LẬP TRÌNH WEB

Nguyễn Thị Mai Trang

1

TRƯỜNG ĐẠI HỌC MỞ TP. HCM
Cơ hội học tập cho mọi người

Nội dung

- Chương 1: Tổng quan
- Chương 2: Ngôn ngữ PHP
- Chương 3: Controller – View – Model
- Chương 4: HTML Helpers
- Chương 5: Làm việc với Cơ sở dữ liệu

2

TRƯỜNG ĐẠI HỌC MỞ TP. HCM
Cơ hội học tập cho mọi người

Chương 1

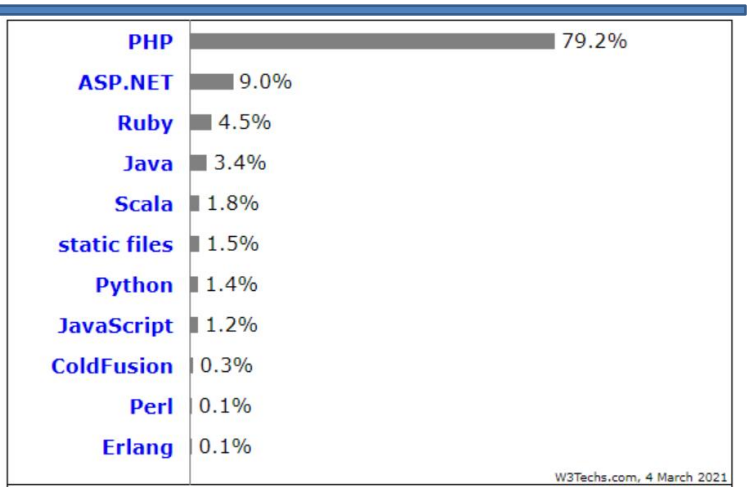
Tổng quan

3

TRƯỜNG ĐẠI HỌC MỞ TP. HCM
Cơ hội học tập cho mọi người

Nội dung chương 1

- Tổng quan về PHP
- Tổng quan về ASP.NET MVC



Percentages of websites using various server-side programming languages
Note: a website may use more than one server-side programming language

4

TRƯỜNG ĐẠI HỌC MỞ TP. HCM
Cơ hội học tập cho mọi người

1.1 Tổng quan về PHP

- Giới thiệu PHP
- Giới thiệu một số framework PHP

5

TRƯỜNG ĐẠI HỌC MỞ TP. HCM
Cơ hội học tập cho mọi người

1.2.1 Giới thiệu PHP

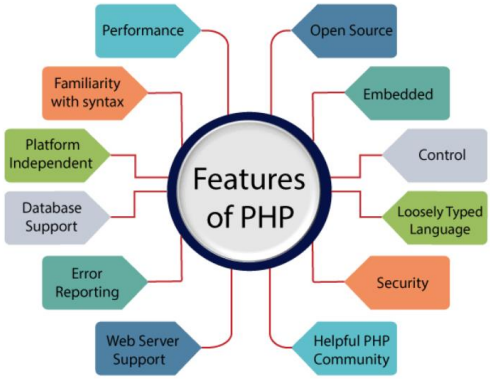
- PHP (Personal Home Page) ra đời năm 1994 do phát minh của Rasmus Lerdorf, sau đó được sử dụng trong môi trường chuyên nghiệp và trở thành chữ viết tắt của Hypertext Preprocessor.
- PHP là ngôn ngữ kịch bản mã nguồn mở, đơn giản và dễ học
- PHP là ngôn ngữ kịch bản phía máy chủ (server script) chạy trên phía server (server side).
- PHP là một ngôn ngữ thông dịch.
- PHP có thể được nhúng vào HTML.
- PHP là một ngôn ngữ hướng đối tượng.
- Cho phép xây dựng ứng dụng web với các HQTCSDL MySQL, Oracle, SQL Server…
- PHP không phụ thuộc vào môi trường (cross-platform), có thể sử dụng được trên nhiều hệ điều hành: Unix, Linux, Windows…

6

TRƯỜNG ĐẠI HỌC MỞ TP. HCM
Cơ hội học tập cho mọi người

Giới thiệu PHP (tt)

- Code PHP được soạn thảo bằng một phần mềm Text Editor như Notepad, EditPlus, Dreamweaver, Visual Code…
- Có nhiều framework
- Các đặc trưng của PHP:



7

TRƯỜNG ĐẠI HỌC MỞ TP. HCM
Cơ hội học tập cho mọi người

Giới thiệu PHP (tt)

- Cài đặt PHP:
  - Download PHP ở địa chỉ: http://php.net/downloads.php
  - Download tài liệu tham khảo : http://www.php.net/docs.php
  - Cài đặt cấu hình WebServer và database (IIS, Apache hoặc PhpMyAdmin,….)
  - Có thể cài Xampp, Wamp server làm máy thực thi PHP code và thao tác với cơ sở dữ liệu MySQL một cách dễ dàng

8

TRƯỜNG ĐẠI HỌC MỞ TP. HCM
Cơ hội học tập cho mọi người

Giới thiệu PHP (tt)

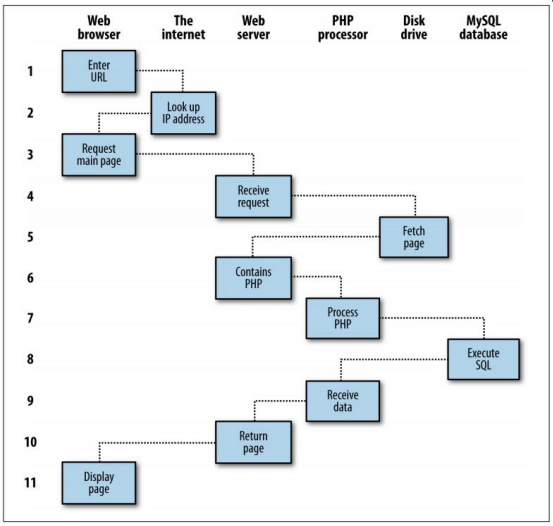
- Các bước phản hồi một request từ trình duyệt đến ứng dụng PHP
  1. User nhập URL http://server.com từ trình duyệt
  2. Trình duyệt xác định địa chỉ IP của máy chủ server.com
  3. Trình duyệt gởi yêu cầu trang **home page** trong thư mục web trên máy chủ server.com
  4. Yêu cầu từ trình duyệt được gởi đến chương trình Web Server trên máy chủ server.com
  5. Web server trên máy chủ server.com nhận yêu cầu, tìm nạp trang home page từ đĩa lên bộ nhớ
  6. Web server chuyển trang web đến trình thông dịch PHP
  7. Trình thông dịch PHP thực thi mã lệnh PHP
  8. Một số mã PHP chứa các câu lệnh SQL, trình thông dịch PHP chuyển đến hệ quản trị CSDL MySQL
  9. MySQL database trả về kết quả của các câu lệnh cho trình thông dịch PHP
  10. Trình thông dịch PHP trả về kết quả của mã PHP, cùng với kết quả từ cơ sở dữ liệu MySQL, đến web server
  11. Web server trả nội dung trang web dạng HTML về cho trình duyệt, trình duyệt hiển thị trang web

9

TRƯỜNG ĐẠI HỌC MỞ TP. HCM
Cơ hội học tập cho mọi người

Giới thiệu PHP (tt)

- Các bước phản hồi một request từ trình duyệt đến ứng dụng PHP



10

TRƯỜNG ĐẠI HỌC MỞ TP. HCM
Cơ hội học tập cho mọi người

Giới thiệu PHP (tt)

- Trang web PHP:
  - Trang web php có phần mở rộng là .php
  - Trang php có thể kết hợp với mã HTML, JavaScript, CSS
  - Mã lệnh PHP được đặt trong cặp thẻ
    <?php và **?>**
    hoặc
    <? và **?>**
  - Ví dụ:

```
<?php
        echo "Hello world";
?>
```

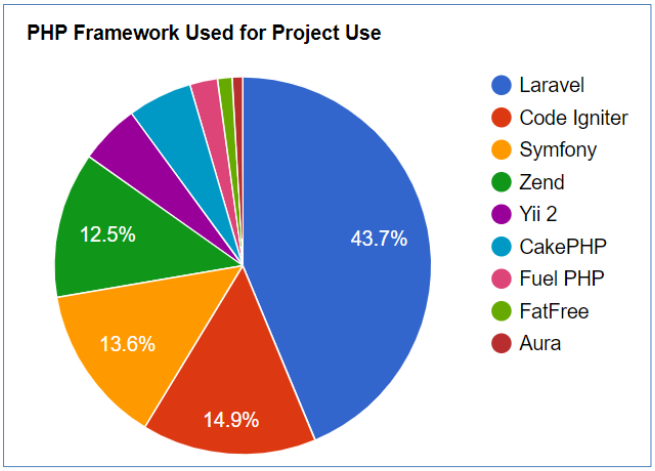
```
<?
        echo "Hello world";
?>
```


11

TRƯỜNG ĐẠI HỌC MỞ TP. HCM
Cơ hội học tập cho mọi người

1.2.2 Giới thiệu một số framework PHP

- Laravel
- Codeigniter
- Symfony
- Cakephp
- Zend Framework
- ...



