

Introduction to ASP.NET Core (MVC)

ASP.NET Core, Controllers & Actions, Routing, Razor, Identity



SoftUni Team
Technical Trainers



SoftUni

Software University

<https://softuni.bg>

1. ASP.NET MVC Overview
2. Creating an ASP.NET MVC App
 - Controllers
 - Views
3. ASP.NET Controllers
4. Razor View Engine

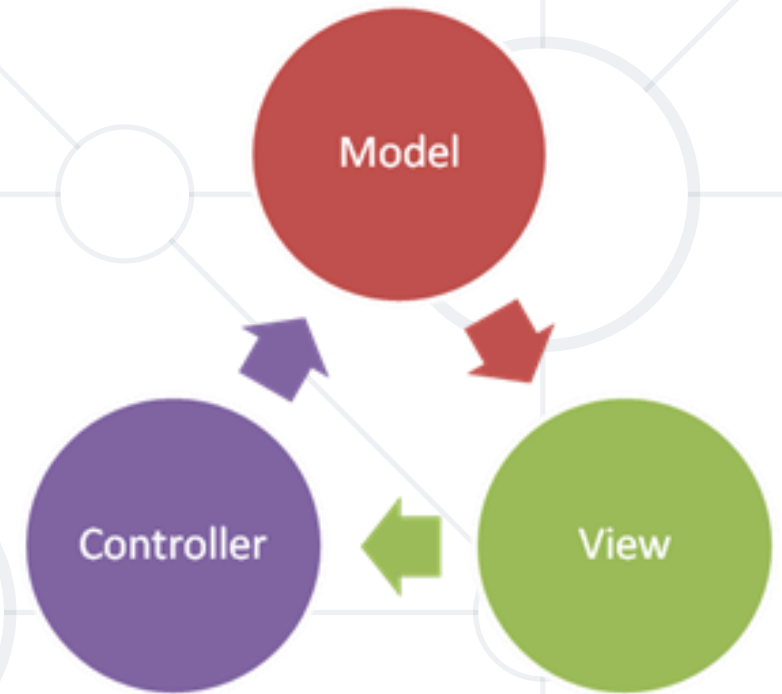


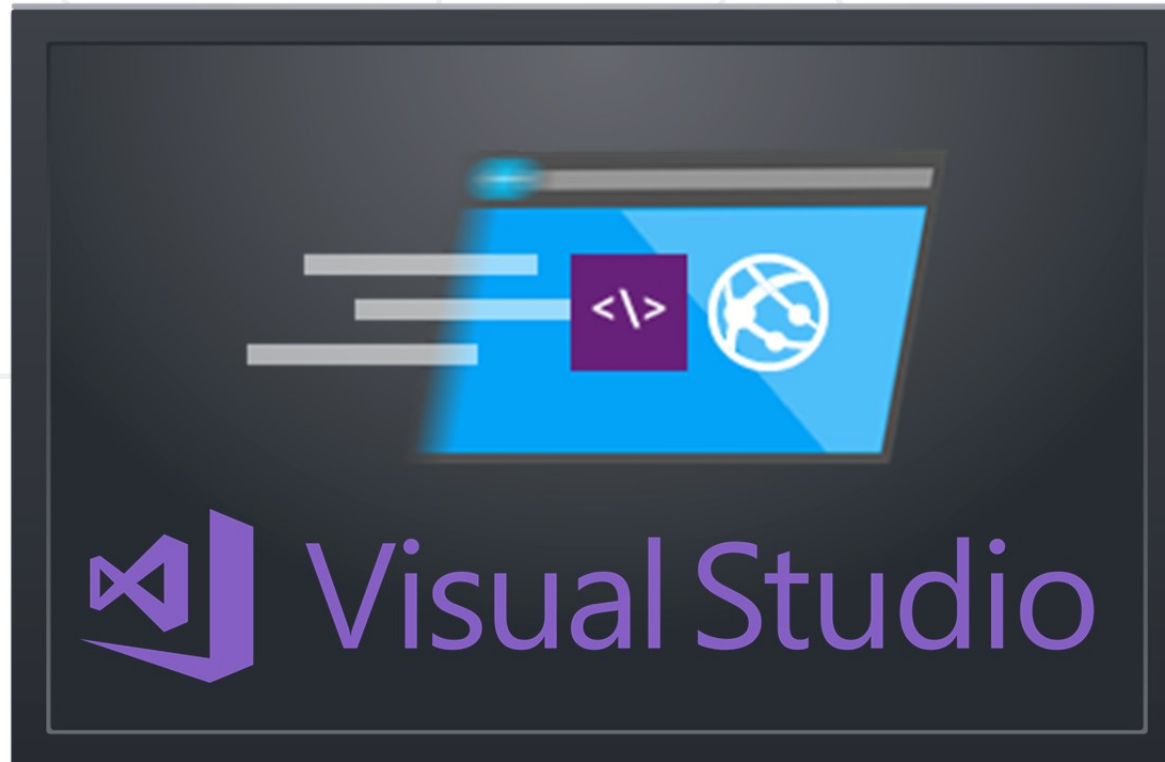


ASP.NET MVC Overview

ASP.NET MVC, Razor

- Open Source Web Application MVC framework
- Developed by Microsoft
- **Code** and **markup** are separated
- Based on **.NET Framework / .NET Core**
 - Develop Web apps using **C#** and use all of its features and **.NET APIs**
- Often combined with **Entity Framework** for **ORM**
- Typically uses **Razor** as a **view engine** (templating engine)





Creating an ASP.NET MVC App

Project Setup in Visual Studio. What's Inside?

Create ASP.NET MVC App: Project Type

Web & Cloud (7)



ASP.NET and web development

Build web applications using ASP.NET, ASP.NET Core, HTML/JavaScript, and Containers including Docker support.



Install this in
VS 2019!

Create a new project

ASP.NET Core Application



Clear all

Recent project templates



NUnit Test Project (.NET Core)

C#



Console App (.NET Core)

C#



ASP.NET Core Web Application

New

