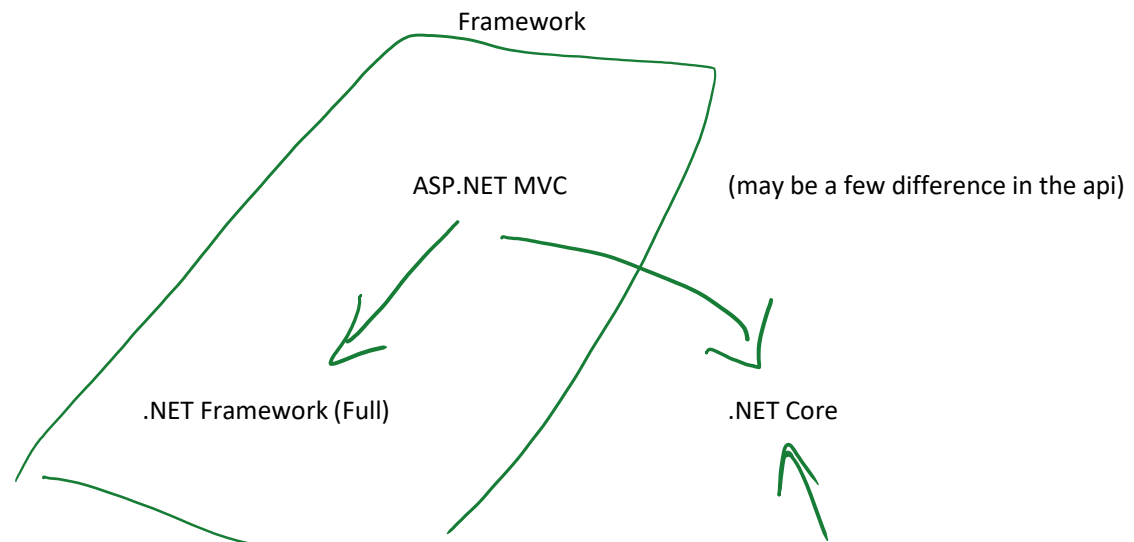


# Agenda

Monday, May 11, 2020 9:35 AM

**ASP.NET MVC <--- Application Framework <---- Helps you develop the application**

**.NET Core Application <----- Runtime Framework/Platform <----- helps you execute the application**



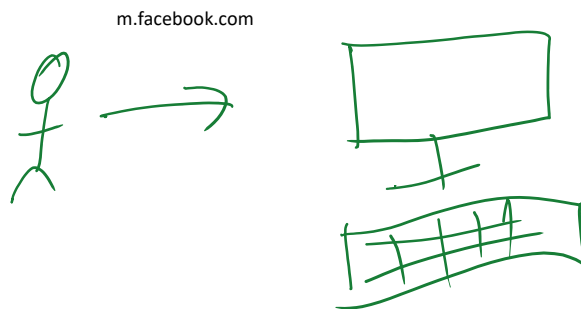
# Http Protocol

Monday, May 11, 2020 10:03 AM

<http://m.facebook.com> <---- Open Mobile Site

<http://www.ircctc.co.in/mobile>

GET m.facebook.com/\r\n  
User-Agent: chrome, windows10\r\n



It is a contradiction between  
the URL and the User-Agent

Url --> give me mobile site  
User-Agent --> give me desktop site



GET /some/url  
**connection: keep-alive**

- Request to keep the connection alive
- Don't close the Tcp socket (Please)
- Browsers almost always send this header to the client
- Server can (and very often does) ignore this request