Top Rated PHP Frameworks of 2019

12

TRƯỜNG ĐẠI HỌC MỞ TP. HCM
Cơ hội học tập cho mọi người

1.2 Tổng quan về ASP.NET MVC

- Giới thiệu ASP.NET MVC
- Tạo ứng dụng ASP.NET MVC
- Các đối tượng duy trì trạng thái ứng dụng
- Định tuyến (Routing)

TRƯỜNG ĐẠI HỌC MỞ TP. HCM
Cơ hội học tập cho mọi người

1.2.1 Giới thiệu ASP.NET MVC

- ASP.NET: công nghệ để xây dựng ứng dụng Web, dịch vụ Web và các ứng dụng khác của Microsoft.
  - ASP.NET WebForm
  - ASP.NET MVC
- ASP.NET Core: phát triển ứng dụng đa nền tảng, mã nguồn mở
  - ASP.NET Core MVC
- ASP.NET Web service, ASP.NET Web API: các dịch vụ web

TRƯỜNG ĐẠI HỌC MỞ TP. HCM
Cơ hội học tập cho mọi người

Giới thiệu ASP.NET MVC (tt)

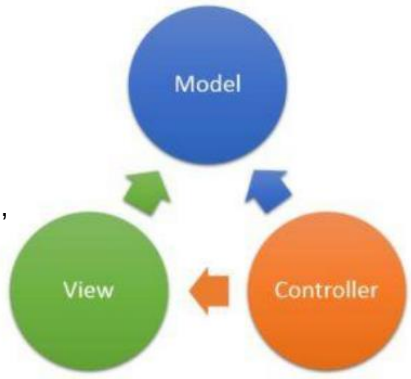
- **Model-View-Controller (MVC)**:
  - Là một mô hình kiến trúc quan trọng trong khoa học máy tính.
  - Ban đầu được đặt tên là Thing-Model-View-Editor vào năm 1979 →Model-View-Controller
  - Phân tách các mối quan tâm trong một ứng dụng
    - Tách về mặt logic việc truy cập dữ liệu với hiển thị dữ liệu
    - Áp dụng rất tốt cho các ứng dụng web
  - Một số MVC framework phát triển dựa trên mô hình MVC
    - Spring MVC (Java)
    - Zeng, CodeIgniter, Laravel (PHP)
    - ASP.NET MVC

15

TRƯỜNG ĐẠI HỌC MỞ TP. HCM
Cơ hội học tập cho mọi người

Giới thiệu ASP.NET MVC (tt)

- Model-View-Controller (tt):
  - Phân chia ứng dụng thành ba lớp logic:
    - Model: tập hợp các lớp mô tả dữ liệu và các quy tắc, thao tác dữ liệu.
    - View: xác định cách hiển thị, cung cấp giao diện tương tác với người dùng.
    - Controller: tập hợp các lớp xử lý giao tiếp từ người dùng, luồng ứng dụng tổng thể và logic dành riêng cho ứng dụng, hoạt động giữa lớp View và Model

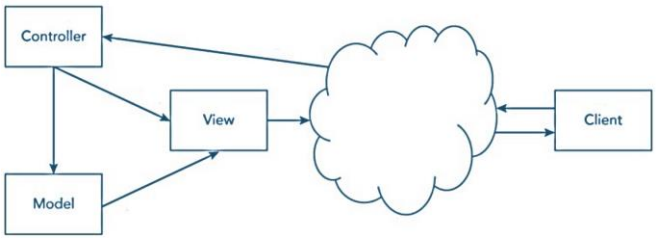


16

TRƯỜNG ĐẠI HỌC MỞ TP. HCM
Cơ hội học tập cho mọi người

Giới thiệu ASP.NET MVC (tt)

- ASP.NET MVC
  - Là một framework cho phép phát triển ứng dụng web động sử dụng .NET Framework theo mô hình MVC
  - Chức năng các lớp logic trong ASP.NET MVC:
    - **Model**: các lớp truy cập dữ liệu, xử lý dữ liệu lưu trữ trong cơ sở dữ liệu (Entity Framework, Hibernate)
    - **View**: mẫu để tạo HTML động, trình bày nội dung trang web.
    - **Controller**: lớp quản lý, xử lý mối quan hệ giữa View và Model, phản hồi đầu vào của người dùng, tương tác với Model và trả kết quả cho lớp View.