Project templates for creating ASP.NET Core web apps and web APIs for Windows, Linux and macOS using .NET Core or .NET Framework. Create web apps with Razor Pages, MVC, or Single Page Apps (SPA) using Angular, React, or React + Redux.

C#

Linux

macOS

Windows

Cloud

Service

Web

Create ASP.NET MVC App: Choose Template

Create a new ASP.NET Core web application

.NET Core

ASP.NET Core 3.1



Empty

An empty project template for creating an ASP.NET Core application. This template does not have any content in it.



API

A project template for creating an ASP.NET Core application with an example Controller for a RESTful HTTP service. This template can also be used for ASP.NET Core MVC Views and Controllers.



Web Application

A project template for creating an ASP.NET Core application with example ASP.NET Razor Pages content.



Web Application (Model-View-Controller)

A project template for creating an ASP.NET Core application with example ASP.NET Core MVC Views and Controllers. This template can also be used for RESTful HTTP services.

Authentication

No Authentication

[Change](#)

Advanced

☒ Configure for HTTPS

☐ Enable Docker Support
(Requires [Docker Desktop](#))

Linux

☐ Enable Razor runtime compilation

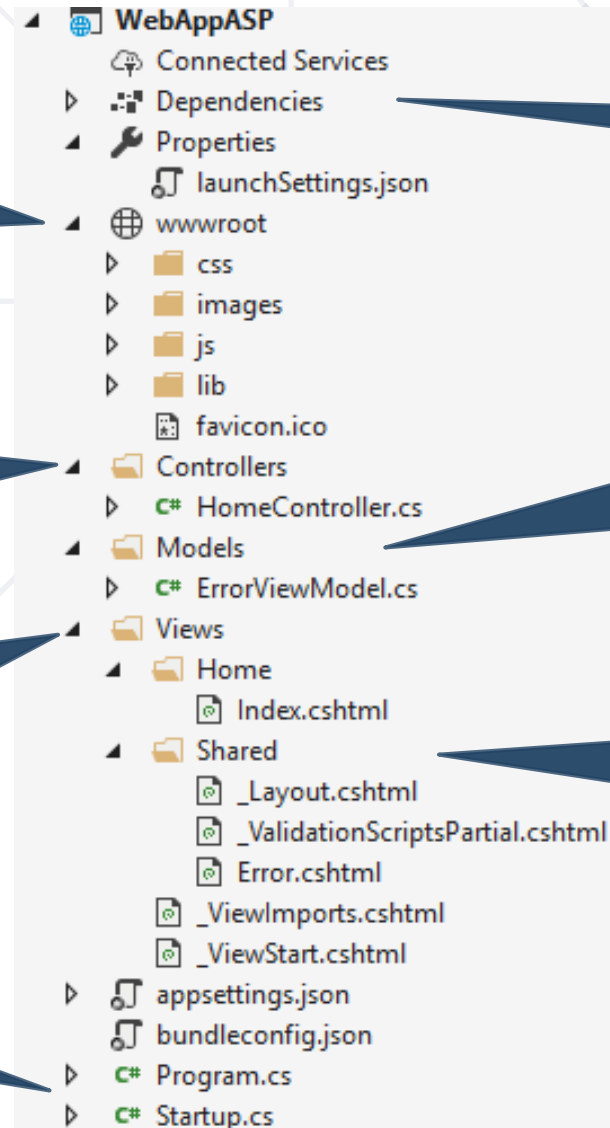
MVC App: What's Inside?

Static files:
CSS styles images,
fonts, ...