Server Response

<https://books.org/list?cat=thriller>

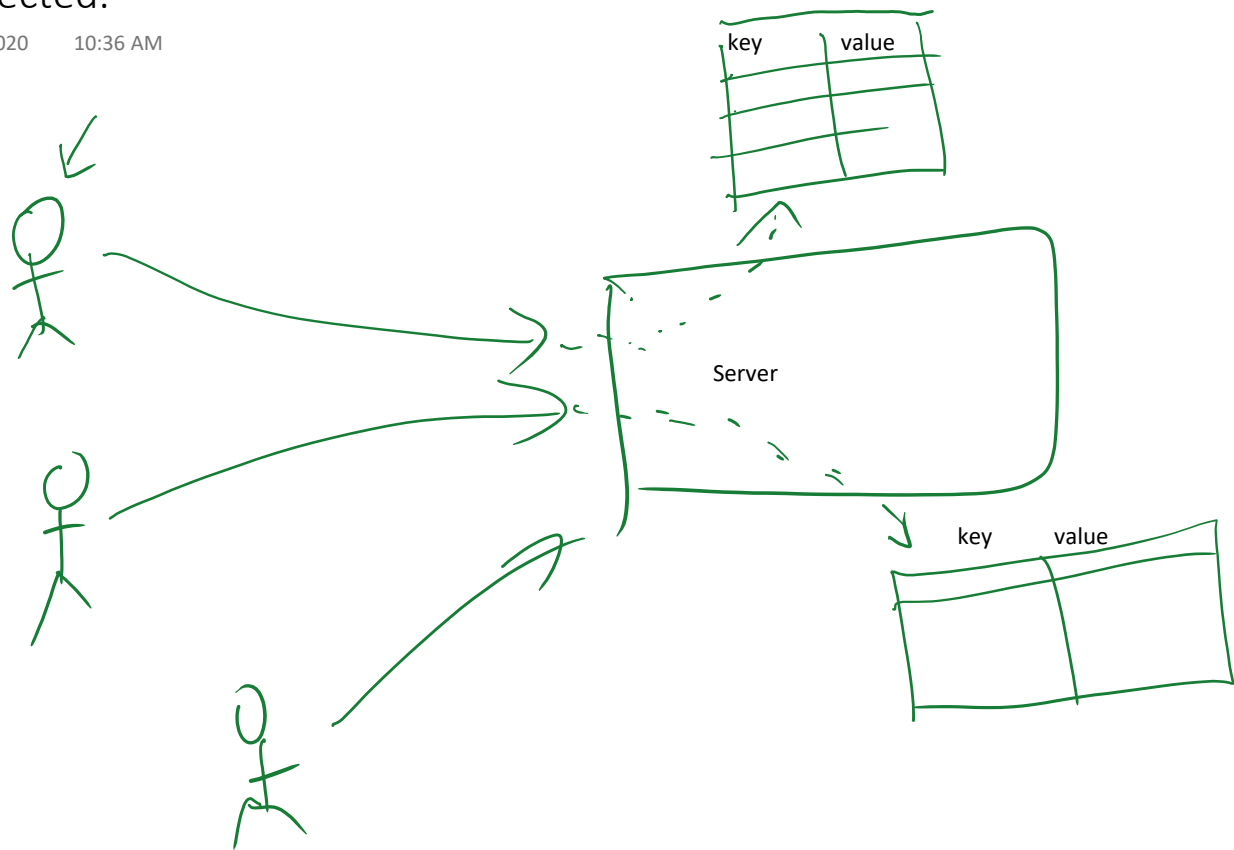
<https://amazon.in/proucts?q=mobile-cover&model=samung-note-8>

200 OK  
Content-type:text/html  
**date-modified:2020/04/13**

Clients second request

How will server connect a user to its session when socket is disconnected.

Monday, May 11, 2020 10:36 AM



# Cache Redirect

Monday, May 11, 2020 11:28 AM

## Server Response

200 OK  
Content-type:text/html  
**date-modified:2020/04/13**

## Client Next Response

GET /resource  
**if-modified-since:2020/01/30**

//content changed after clients previous visit

200 OK  
Content-type:text/html  
**date-modified:2020/04/13**

**Gets new Data**

## Client Next Response

GET /resource  
**if-modified-since:2020/05/04**

//content not modified after the last visit

**304 Not Modified**  
\r\n

Cache Redirect

Can use the existing

# MVC

Monday, May 11, 2020 12:19 PM

Presentation

Business

Data

View

~~Controller~~

Model

Model



# Standard Network Port

Tuesday, May 12, 2020 9:26 AM

As an admin I can run any server on any available (free) port.

