**TRƯỜNG ĐẠI HỌC SÀI GÒN**

**KHOA CÔNG NGHỆ THÔNG TIN**

----- 🙡 🕮 🙣 -----



**ĐỀ CƯƠNG ĐỀ TÀI : ASP .NET CORE**

**Giáo viên hướng dẫn** : Phạm Thi Vương

**Nhóm sinh viên thực hiện**

3120410586 - Nguyễn Anh Tuấn

3120410264 - Vũ Trung Kiên

3120410005 - Phạm Quang Minh

Thành phố Hồ Chí Minh, 2024

CHƯƠNG I. GIỚI THIỆU

1.1. Giới thiệu ASP.NET Core

1.2. Lý do chọn đề tài

CHƯƠNG II. TỔNG QUAN VỀ ASP.NET CORE

2.1. ASP.NET Core là gì?

2.2. Lịch sử phát triển của ASP.NET Core

2.3. Các thành phần chính của ASP.NET

2.4. So sánh ASP.NET với PHP

2.5. Ứng dụng thực tế của ASP.NET

2.6. Cấu trúc dự án ASP.NET Core

2.7. Xây dựng ứng dụng đầu tiên

CHƯƠNG III. KESTREL – ASP.NET CORE WEB SERVER

3.1. Kestrel là gì?

3.2. Tại sao sử dụng Kestrel

3.3. Sử dụng Kestrel

3.4. Triển khai ứng dụng ASP.NET trên Server Linux với Kestrel Apache Nginx

CHƯƠNG IV. MIDDLEWARE VÀ REQUEST PIPELINE TRONG ASP.NET CORE

4.1. Middleware là gì?

4.2. Request Pipeline

4.3. Tạo Middleware riêng trong ASP.NET Core

4.4. Xây dựng Middleware triển khai từ IMiddleware

4.4. Truyền dữ liệu giữa các Middleware

CHƯƠNG V. LÀM VIỆC VỚI FILE CẤU HÌNH TRONG ASP.NET CORE

5.1. Cấu hình là gì

5.2. Cấu hình trong ASP.NET Core

5.3. Nguồn cấu hình

5.4. Load cấu hình

5.5. Đọc cấu hình

5.6. Thêm file cấu hình theo ý muốn

CHƯƠNG VI. SỬ DỤNG STATIC FILES TRONG ASP.NET CORE

CHƯƠNG VII. MVC DESIGN PATTERN TRONG ASP.NET CORE

7.1. MVC Design Pattern trong ASP.NET Core

7.2. Chia để trị (Separation of concerns)

7.3. MVC Pattern làm việc trong ASP.NET Core ra sao?

CHƯƠNG VIII. XÂY DỰNG ỨNG DỤNG ASP.NET CORE MVC ĐẦU TIÊN

8.1. Tạo ứng dụng ASP.NET MVC đầu tiên

8.2. Truy cập ứng dụng với HTTPS

8.3. Route - Định tuyển trong ASP.NET MVC

8.4. Tạo Controller trong ASP.NET MVC

8.5. Tạo View trong ASP.NET MVC - Razor View Engine

8.6. Truyền dữ liệu Controller tới View với ViewBag

8.7. Truyền dữ liệu Controller tới View với Model

8.8. Phát sinh controller, view, razor page, area

CHƯƠNG IX. CƠ BẢN VỀ ASP.NET CORE CONTROLLER

9.1. Controller là gì?

9.2. Thêm mới controller

CHƯƠNG X. ROUTING TRONG ASP.NET CORE MVC

10.1. Một vài định nghĩa

10.2. Routing mặc định của dự án MVC

10.3. Routing trong ASP.NET Core MVC

10.4. Conventional Routing

10.5. Attribute Routing

CHƯƠNG XI. ROUTE CONSTRAINS

11.1. Route Constrains làm việc như thế nào?

11.2. Sử dụng Constraints

11.3. Danh sách của Route Constraint

CHƯƠNG XII. ACTION SELECTORS & ACTION VERBS

12.1. Action Selector là gì?

12.1.1. Action Name

12.1.2. Non Action

12.1.3. Action Verbs

CHƯƠNG XIII. ACTION RESULT

13.1. Action Result là gì?

13.2. Sử dụng Action Result

13.3. Các loại Action Result

CHƯƠNG XIV. VIEW

14.1. Định nghĩa?

14.2. ASP.NET Core tìm View như thế nào?

CHƯƠNG XV. RAZOR VIEW ENGINE TRONG ASP.NET CORE MVC

15.1. Định nghĩa

15.2. Cú pháp Razor

CHƯƠNG XVI. SỬ DỤNG LAYOUTS VÀ SECTION

16.1. Layout của trang web

16.2. Layout trong ASP.NET Core là gì?

16.3. Layout

16.4. Section

CHƯƠNG XVII. VIEWBAG VÀ VIEWDATA

17.1. ViewData

17.2. ViewBag

17.3. So sánh ViewData và ViewBag

CHƯƠNG XVIII. MODEL VÀ VIEWMODEL

18.1. Model là gì?

18.2. Model trong MVC Design Pattern

18.3. Best practice với ViewModel

18.4. Truyền dữ liệu từ Controller sang View

CHƯƠNG XIX. XÂY DỰNG HTML FORM

19.1. Tạo mới project

19.2. Cài đặt MVC Middleware

19.3. Tạo một View model

19.4. Tạo một Controller

19.5. Tạo một View

19.6. Index View

CHƯƠNG XX. TAG HELPERS TRONG ASP.NET CORE MVC

20.1. Tag Helper là gì?

20.2. Sử dụng Tag Helper?

20.3. Lợi ích của Tag Helper

CHƯƠNG XXI. CƠ CHẾ MODEL BINDING: TRUYỀN DỮ LIỆU TỪ VIEW LÊN CONTROLLER

21.1. Model Binding là gì?

21.2. Lấy dữ liệu từ Form Data trong Controller

21.3. Cơ chế Model Binding?

21.4. Khi không dùng model binding

21.5. Các nguồn cho Model binding

21.6. Điều khiển Binding Source

CHƯƠNG XXII. VALIDATION

22.1. Giới thiệu về Model Validation

22.2. Model Validation làm việc ra sao?

22.3. Cách sử dụng Validation Attributes

22.4. Model State

22.5. Validation Attributes

22.6. Validation Tag Helper

22.7. Unobtrusive client-side validation

CHƯƠNG XXIII. CƠ CHẾ DEPENDENCY INJECTION

23.1. Giới thiệu về Dependency Injection

23.2. Depedency Injection là gì?

23.3. DI Container

23.4. Ví dụ về hệ thống Dependency Injection

23.5. Constructor Injection và Action Injection

23.6. Lợi ích của Dependency Injection

23.7. Vòng đời của Dependency Injection: Transient, Singleton và Scoped