



C# Corner
Kolkata Chapter



@csharpcorner



<https://www.facebook.com/CsharpCorner>

Deep Drive Into ASP.Net Core 2.1 (Day 1)

Presented by
DEBASIS SAHA



C# Corner MVP

| Team Lead | Author | Blogger | Speaker



debasis.ds@outlook.com



[@debasiskolsaha](https://twitter.com/debasiskolsaha)



<https://github.com/debasis-saha>

Events Topics

- ☐ Introduction to ASP.Net Core
- ☐ Difference Between Asp.Net MVC & .Net Core
- ☐ Understanding the Project Structure
- ☐ Concept of .Net CLI With Templating
- ☐ Create Console Applications
- ☐ Create Simple Web Application
- ☐ Dynamic HTML Rendering
- ☐ Configuration
- ☐ Q n A

History of ASP.Net Core 2.1

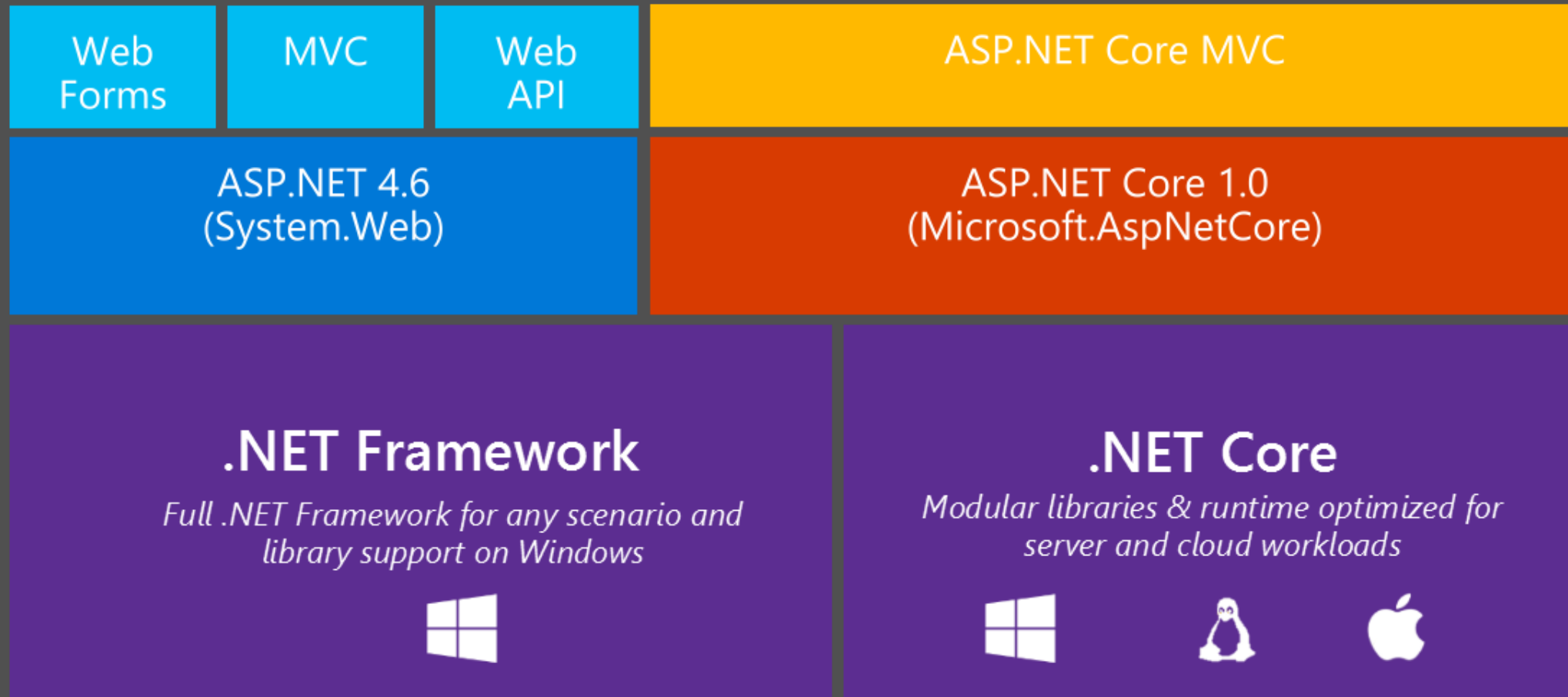
ASP.NET CORE RELEASE HISTORY



FEATURES OF ASP.NET CORE

- ☐ Cross Platform Support
- ☐ Open Source Language
- ☐ Modular Web Packages
- ☐ High Performance
- ☐ Unified MVC & Web API Frameworks
- ☐ Web Sockets & SignalR
- ☐ Can be used within Visual Code or Visual Studio

ASP.NET Core in a Nutshell



Install .Net Core

How To Install .Net Core



Download .NET

Overview

[.NET Core](#)

[.NET Framework](#)

[Xamarin](#)

[.NET SDKs](#)



.NET Framework

The .NET framework helps you create mobile, desktop, and web applications that run on Windows PCs, devices and servers.