One Machine ----> One Port ----> One App (One socket)

Standard Practice

First 1K port is reserved for standard protocol

- 20,21 ----> FTP
- 25 ----> SMTP
- 80 ----> HTTP
- 110 ----> POP
- 443 ----> HTTPS

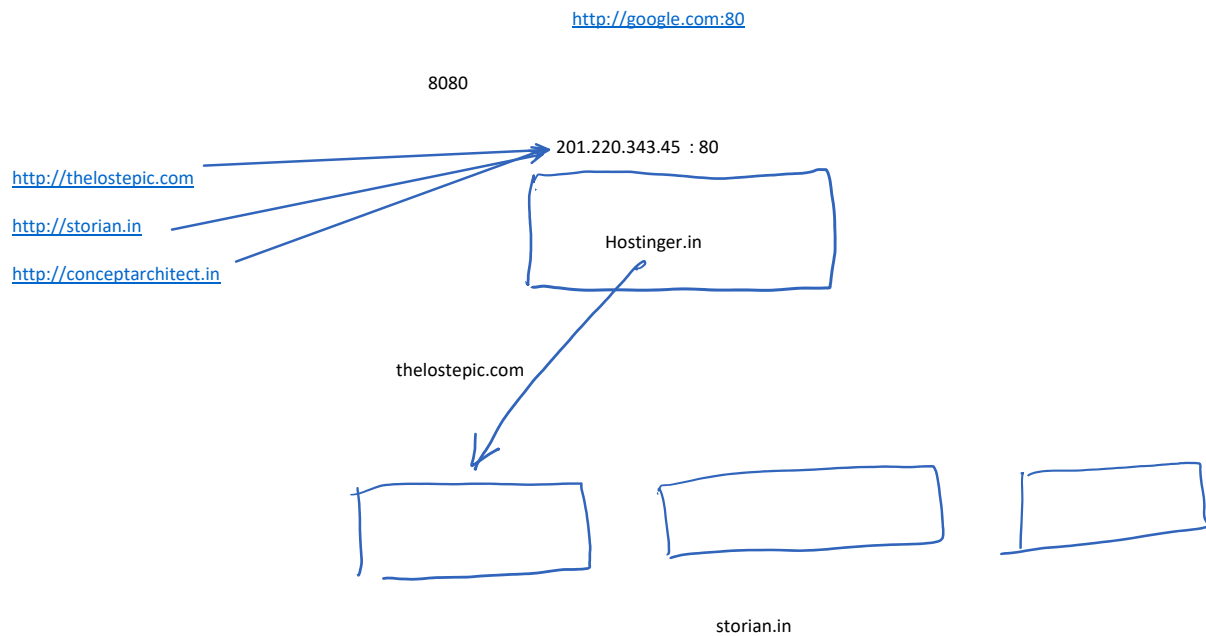
Q. What if I run HTTP Server on a Non Standard Port

A. The URL Will Change

a. <http://books.org:3393/books>

- now we say the server will be available on port 3393.

- You can imagine that default url is a shortform for
  - <http://books.org:80/books>
  - <http://books.org:445/books>



# MVC URL

Tuesday, May 12, 2020 10:00 AM

## Traditional View

<http://localhost:57586/> <----- index.html/default.html/default.aspx/default.php ---> from root directory

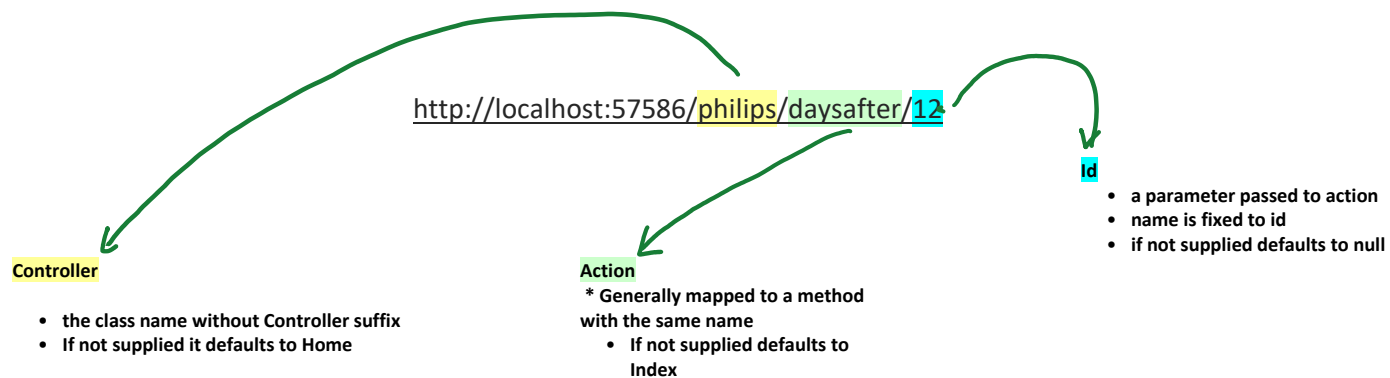
<http://localhost:57586/Home/Contact> <----- index.html/default.html/default.aspx from /Home/Contact directory