Controller classes
holding actions

Views:
HTML templates
for the pages

App start files



NuGet packages

Models: EF classes
+ view models

Shared views:
layout for all pages
+ partial views

- MVC controllers hold logic to process user actions
- The URL **/Home/About** invokes **HomeController → About()**

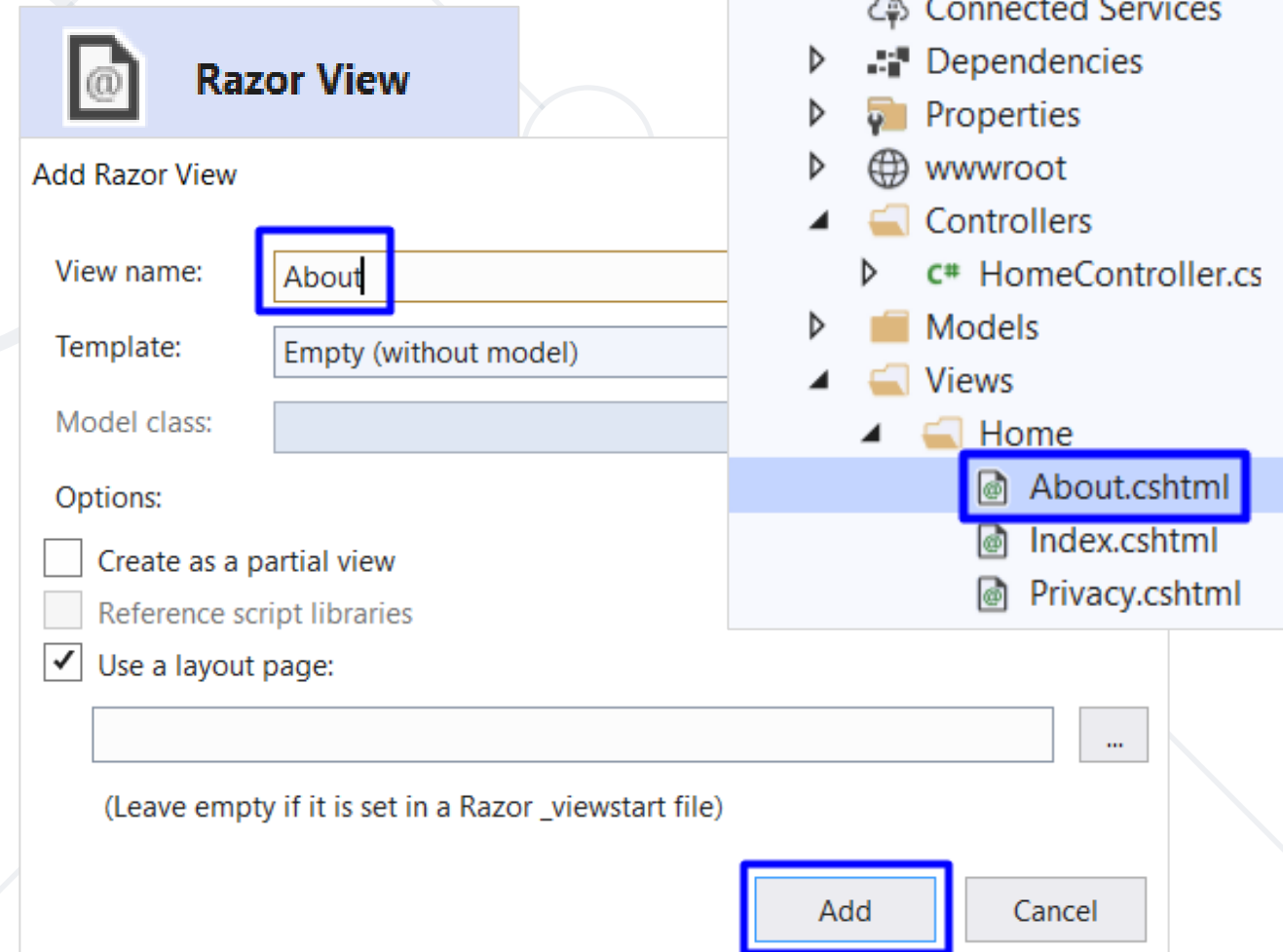
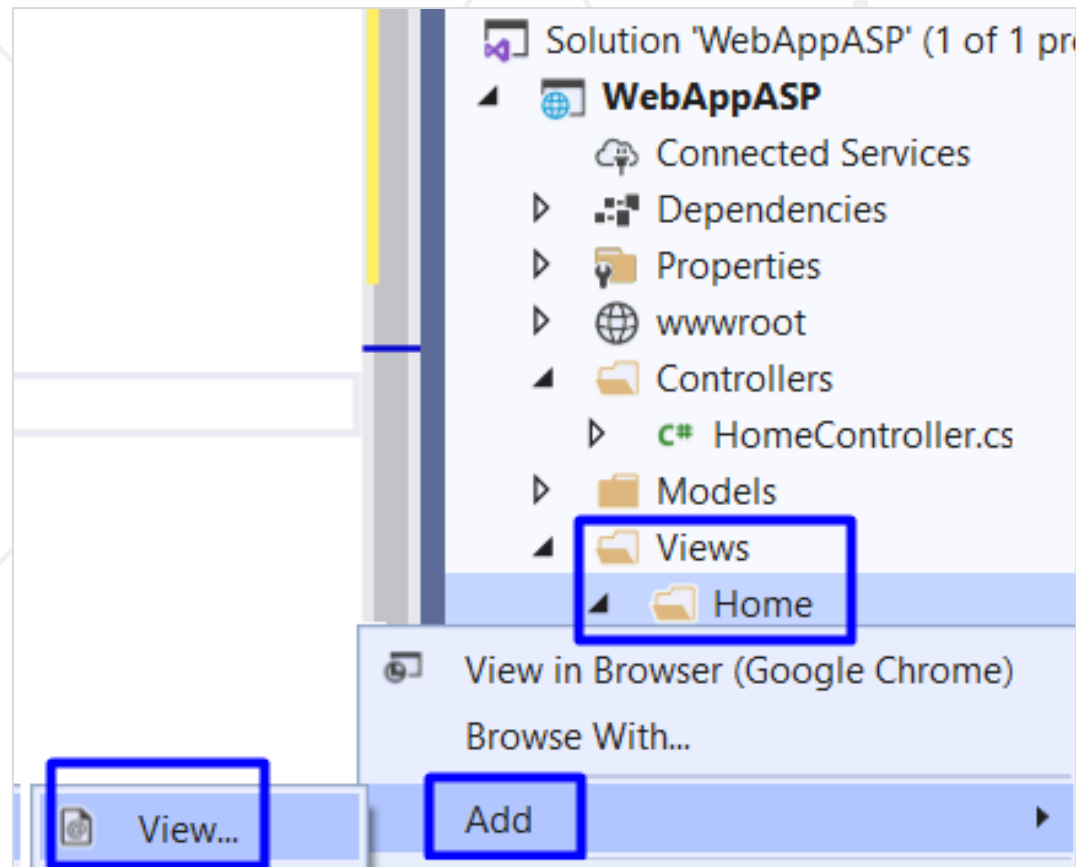
\Controllers\HomeController.cs

```
public class HomeController : Controller
{
    public ActionResult About()
    {
        ViewBag.Message = "Your application description page.";
        return View();
    }
}
```

