giao diện trang chủ web bán hoa của chúng tôi.

* Giao diện bắt đầu với một header, chứa logo của cửa hàng bán hoa và thanh menu chính, giúp người dùng dễ dàng điều hướng trang web.
* Một phần giới thiệu ngắn về cửa hàng, mô tả về cam kết chất lượng và sứ mệnh của cửa hàng trong việc mang lại niềm vui qua những bó hoa tươi.
* Hiển thị các danh mục sản phẩm nổi bật, giúp khách hàng dễ dàng tìm kiếm theo loại hoa, dịp kỷ niệm, màu sắc hoặc giá cả.
* Các hình ảnh sản phẩm được trình bày một cách thẩm mỹ và thu hút người xem. Khi di chuột qua sản phẩm, hiển thị thông tin cơ bản và nút mua hàng nhanh chóng.
* Cung cấp ô tìm kiếm nhanh và bộ lọc để người dùng có thể chọn lọc sản phẩm theo nhu cầu cụ thể của họ, như màu sắc, loại hoa, giá cả, v.v.
* Chân trang chứa các liên kết quan trọng như trang giới thiệu, chính sách bảo mật, điều khoản sử dụng và các mạng xã hội của cửa hàng.

**- Giao diện sản phẩm**

**A screenshot of a flower shop

Description automatically generated**

**Hình 3-12: Giao diện sản phẩm**

**A screenshot of a computer

Description automatically generated**

**Hình 3-13: Footer**

* Danh Mục Rõ Ràng: Phân loại hoa thành các danh mục như hoa tươi, hoa nhân dịp, và hoa theo mùa để người dùng dễ dàng tìm kiếm.

**- Giao diện mô tả sản phẩm**

**A person holding a bouquet of flowers

Description automatically generated**

**Hình 3-14: Giao diện mô tả sản phẩm**

* Thông Tin Chi Tiết: Hiển thị mô tả chi tiết về sản phẩm, giá cả, thông tin về nguồn gốc, các loại hoa tạo nên và hoa chủ đạo..
* Nút Đặt Hàng Nhanh Chóng: Cung cấp nút đặt hàng ngay để người dùng có thể mua sản phẩm một cách thuận tiện.

**- Giao diện giỏ hàng**

**A screenshot of a computer

Description automatically generated**

**Hình 3-15: Giao diện giỏ hàng**

* Quản Lý Giỏ Hàng: Cho phép người dùng kiểm tra và chỉnh sửa giỏ hàng trước khi tiến hành thanh toán.
* Nhiều Phương Thức Thanh Toán: Hỗ trợ nhiều phương thức thanh toán an toàn và tiện lợi như thẻ tín dụng, chuyển khoản ngân hàng, hoặc thanh toán khi nhận hàng.
* Thông tin khách đặt hàng cần điền vào về thuận tiện cho việc giao hàng.

**- Giao diện thông tin người dùng**

**A screenshot of a computer

Description automatically generated**

**Hình 3-16: Giao diện thông tin người dùng**

* Đăng Ký và Đăng Nhập: Cho phép người dùng tạo tài khoản để theo dõi đơn hàng và lưu trữ thông tin cá nhân.
* Lịch Sử Đơn Hàng: Hiển thị lịch sử đơn hàng để người dùng có thể theo dõi các giao dịch trước đây.

A screenshot of a computer

Description automatically generated

**Hình 3-17: Đơn hàng đã đặt**

**- Giao diện thêm/sửa/xóa mặt hàng**

**A screenshot of a computer

Description automatically generated**

**Hình 3-18: Giao diện thêm/sửa/xóa mặt hàng**

* Mẫu Nhập Liệu: Admin có một mẫu nhập liệu hoặc giao diện đơn giản để điền thông tin về sản phẩm mới, bao gồm tên sản phẩm, mô tả, giá cả, loại hoa, hình ảnh, và các thuộc tính khác.
* Quản Lý Hình Ảnh: Cung cấp khả năng tải lên và quản lý hình ảnh sản phẩm để hiển thị chất lượng cao trên trang web.
* Xác Nhận Xóa: Trước khi xóa sản phẩm, admin được yêu cầu xác nhận để tránh xóa thông tin không mong muốn.
* Thêm và Xóa Danh Mục: Admin có thể thêm hoặc xóa các danh mục sản phẩm để cập nhật và sắp xếp sản phẩm theo cách hợp lý.
* Nhập và Chỉnh Sửa Giá: Admin có thể nhập giá mới cho sản phẩm và thực hiện điều chỉnh khi cần thiết.