.NET Core

.NET Core gives you a blazing fast and modular platform for creating server applications that run on Windows, Linux and Mac.



Xamarin

Xamarin brings the power and productivity of .NET to iOS and Android, reusing skills and code while getting access to the native APIs and performance.

[Help me choose >](#)

Install Using Visual Studio 2017

Visual Studio Installer

Modifying — Visual Studio Community 2017 — 15.8.5

Workloads

Individual components

Language packs

Installation locations

Windows (3)



.NET desktop development

Build WPF, Windows Forms, and console applications using C#, Visual Basic, and F#.



Universal Windows Platform development

Create applications for the Universal Windows Platform with C#, VB, JavaScript, or optionally C++.



Desktop development with C++

Build Windows desktop applications using the Microsoft C++ toolset, ATL, or MFC.



Web & Cloud (7)



ASP.NET and web development

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



Azure development

Azure SDKs, tools, and projects for developing cloud apps, creating resources, and building Containers including...



Python development

Editing, debugging, interactive development and source control for Python.



Node.js development

Build scalable network applications using Node.js, an asynchronous event-driven JavaScript runtime.



Location

C:\Program Files (x86)\Microsoft Visual Studio\2017\Community [Change...](#)

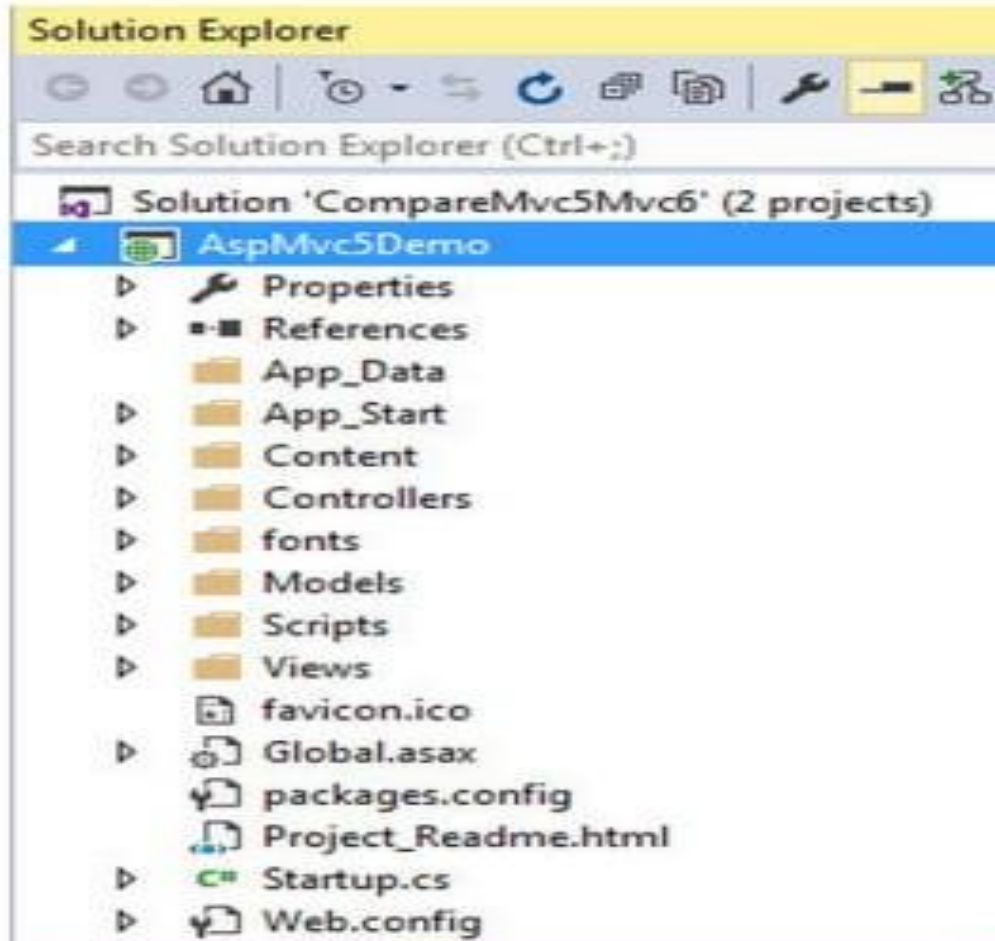
Difference Between ASP.Net MVC Vs ASP.Net Core