<http://localhost:57586/Home/About>

<http://localhost:57586/books.html> <----- request for books.html file in root directory

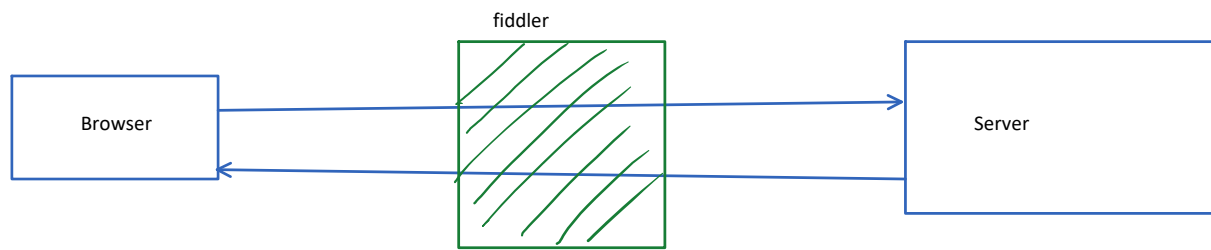
<http://localhost:57586/Home/About/authors.aspx> <----- execute authors.aspx page from /Home/About folder

## MVC URL has by default 3 parts



# How Fiddler Works

Tuesday, May 12, 2020 10:41 AM





# Controller--->Action--->ActionResult

Tuesday, May 12, 2020 11:38 AM

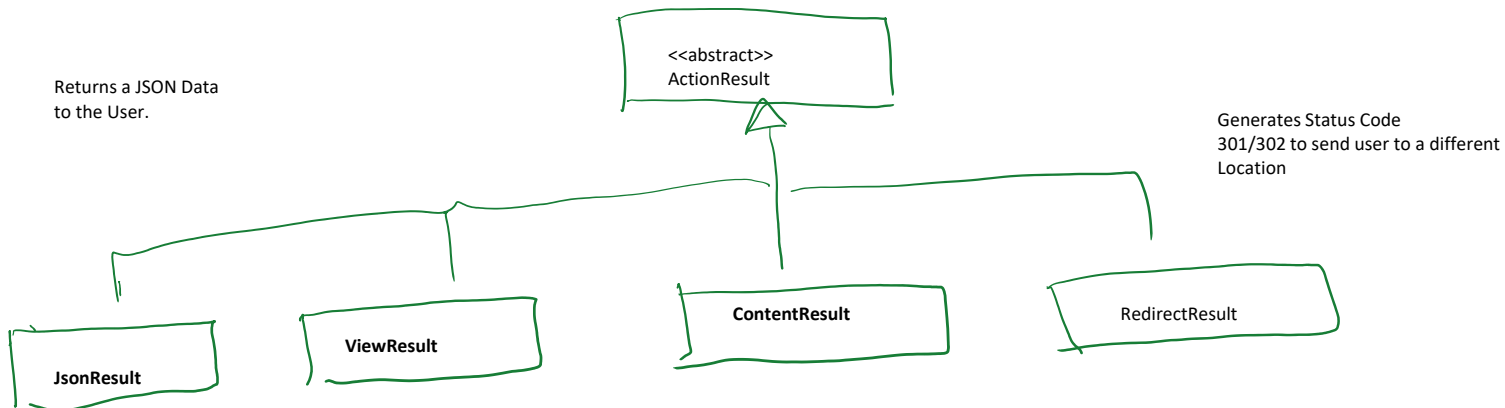
1. An Url is a request for a **Controller** to perform some **Action**
2. Once an Action is performed it returns an **ActionResult**
3. ActionResult is returned to the User As a **View**

**Controller** ----> is an object

**Action** ----> is what controller do

**ActionResult** ----> return of the Action

**View** ----> what user sees (ActionResult)



## ViewResult

ViewResult uses a ViewEngine to generate a dynamic Html View using Model Data and a ViewTemplate

## ContentResult

One of the simplest ActionResult  
It returns a given String content to the user without any processing.