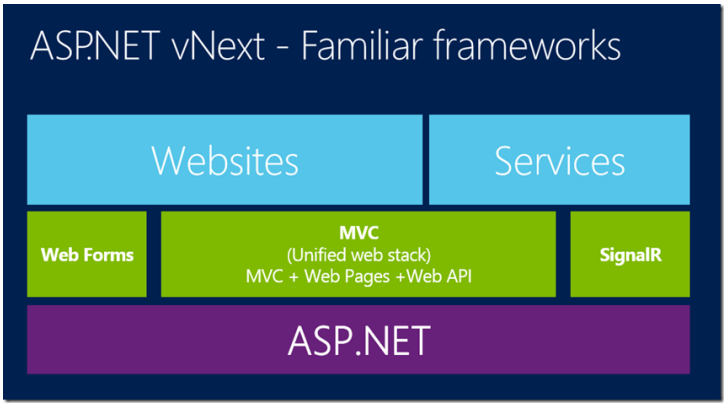


17

TRƯỜNG ĐẠI HỌC MỞ TP. HCM
Cơ hội học tập cho mọi người

Giới thiệu ASP.NET MVC (tt)

- ASP.NET MVC trong .NET Framework



18

TRƯỜNG ĐẠI HỌC MỞ TP. HCM
Cơ hội học tập cho mọi người

Giới thiệu ASP.NET MVC (tt)

- Lợi ích của ASP.NET MVC
  - Phân tách ứng dụng thành các thành phần (Model, View, Controller) trong khi vẫn cung cấp một mối quan hệ không chặt chẽ giữa chúng → cho phép thay đổi một trong các thành phần mà không ảnh hưởng nhiều đến thành phần khác
  - Kiểm tra và bảo trì đơn giản: cho phép kiểm tra từng thành phần một cách độc lập, đơn giản hóa quy trình kiểm tra, bảo trì và xử lý sự cố.
  - Khả năng mở rộng: dễ dàng sửa đổi hoặc thay thế theo yêu cầu ứng dụng.
  - Hỗ trợ chức năng kiểm thử

19

TRƯỜNG ĐẠI HỌC MỞ TP. HCM
Cơ hội học tập cho mọi người

Giới thiệu ASP.NET MVC (tt)

- Kiến trúc của ứng dụng ASP.NET MVC: gồm các thành phần
  - MVC Framework
  - Route engine, Route configuration
  - Model, View, Controller
  - View Engine
- ASP.NET MVC hỗ trợ các kỹ thuật tạo web động và mang tính đáp ứng như:
  - JavaScript, JQuery
  - AJAX
  - IIS
  - Windows Azure

20

TRƯỜNG ĐẠI HỌC MỞ TP. HCM
Cơ hội học tập cho mọi người

Giới thiệu ASP.NET MVC (tt)

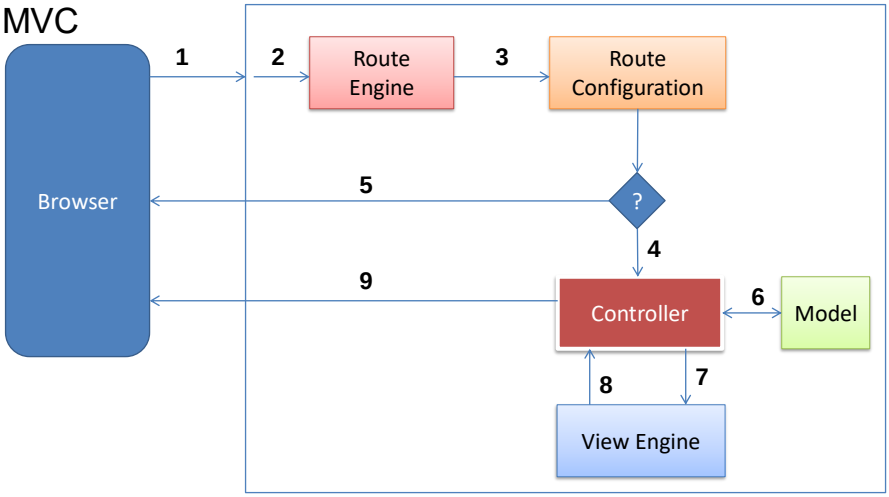
- Các bước ASP.NET MVC Framework xử lý yêu cầu từ trình duyệt:
  1. Trình duyệt gửi yêu cầu đến ứng dụng ASP.NET MVC
  2. MVC Framework chuyển tiếp yêu cầu đến routing engine
  3. routing engine kiểm tra cấu hình định tuyến của ứng dụng để tìm bộ điều khiển (Controller) thích hợp để xử lý yêu cầu.
     - Nếu tìm thấy: thực thi
     - Ngược lại: trả về lỗi cho phía trình duyệt
  4. Controller giao tiếp với Model (nếu cần)
  5. Controller yêu cầu một view engine cấp một View dựa trên dữ liệu của Model
  6. View engine trả về kết quả cho controller
  7. Controller trả về kết quả dưới dạng HTTP response cho trình duyệt