Renders

Views\Home\About.cshtml

- **Views** render the **HTML code** for the invoked action
- Create **About.cshtml** view



- ASP.NET MVC uses Razor view engine
- Views combine HTML with C# code

\Views\Home>About.cshtml

```
@{  
    ViewBag.Title = "About";  
}
```

@ { ... } inserts C# code block

```
<h2>@ViewBag.Title</h2>  
<h3>@ViewBag.Message</h3>
```

@Something
prints a C# variable

Everything else
is **HTML** code

```
<p>Use this area to provide additional information.</p>
```

Example: Print the Numbers 1...50

- Create an action **/Home/Numbers** + view **Numbers.cshtml**

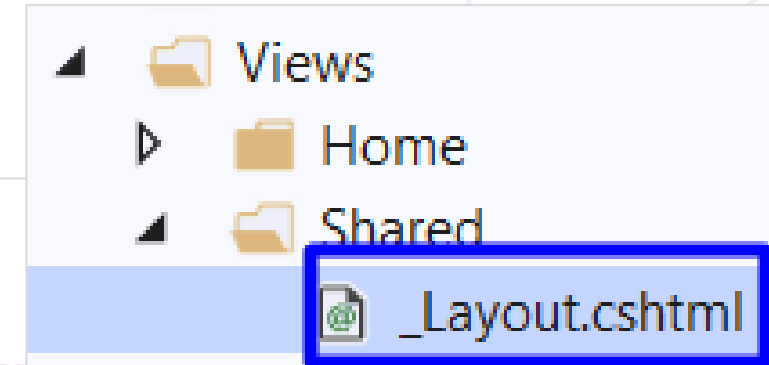
HomeController.cs

```
public class HomeController :  
Controller  
{  
    public ActionResult Numbers()  
    {  
        return View();  
    }  
}
```

\Views\Home\Numbers.cshtml

```
@{ ViewBag.Title = "Nums 1 ... 50"; }  
<h2>@ViewBag.Title</h2>  
<ul>  
    @for (int i = 1; i <= 50; i++)  
    {  
        <li>@i</li>  
    }  
</ul>
```