1. Cross Platform Supports

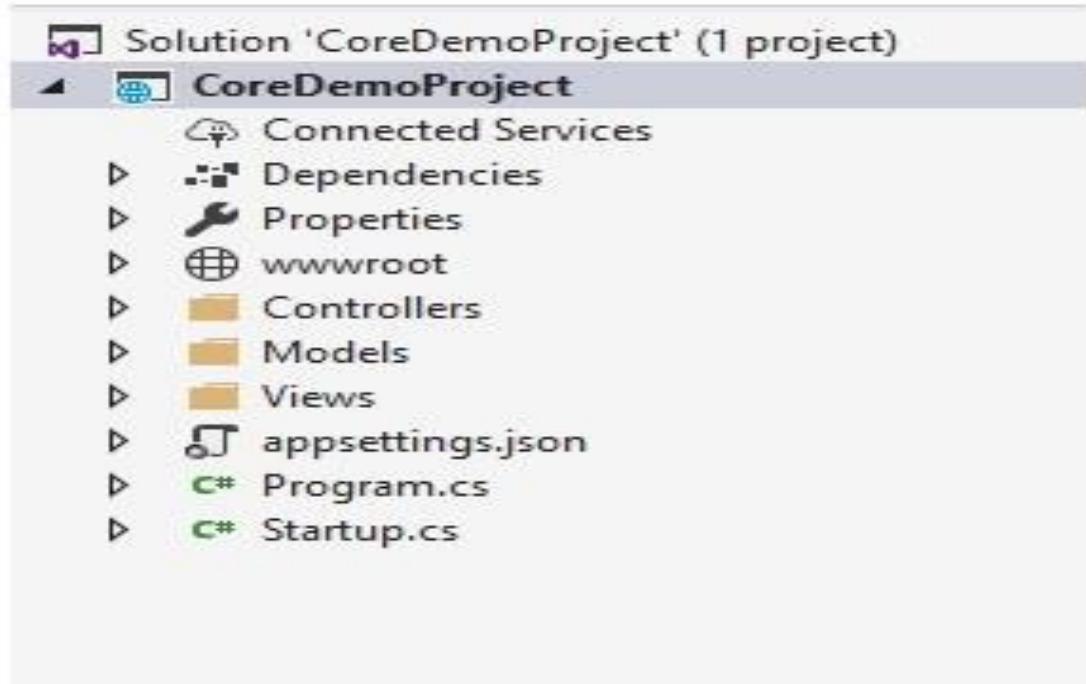
❑ ASP.NET Core support Cross Platform OS