If your **Action** is not returning an **ActionResult**  
The current object is automatically converted to **ContentResult** by the controller

A code like

```
public Person Contact()
{
    var p=new Person();
    return p;
}
```

Is internally converted to (psudocode)

```
public ContentResult Contact(){
    var p=new Person();
    return new ContentResult()
    {
        Content=p.ToString()
    };
}
```

This happens with the base Controller

# Razor view engine processing

Tuesday, May 12, 2020 12:26 PM

Razor is one of the simplest view engine to combine view template (HTML) with C# Code (**CS+HTML = CSHTML**)

```
<h2>Welcome To Philips Server</h2>
@{
    //this is a c# code block. you can write any c# here
    var date = DateTime.Now;
}
<p><strong>Date is </strong>@date.ToLongDateString()</p>
<p><strong>Time is </strong>@date.ToLongTimeString()</p>
```

## Razor Block Code

```
@{
    //razor block
}
```

You can write any processing statement, assignments etc in this block. This block doesn't have direct output on the screen unless you write `Response.Write()`

## Razor Display Expression

`@something`

It is a shortcut for

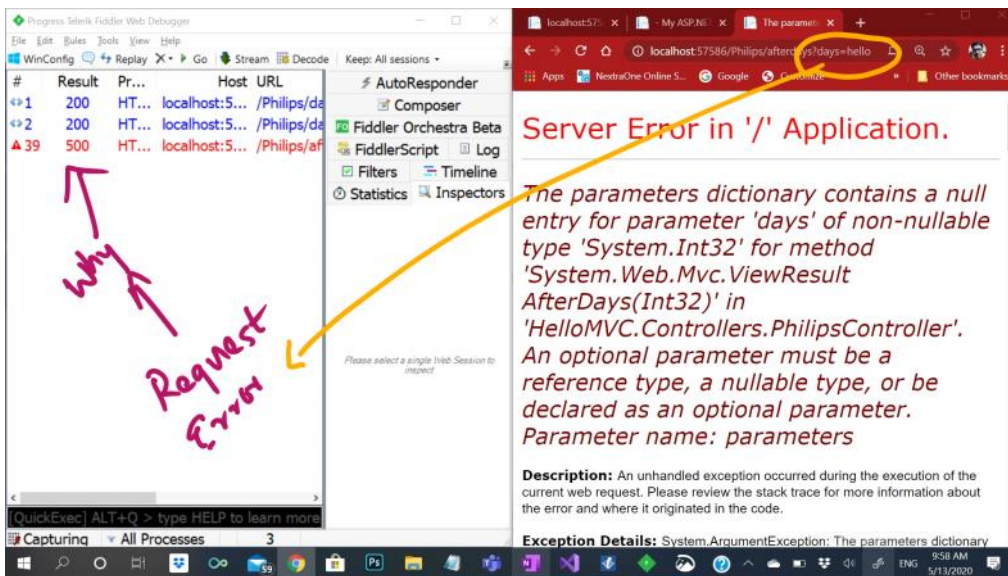
```
@{ Response.Write(something); }
```

Notice there is no semicolon after simple razor expression what ever you type automatically used as argument to `Response.Write()`

Also Note @ is automatically closed when it finds html code

# HTTP Errors!

Wednesday, May 13, 2020 9:56 AM

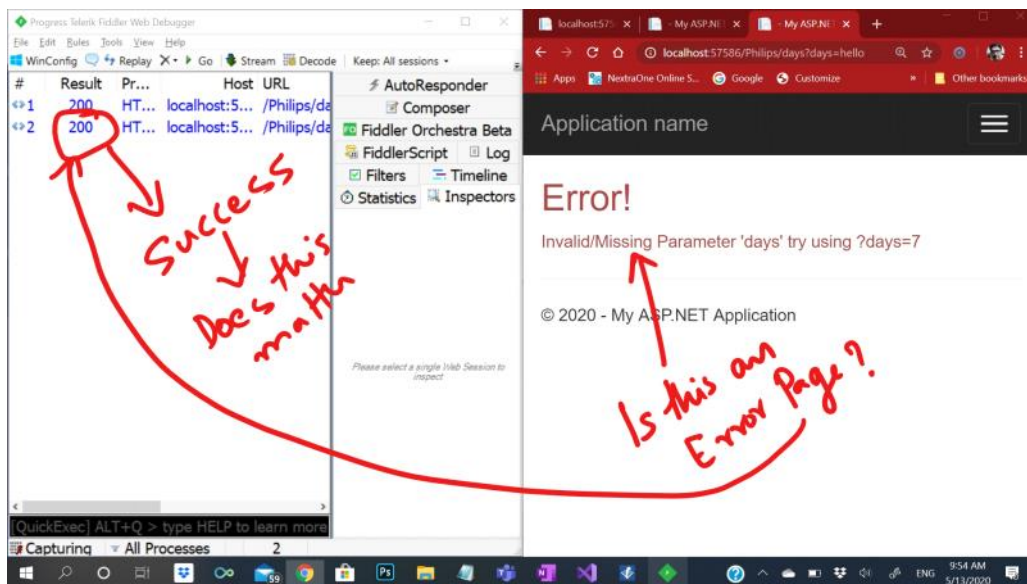


Is this a client request error or a Server Side error as notified.