Changing the Page Layout



```
<div class="navbar-collapse collapse d-sm-inline-flex flex-sm-row-reverse">
  <ul class="navbar-nav flex-grow-1">
    <li class="nav-item">
      <a class="nav-link text-dark" asp-area="" asp-controller="Home" asp-action="Index">Home</a>
    </li>
    <li class="nav-item">
      <a class="nav-link text-dark" asp-area="" asp-controller="Home" asp-action="Privacy">Privacy</a>
    </li>
    <li class="nav-item">
      <a class="nav-link text-dark" asp-area="" asp-controller="Home" asp-action="About">About</a>
    </li>
    <li class="nav-item">
      <a class="nav-link text-dark" asp-area="" asp-controller="Home" asp-action="Numbers">Nums 1...50</a>
    </li>
  </ul>
</div>
```

Checking Result

- Press **[Ctrl+F5]**
- You will see your app's **home page**
- Go to the **newly created link** and check result

WebAppASP Home Privacy About Nums 1...50

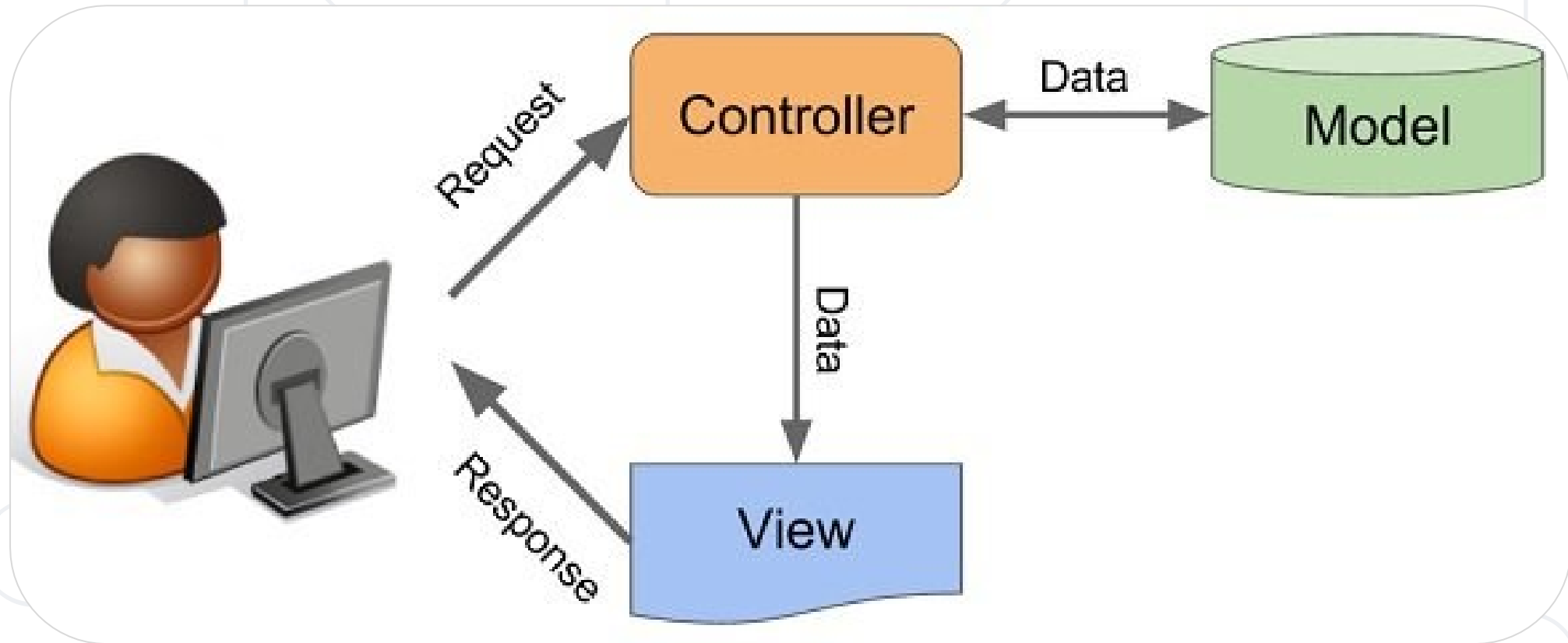
Welcome

Learn about [building Web apps with ASP.NET Core](#).