21

TRƯỜNG ĐẠI HỌC MỞ TP. HCM
Cơ hội học tập cho mọi người

Giới thiệu ASP.NET MVC (tt)

- Các bước ASP.NET MVC Framework xử lý yêu cầu từ trình duyệt:

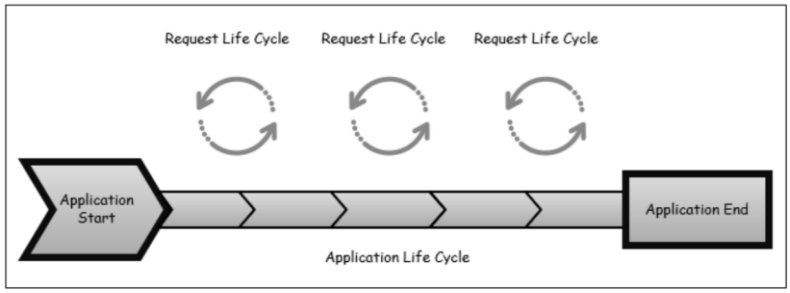


22

TRƯỜNG ĐẠI HỌC MỞ TP. HCM
Cơ hội học tập cho mọi người

Giới thiệu ASP.NET MVC (tt)

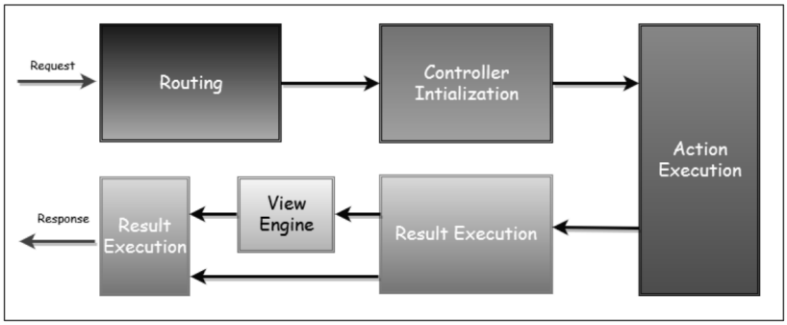
- **Vòng đời của ứng dụng web**
  - Bắt đầu khi ứng dụng web bắt đầu chạy trên web server cho đến khi nó kết thúc
  - Đánh dấu bằng các sự kiện Application_Start() và Application_End () trong tập tin khởi động của ứng dụng (Global.asax)



TRƯỜNG ĐẠI HỌC MỞ TP. HCM
Cơ hội học tập cho mọi người

Giới thiệu ASP.NET MVC (tt)

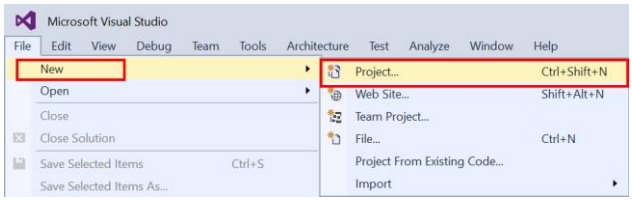
- **Vòng đời của một Request**
  - Là chuỗi các sự kiện xảy ra mỗi khi yêu cầu HTTP được ứng dụng xử lý.
  - Điểm bắt đầu cho mọi ứng dụng MVC là định tuyến, ASP.NET platform nhận được yêu cầu sẽ xử lý thông qua URL Routing Module