Ideally it should be request error (400)

But it is server error, because Server didn't tell user they made a mistake

Server is uselessly taking on the guilt!



If This is an error page,

Why is the status code 200?

What should be the status code here?

It should be **400 Bad Request**

Does it matter 200/400? After all end user is going to see Error page and not the actual status code?

YES.  
Humans are not the only one who see a webpage

Automated system like search engine crawlers or automated testing frameworks also see these outputs

- Do you want search engine crawlers to index even error page?
- Do you want automated systems to check color of message to decide if it is an error page?

# http header

Wednesday, May 20, 2020 9:52 AM

**Accept-Language: en-US,en;q=0.9,hi;q=0.8,gu;q=0.7**

- **Comma separates multiple items**
  - semicolon is part of an item that precedes it
  - adds additional information about that item
- **I am willing to accept four languages**
  - en-US
  - en (q=0.9)
  - hi (q=0.8)
  - gu (q=0.7)
- **the meta information varies from elements we are using**
  - q indicates a priority on a scale of 0-1
  - default value of q is 1
- **It says my preference is**
  - en-US > en > hi > gu

# URL Formats

Wednesday, May 13, 2020 10:41 AM

Which is a better URL to get a list of all Books By Jeffery Archer?

Approach#1 -- Traditional with id

<http://books.org/books.aspx?author=2915>

Approach#2 -- Traditional with name

<http://books.org/books.aspx?author=jeffrey-archer>

Approach#3 -- MVC URL

<http://books.org/books/by/jeffrey-archer>

Problem Approach#1

- need to remember an id
- what will be the list of books by john grisham called?

Problem Approach#2

- long and complicated url
- It looks too techy
  - Query Strings are ugly
- **Technology Id (.aspx)**
  - **Why should I include technology id in an URL?**
  - **What Jeffrey Archer book list has got to do with .net framework?**
- **Why Url shouldn't include technology**
  - **because we may switch our technology tomorrow**
  - **Then all urls we created today will be useless**
  - **All bookmarks will be dead**

## W3C Recommendation for a URL

A URL should be

- Short
- Simple (Easy to Remember)
- Permanent (Removed Technology ID)
- identification of a resource (business)

URL shouldn't include

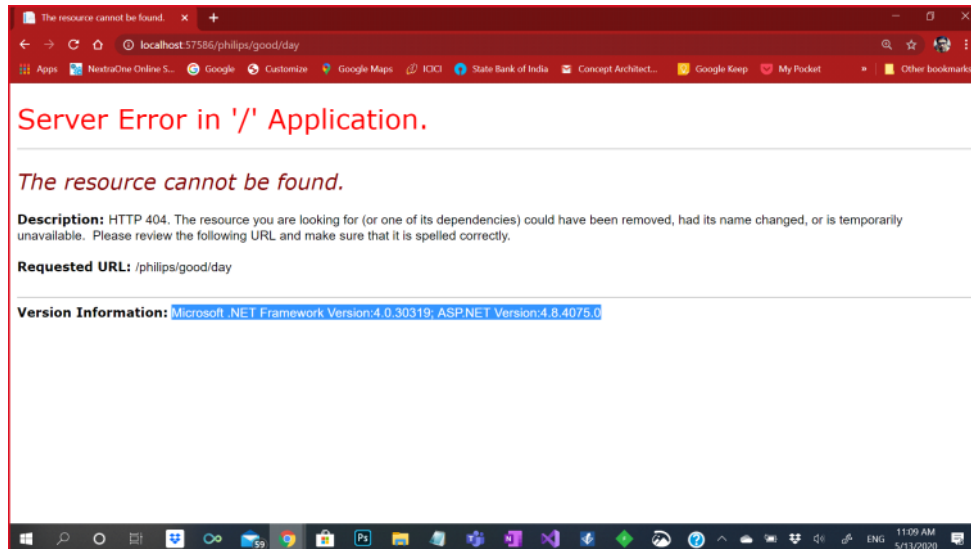
- Technology id such as
  - aspx/php/jsp
  - data format json/xml
  - device id eg m.facebook.com or irctc.in/mobile