WebAppASP Home Privacy About **Nums 1...50**

Nums 1 ... 50

- 1
- 2
- 3
- 4
- 5



ASP.NET Controllers

- **Controllers** hold multiple actions on different **routes**
- Route configuration is defined in **RouteConfig.cs**
- Routes are **automatically generated**, based on **RouteConfig**

```
public class ArticleController : Controller
{
    public IActionResult List(int id)
    {
        // Code comes here ...
    }
}
```

Action parameters

Generated route:
/Article/List/{id}

Example: Print the Numbers 1...N

HomeController.cs

```
public class HomeController
    : Controller
{
    public ActionResult
        Numbers(int count = 5)
    {
        ViewBag.Count = count;
        return View();
    }
}
```

Nums 1 ... 3

- 1
- 2
- 3

\Views\Home\Numbers.cshtml

```
@{ ViewBag.Title =
    "Nums 1 ... " + ViewBag.Count; }
<h2>@ViewBag.Title</h2>
<ul>
    @for (int i = 1; i <= ViewBag.Count; i++)
    { <li>@i</li> }
</ul>
@using (Html.BeginForm())
{
    @Html.TextBox("count");
    <input type="submit" />
}
```

- Annotated with `[Http{method}]`

```
[HttpGet]  
public IActionResult Index() {  
    return View();  
}
```

```
[HttpPost]  
public IActionResult Create()  
{  
    ...  
}
```

- Create **controller action**, which processes a HTTP **GET** request:

```
[HttpGet]  
public ActionResult Index()  
{  
    return View();  
}
```

- Returns the view in **Views/{controller}/{action}.cshtml**

- Similar to **[HttpGet]**, there is also an alias for method **POST**:

```
[HttpPost]  
public IActionResult Register()  
{  
    ...  
}
```

- Similar attributes exist for **all** other types of **request** methods

```
<ul id="products">  
  @foreach (var p in products)  
  {  
    <li>@p.Name ($@p.Price)</li>  
  }  
</ul>
```


Razor View Engine

What is it? How do I use it?

What is Razor?

- Simple-syntax **view engine**
- **Code-focused** templating approach
- Easy transition between HTML and code
- Examples: combining **HTML** and **C#**:

```
1 <!DOCTYPE html>
2 <html>
3   <head>
4     <title>Example</title>
5     <link rel="stylesheet" href="style.css">
6   </head>
7   <body>
8     <div id="header">
9       <a href="/">Header</a>
10    </div>
11    <nav>
12      <a href="one/">One</a>
13      <a href="two/">Two</a>
14      <a href="three/">Three</a>
15    </nav>
```



```
<div class="article">
  <div>@article.Title</div>
  <div>@article.Content</div>
</div>
```

```
<ul id="products">
  @foreach (var p in products)
  {
    <li>@p.Name ($@p.Price)</li>
  }
</ul>
```

Razor View Engine: Example

- HTML mixed with C# code (@ switches to C#):

```
<div class="row">
  @foreach(var article in Model)
  {
    <article>
      <h2>@article.Title</h2>
      <p>@article.Content</p>
      <small>--@article.Author.FullName</small>
    </article>
  }
</div>
```

HTML
Syntax

C# code

C# foreach

C# code

- We can use **HTML helpers** in Razor to generate HTML:

```
@Html.ActionLink(  
    article.Title,  
    "Details",  
    "Article",  
    new {  
        id = article.Id  
    },  
    new {  
        @class = "text-uppercase"  
    }  
)
```

Link text

Action

Controller

Controller
Route Values

HTML
Attributes

HTML Result

```
<a class="text-uppercase"  
    href="/Article/Details/2">  
    Article 2  
</a>
```