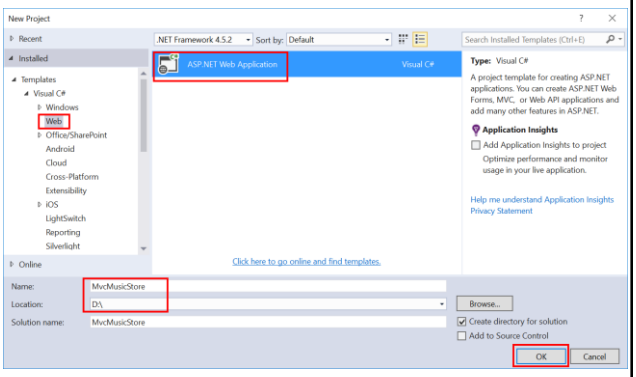


TRƯỜNG ĐẠI HỌC MỞ TP. HCM
Cơ hội học tập cho mọi người

1.2.2 Tạo ứng dụng ASP.NET MVC

- File → New → Project...



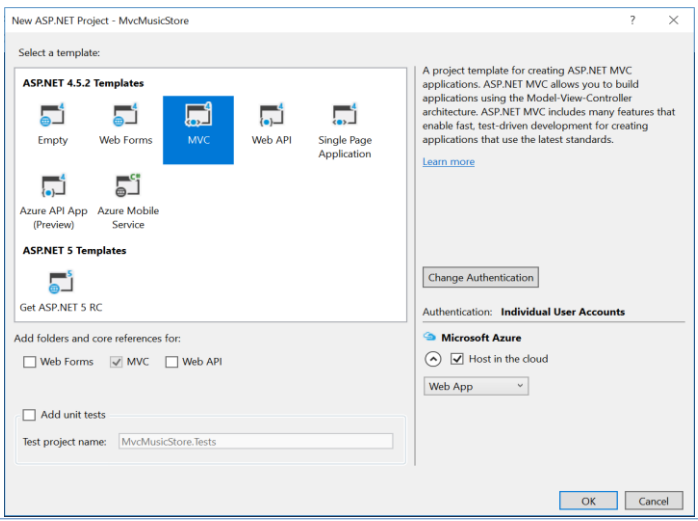


25

TRƯỜNG ĐẠI HỌC MỞ TP. HCM
Cơ hội học tập cho mọi người

Tạo ứng dụng ASP.NET MVC (tt)

- Chọn MVC

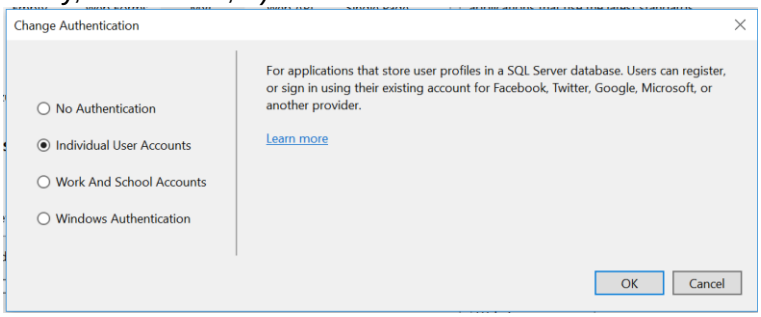


26

TRƯỜNG ĐẠI HỌC MỞ TP. HCM
Cơ hội học tập cho mọi người

Tạo ứng dụng ASP.NET MVC (tt)

- Có thể click vào Change Authentication để thay đổi phương thức xác thực
  - No Authentication: không xác thực
  - Individual User Accounts: xác thực tài khoản người dùng cá nhân
  - Organizational Accounts: xác thực tài khoản của tổ chức thông qua một số dạng Active Directory (Azure Active Directory, Office 365,...)
  - Windows Authentication: xác thực tài khoản trên Windows