**- Giao diện quản lý người dùng**

**A screenshot of a computer

Description automatically generated**

**Hình 3-19 Giao diện quản lý người dùng**

**A screenshot of a computer

Description automatically generated**

**Hình 3-20: Giao diện quản lí đơn hàng**

* Tổng Quan: Trang quản lý tài khoản người dùng cung cấp một tổng quan về tất cả các tài khoản đang hoạt động trên trang web.
* Xem và Chỉnh Sửa Thông Tin: Admin có thể xem và chỉnh sửa thông tin cá nhân của người dùng, bao gồm tên, địa chỉ, số điện thoại, và địa chỉ email.
* Quản Lý Đơn Hàng: Xem chi tiết đơn hàng của người dùng, bao gồm trạng thái đơn hàng, sản phẩm đã mua.
* Phân Quyền Người Dùng: Admin có khả năng gán quyền và phân quyền cho các nhóm người dùng, chẳng hạn như quyền xem, chỉnh sửa, hoặc xóa thông tin.

A screenshot of a computer

Description automatically generated

**Hình 3-21: Phân quyền người dùng**

* Quản Lý Quyền Truy Cập: Điều chỉnh quyền truy cập của admin và nhân viên để đảm bảo an toàn thông tin người dùng.
* Giám Sát Hoạt Động Tài Khoản: Cung cấp khả năng theo dõi và giám sát các hoạt động trên tài khoản admin để ngăn chặn các hành động không phù hợp.

**- Giao diện mua hàng thành công**

**A screenshot of a computer

Description automatically generated**

**Hình 3-22 : Giao diện mua hàng thành công**

* Form Liên Hệ và Hỗ Trợ Trực Tuyến: Cung cấp form liên hệ và tính năng hỗ trợ trực tuyến để giúp đỡ khách hàng khi cần thiết.
* Lời Nhắn : Hiển thị thông báo cảm ơn và đường link chăm sóc khách hàng.

**CHƯƠNG 4: KẾT LUẬN**

* **Kết quả đạt được**

•Kết quả đạt được:

- Website kết nối được CSDL.

- Website cho phép người dùng đăng ký, đăng nhập, xem sản phẩm, xem chi tiết sản phẩm,thêm đồ vào giỏ hàng và thanh toán.

- Website cho phép người quản lý đăng nhập, quản lý thông tin về các loại hoa, quản lý danh mục các loại hoa, quản lý khách hàng và thông tin đơn hàng.

* **Những hạn chế cần khắc phục**

•Ngoài ra còn những hạn chế cần khắc phục:

+ Thiết kế giao diện còn đơn giản chưa đạt tính thẩm mỹ cao.

+ Các chức năng chưa được tối ưu hóa cao nhất.

Trong quá trình thực hiện đề tài, em đã cố gắng rất nhiều nhưng vẫn

không tránh khỏi những thiếu sót cũng như những hạn chế nhất định. Em mong

nhận được sự đóng góp ý kiến, chỉ bảo tận tình của các thầy cô và các bạn để đề

tài của em hoàn thiện hơn và thiết thực hơn.

TÀI LIỆU THAM KHẢO

Website tài liệu tham khảo:

[1] http://www.codeproject.com

[2] https://www.youtube.com/

[3] https://www.howkteam.vn/course/lap-trinh-website-voi-aspnet-mvc-co-ban-15

[4] https://tedu.com.vn/khoa-hoc-mien-phi/khoa-hoc-lap-trinh-aspnet-mvc-25.htm