ARTICLE 2

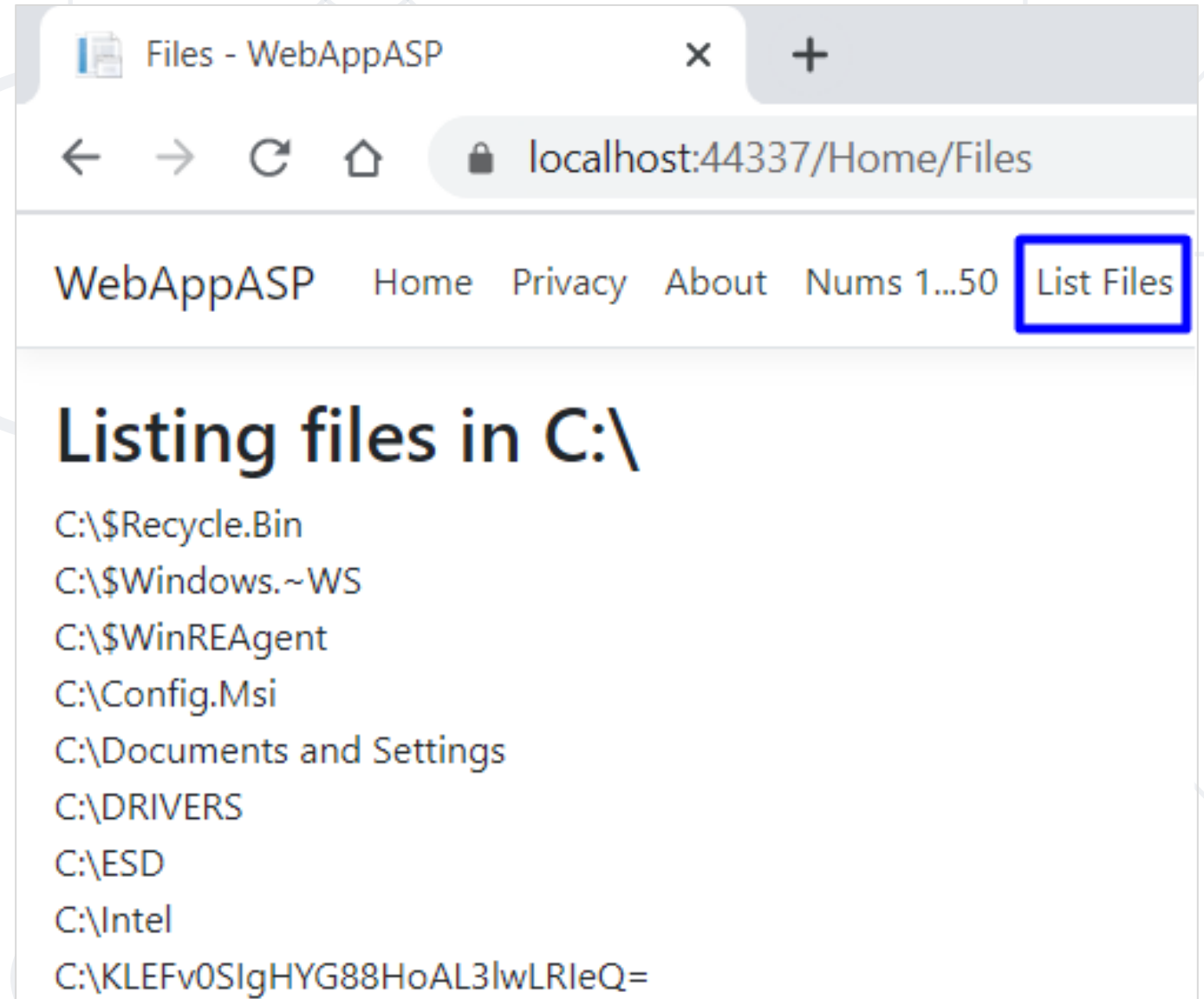
Article Content

--author Gosho

localhost:51209/Article/Details/2

Example: List Files

- Create an action **Files** to show the folders and files in **C:**



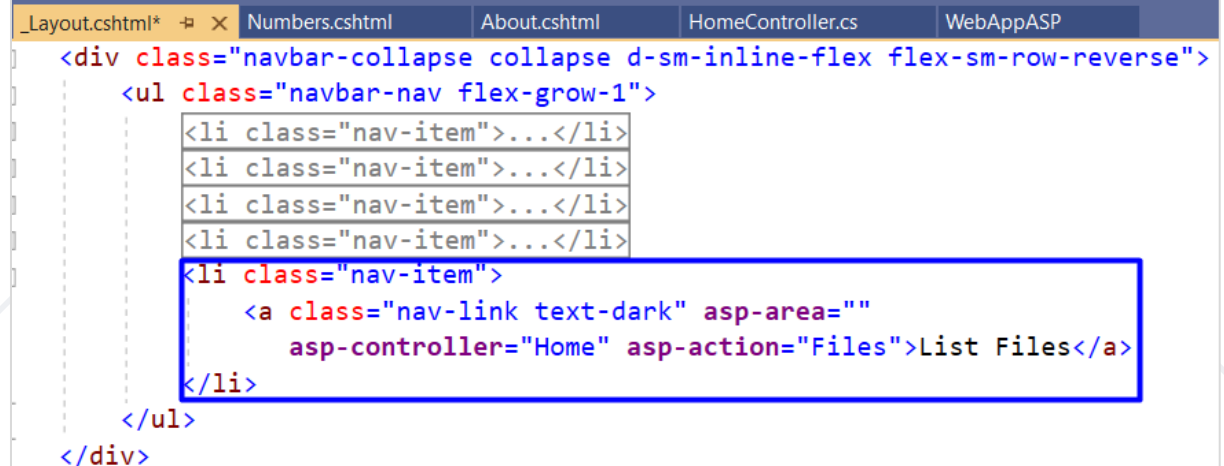
Solution: List Files

HomeController.cs

```
public ActionResult Files()  
{  
    var path = @"C:\";  
    var files = Directory  
        .GetDirectories(path)  
        .ToList();  
    files.AddRange(  
        Directory.GetFiles(path));  
    return View(files);  
}
```