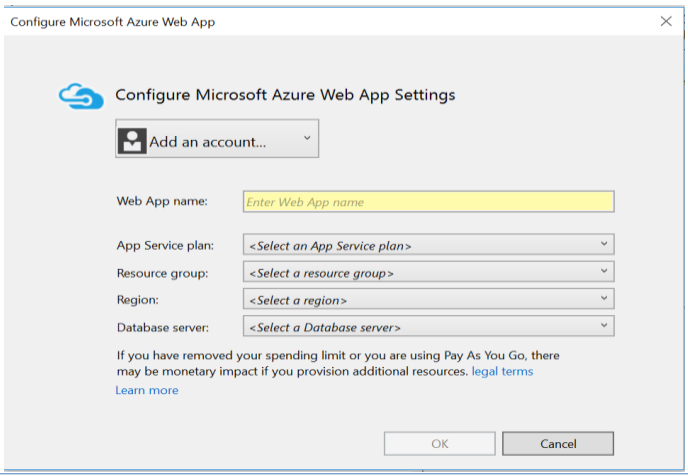


27

TRƯỜNG ĐẠI HỌC MỞ TP. HCM
Cơ hội học tập cho mọi người

Tạo ứng dụng ASP.NET MVC (tt)

- Có thể cấu hình lưu trữ ứng dụng web trên máy chủ Azure của Microsoft (phải có tài khoản)

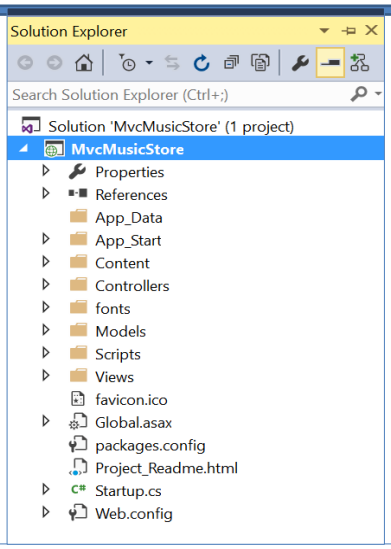


28

TRƯỜNG ĐẠI HỌC MỞ TP. HCM
Cơ hội học tập cho mọi người

Tạo ứng dụng ASP.NET MVC (tt)

- Cấu trúc của ứng dụng MVC
  - **Controllers**: các lớp điều khiển đáp ứng yêu cầu đến từ trình duyệt, xử lý và phản hồi kết quả.
  - **Models**: chứa các mô hình lưu trữ và thao tác với dữ liệu
  - **Views**: chứa các tập tin .cshtml để hiển thị giao diện
  - **App_Data**: chứa các file cơ sở dữ liệu (nếu có)
  - **App_Start**: chứa các lớp cấu hình sẽ kích hoạt thực thi trước khi ứng dụng web thực thi
  - **Content**: chứa tài nguyên tĩnh như hình ảnh, CSS…
  - **Scripts**: chứa các tập tin Java Script cần thiết của ứng dụng
  - **Global.asax**: tập tin chứa lớp cấu hình Session, Cookies, Application cho ứng dụng web
  - **Web.config**: tập tin chứa thông tin cấu hình của ứng dụng



TRƯỜNG ĐẠI HỌC MỞ TP. HCM
Cơ hội học tập cho mọi người

1.2.3 Các đối tượng duy trì trạng thái ứng dụng

- Application
- Session
- Cookie

TRƯỜNG ĐẠI HỌC MỞ TP. HCM
Cơ hội học tập cho mọi người

Application

- Đối tượng được dùng để lưu trữ dữ liệu trên toàn ứng dụng
- Cách sử dụng:
  - Trong Global.asax:
    - Application[“Visitor"] = 0;
  - Controller: HttpContext.Application
    - HttpContext.Application[“Visitor"] = 0;
  - View: @AppState
    - @AppState[“Visitor"]
  - Class bất kỳ: HttpContext.Current.Application
    - HttpContext.Current.Application[“Visitor"] = 0;

31

TRƯỜNG ĐẠI HỌC MỞ TP. HCM
Cơ hội học tập cho mọi người

Application (tt)

- Một số thao tác cơ bản với Application
  - Thêm mới một đối tượng vào Application:
    - Add (Key, Value)
      ví dụ: HttpContext.Application.Add("Visitor", 0);
    - HttpContext.Application[Key] = Value
      ví dụ: HttpContext.Application["Visitor"] = 0;
  - Xóa đối tượng: Remove(Key)
    ví dụ: HttpContext.Application.Remove("Visitor");
  - Xóa tất cả các đối tượng trong Application: Clear()
    ví dụ: HttpContext.Application.Clear();
  - Khóa Application:Lock()
    ví dụ HttpContext.Application.Lock();
  - Mở khóa Application: UnLock()
    ví dụ: HttpContext.Application.UnLock();

32

TRƯỜNG ĐẠI HỌC MỞ TP. HCM
Cơ hội học tập cho mọi người

Application (tt)