<http://yt.be/5x9q8z>

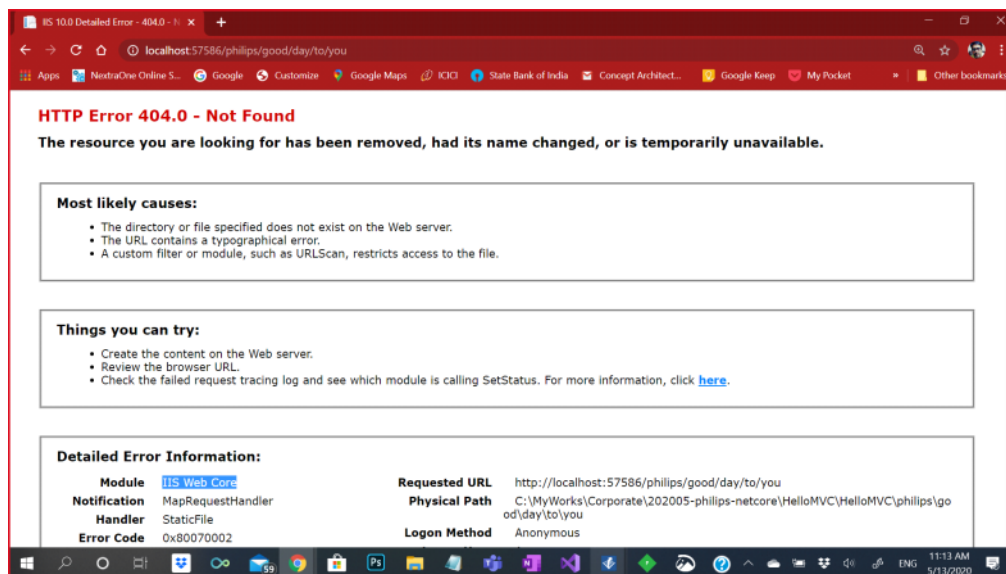
- SHORT BUT NOT SIMPLE!

# MVC Lifecycle Error Handler

Wednesday, May 13, 2020 11:09 AM



- A 3 part URL is recognized as an MVC URL
- Routing Module passes control to MvcRouteHandler
- MvcRouteHandler breaks the URL into
  - controller: philips
  - action: good
  - id: 'day'
- MvcRouteHandler passes control to MvcRequestHandler
- MvcRequestHandler successfully creates controller 'PhilipsController'
- Not the action 'good' is not available
- **MvcRequestHandler** returns 404



- A Route that is not configured by the RoutingModule falls back to IIS
- IIS tries to handle the url in traditional way by either
  - extension mapping
  - physical file search
- when both fails **IIS (not asp.net)** generates 404

# Assignment

Wednesday, May 13, 2020 11:54 AM

- Create a request for generating the Multiplication Table of a given number
- You choose your own URL structure
- The Url should take the number whose table is to be generated
- It should take an option second parameter for highest multiplier which should default to 10 if not passed
- Define success and error results properly

Use case 1 ---> Multiplication Table of 19 (Till its 10th Multiple)

Table of 19

Number	X	Result
19	1	19
19	2	38
19	3	
19	4	
19	5	
19	6	
19	7	
19	8	
19	9	
19	10	190

Use case 2 ----> Multiplication Table of 19 optional parameter sets highest multiple to 5

Table of 19

Number	X	Result
19	1	19
19	2	38
19	3	
19	4	
19	5	

# Conventional Programming

Thursday, May 14, 2020 9:20 AM

## Convention over configuration

- \* Prefer a conventional coding
  - \* reduces the need of configuration
- You are allowed to break/override convention if required
  - \* configuration can be one way to override convention



# Controller Execute

Thursday, May 14, 2020 9:41 AM

```
class Controller : Icontroller
{

    public void Execute( RequestContext context)
    {
        var httpContext= context.HttpContext;
        var routeData=context.RouteData;

        var actionName= routeData.Values["action"];

        var result= ActionInvoker.Invoke( actionName ); //this will search and execute an action method present in the controller

        ActionResult actionResult;

        if(result is ActionResult)
            actionResult=result;
        else
            result=new ContentResult(){ Content= result.ToString(); } //any normal return type is converted to ActionResult

        result.Exectue(); //generates the response!
    }

}
```