\Views\Home\Files.cshtml

```
@{ ViewBag.Title = "Files"; }  
<h2>Listing files in C:\</h2>  
@model List<string>  
@foreach (var file in Model)  
{  
    <div> @file </div>  
}
```



```
<div class="navbar-collapse collapse d-sm-inline-flex flex-sm-row-reverse">  
    <ul class="navbar-nav flex-grow-1">  
        <li class="nav-item">...</li>  
        <li class="nav-item">...</li>  
        <li class="nav-item">...</li>  
        <li class="nav-item">...</li>  
        <li class="nav-item">  
            <a class="nav-link text-dark" asp-area=""  
                asp-controller="Home" asp-action="Files">List Files</a>  
        </li>  
    </ul>  
</div>
```

- **ASP.NET MVC** is powerful Web dev platform

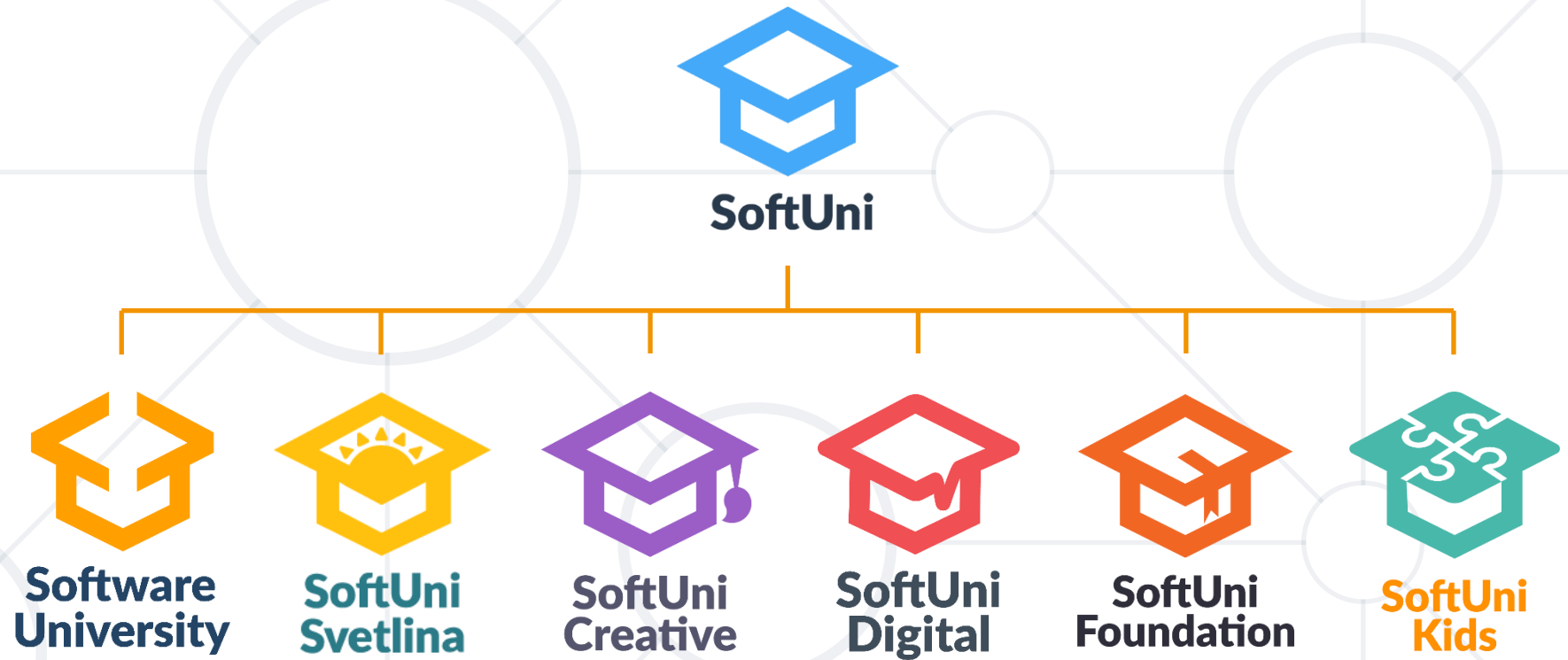
- **Views** render HTML code

```
@foreach (var item in @Model)
{ <li>@item</li> }
```

- **Controllers** process HTTP GET / POST actions

```
public ActionResult Index()
{ return this.View(GetAllItems()); }
```

Questions?



- This course (slides, examples, demos, exercises, homework, documents, videos and other assets) is **copyrighted content**
- Unauthorized copy, reproduction or use is illegal
- © SoftUni – <https://softuni.org>
- © Software University – <https://softuni.bg>



- Software University – High-Quality Education, Profession and Job for Software Developers

- softuni.bg, softuni.org

- Software University Foundation

- softuni.foundation

- Software University @ Facebook

- facebook.com/SoftwareUniversity

- Software University Forums

- forum.softuni.bg



Software University