- Ví dụ sử dụng Application: đếm số người truy cập online
  - Trang Global.asax

```
public class MvcApplication : System.Web.HttpApplication
{
    0 references
    protected void Application_Start()
    {
        Application["visitor"] = 0;
        AreaRegistration.RegisterAllAreas();
        FilterConfig.RegisterGlobalFilters(GlobalFilters.Filters);
        RouteConfig.RegisterRoutes(RouteTable.Routes);
        BundleConfig.RegisterBundles(BundleTable.Bundles);
    }
    0 references
    protected void Session_Start()
    {
        long n = Convert.ToInt64(Application["visitor"]);
        Application["visitor"] = n + 1;
    }
}
```

  - Trang Index.cshtml: @AppState["visitor"]

TRƯỜNG ĐẠI HỌC MỞ TP. HCM
Cơ hội học tập cho mọi người

Session

- Đối tượng được dùng để lưu trữ dữ liệu cho một phiên làm việc, sẽ được giải phóng khi hết phiên làm việc (chuyển sang website khác, đóng trình duyệt…)
- Trong **Controller**
  - Session[“user”] = “admin”;
- Trong **View**
  - @Session[“user”]
- Trong một **class** bất kỳ
  - HttpContext.Current.Session[“user”] = “admin”;

TRƯỜNG ĐẠI HỌC MỞ TP. HCM
Cơ hội học tập cho mọi người

Session (tt)

- Một số thao tác cơ bản với Session
  - Tạo Session:
    - Session. Add(Key, Value)
      ví dụ: Session.Add("Name",“Cathy");
    - Session[“key"] = “value";
      ví dụ: Session[“Name”] =“Cathy”;
  - Xóa một đối tượng trong Session: Session.Remove(“key")
  - Xóa tất cả các đối tượng trong Session: Session.Clear()
  - Hủy Session: Session.Abandon()
  - Lấy id của Session hiện tại: truy xuất thuộc tính Session.SessionID
    - Ví dụ: var id = Session.SessionId;

35

TRƯỜNG ĐẠI HỌC MỞ TP. HCM
Cơ hội học tập cho mọi người

Session (tt)

- Ví dụ sử dụng Session
  - Sau khi thêm khách hàng thành công →lưu vào session→ hiển thị tên khách hàng

```
[HttpPost]
[ValidateAntiForgeryToken]
0 references
public ActionResult Create([Bind(Include = "MaKH,TenKH,DiaChi,DienThoai,Fax,Email")] KhachHang khachHang)
{
    if (ModelState.IsValid)
    {
        db.KhachHangs.Add(khachHang);
        db.SaveChanges();
        Session["khachhang"] = khachHang;
        return RedirectToAction("Index");
    }

    return View(khachHang);
}
```

Controller

```
@{
    var kh = @Session["khachhang"] as QLBanhang.Models.KhachHang;
    if(@kh!=null)
    <h1>@kh.TenKH</h1>
}
```

View

36

TRƯỜNG ĐẠI HỌC MỞ TP. HCM
Cơ hội học tập cho mọi người

Cookie

- Là dữ liệu được lưu trữ trên máy tính của client, dạng text, kích thước nhỏ (4KB)
- Cookie được gởi qua lại giữa trình duyệt và Server thông qua Request và Response.
- Sử dụng Cookie:
  - **Controller**: Request.Cookies
    - Request.Cookies["id"];
  - **View**: @Request.Cookies
    - @Request.Cookies["id"]
  - **Model**: HttpContext.Current.Request.Cookies
    - HttpContext.Current.Request.Cookies["id"];

37

TRƯỜNG ĐẠI HỌC MỞ TP. HCM
Cơ hội học tập cho mọi người

Cookie (tt)

- Một số thuộc tính của Cookie:
  - Name: tên Cookie
  - Value: giá trị của Cookie đơn
  - Expires: thời gian tồn tại của cookie
  - Values: tập các giá trị trong Cookie
- Ví dụ:
  - myCookie.Name = “id1";
  - myCookie.Value = “value1";
  - myCookie.Expires = DateTime.Now.AddDays(1);
  - myCookie.Values.Add("id2", “value2"); (thêm một giá trị vào Cookie)
  - myCookie.Values["id1"] = “new-value"; (cập nhật giá trị cho Cookie có thuộc tính name=“id1”)
  - myCookie.Expires = DateTime.Now.AddDays(-1); (xóa Cookie)

38