2. Project Structure Changed



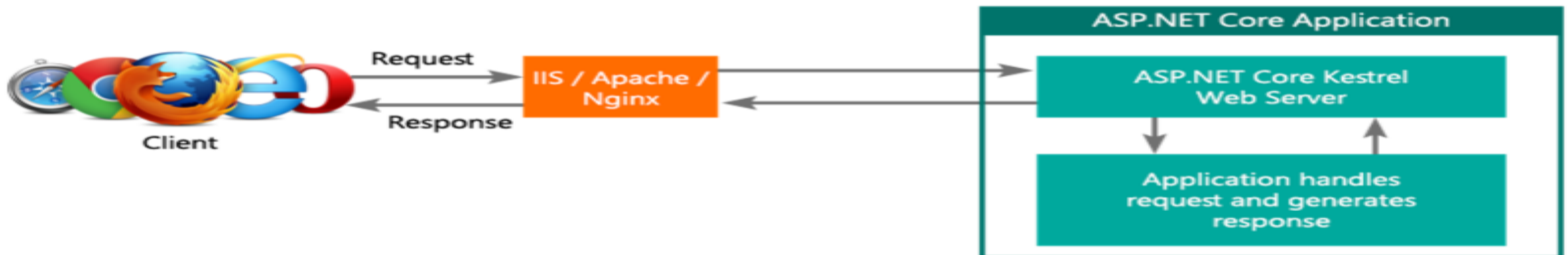
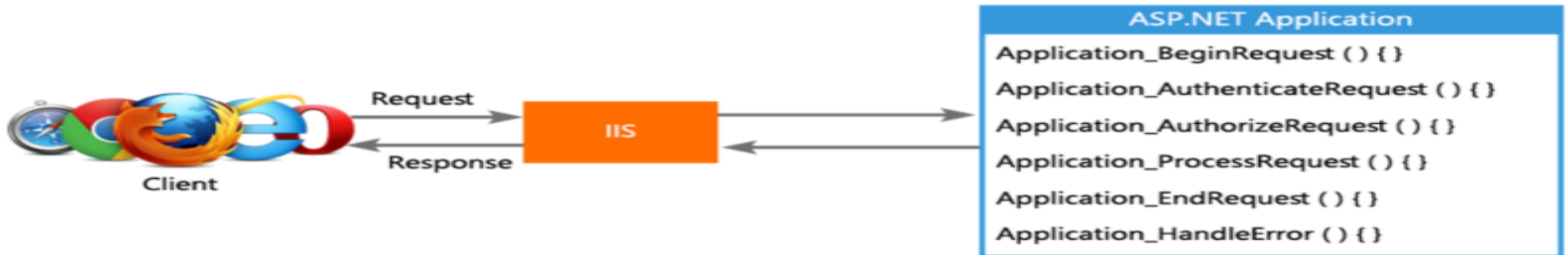
ASP.NET MVC 5 project structure



.Net Core Project Structure

3. Does not Depends on IIS For Hosting

- ❑ Can be Host in Any Platform
- ❑ Nginx or Apache or Docker



4. Installation Process of Applications

- ☐ Support multiple packages
- ☐ Code need to compile along with Nuget Packages
- ☐ Can download packages from NPM or Bower

5. Mobile App Development

- ☐ Can develop Mobile Based Applications
- ☐ Can Use Xamarin for Mobile App Development
- ☐ Can create Shared Layer for all types of Mobile Applications
- ☐ Xamarin can develop Android, IOS and Windows Based Applications

6. Microservices

- ☐ **Support Microservices Oriented System**
- ☐ **It is a Software Development Technique**
- ☐ **A Variant of Service-Oriented Architecture(SOA)**
- ☐ **Collections of a Loosely coupled services.**

7. Performance Improvements

- ❑ .Net Core provide Performance Boost in the Application
- ❑ It provide direct deployment in Cloud Platform



Request / second (millions) based on official tests available at [TechEmpower Round 14](#).

.Net CLI With Templating

How to Use CLI for .Net Core



- ❑ CLI Means Command Line Interface
- ❑ Can create .Net Core Projects using CLI Command
- ❑ Command called Template Engine to create project structure on the disk
- ❑ `dotnet new <TEMPLATE> [--force] [-i|--install] [-lang|--language] [-n|--name] [--nuget-source] [-o|--output]`
- ❑ `dotnet new [-h|--help]`

Popular Templates in .Net CLI

Template Name	Template Command	Languages Support
Console Application	console	[C#], F#, VB
Class Library	classlib	[C#], F#, VB
Asp.Net Core Empty	web	[C#], F#
Asp.Net Core Web App (MVC)	mvc	[C#], F#
Asp.Net Core Web App	razor	[C#], F#
Asp.Net Core with Angular	angular	[C#]
Asp.Net Core Web API	webapi	[C#], F#
Web Config	webconfig	

Create Projects Using Template Commands



Create Console Applications

dotnet new console -lang "F#"

**Create a new ASP.NET Core C# MVC application
project in the current directory with no
authentication:**

dotnet new mvc -au None

DEMOSTRATIONS

Create Console Applications

DEMOSTATIONS

Create Simple Web Applications

Dynamic HTML Rendering

- ❑ **Dynamic content in the context of HTML and the World Wide Web refers to website content that constantly or regularly changes based on user interactions, timing and other parameters that determine what content is delivered to the user.**

Configurations

- ☐ Store Application Levels Configurations
- ☐ Standard file Name convention is appsettings.json
- ☐ It store key-value pair based json objects related to configurations.
- ☐ It can be .json or .ini or .xml files

THANK YOU For Attending the Session

Event Related Documents is also available at –

- **Presentations & Sample Code :- [GitHub Code Repository](#)**
- **Events Recerded Video – [Kolkata Chapter Channel](#)**