# Preventing Javascript Injection Attack!

Thursday, May 14, 2020 10:04 AM

## What if I type these content in a facebook wallpost

```
<h1>Hello There, I hav a news</h1>
```

```
<script>alert("Hi, I have a news");</script>
```

```
<script> for(i=0;i<100;i++) alert("Hello World"); </script>
```

If you are allowed to add any arbitrary html content to a web page user may

- Use Rich HTML/CSS format to express their idea, but it may break the site's theme
- Inject Scripts that may
  - deface the page at best
  - may inject malicious javascript to steal information and so on.
- Damage potential is endless

## Solution!

- Most modern ViewEngine doesn't allow you to inject any HTML code as data to a page
- Any data inject to a page will automatically be encoded by replacing "<" with "&lt;" and ">" with "&gt;"
- These codes will appear on the page but will not be treated as HTML
- Some application may simple refuse to inject the code and fail with a validation error.

# Books Web Site

Thursday, May 14, 2020 12:49 PM

## Overview

- A web app for book lovers where user can find
  1. A List of Books with informations such as
    - i. Title
    - ii. Author
    - iii. Price
    - iv. Description
    - v. Coverpage etc
    - vi. Reviews and Rating
    - vii. User can also leave their review about the book
  2. A List of Authors
    - i. Author Name
    - ii. Their Biography
    - iii. Photograps
    - iv. Books Written By Them
    - v. Reviews and Rating
  3. User
    - i. Login
    - ii. Logout
    - iii. Maintains A Book Shelf
    - iv. Maintain Contributions such as
      - 1) Reviews

## Assignment

1. Create The Model Classes
  - a. Book
  - b. Author
  - c. User
2. Plan the Routes (URL) to
  - a. what information you want to access
  - b. what operations you need to provide etc
3. Plan a outline of the pages that you need

# Book List

Friday, May 15, 2020

9:35 AM

Title	Author
My Experiments with Truth	Mahatma Gandhi
Mahatma Gandhi	Xyz
My Understanding Gita	Mahatma Gandhi

```
<table>
  <tr>
    <td>Title</td>
    <td>Author</td>
  </tr>

  <tr>
    <td>My Experiements with Truth</td>
    <td>Mahatma Gandhi</td>
  </tr>

  <tr>
    <td>Mahatma Gandhi</td>
    <td>Xyz</td>
  </tr>

  <tr>
    <td>My Understanding of Gita</td>
    <td>Mahatma Gandhi</td>
  </tr>
</table>
```

**What is difference Xml Vs HTML?**

**Xml is the information, HTML is presentation of information.**

**Xml retains the structure and meaning of data it holds**

- you can easily find out all the books authored by Mahatma Gandhi
  - Mahatama Gandhi doesn't appear in 'td' but in 'author'

**HTML looses the information in trying to a create a presentaion!**

**Xml vs Json**

- both are used in same scenario -- represent information
- JSON is more compact than XML
- JSON is easily understandable by Javascript
  - The most popular client for consuming services

**How to I select Book by Mahatma Gandhi and not on Mahatma Gandhi????**

**HTML is a picture of the information  
And sorrounded by HTML tags the information is often lost**

**Xml Representation**

```
<books>

  <book>
    <title>My Experiements with Truth</title>
    <author>Mahatma Gandhi</author>
  </book>

  <book>
    <title>Mahatma Gandhi</title>
    <author>Xyz</author>
  </book>

  <book>
    <title>My Understanding of Gita</title>
    <author>Mahatma Gandhi</author>
  </book>
</books>
```

**JSON Representation**

```
[
  "book":{
    "title": "My Experiements with Truth",
    "author": "Mahatma Gandhi"
  },
  "book":{
    "title": "Mahatma Gandhi",
    "author": "Xyz",
  }
  "book":{
    "title": "My Understanding of Gita",
    "author": "Mahatma Gandhi",
  }
]
```

# Cloud Vs Normal Internet Deployment

Friday, May 15, 2020 11:07 AM

- Internet deployment may take form of IAAS, PAAS, SAAS etc
- Each will have few layers

## Cloud


- defining feature is **Virtualization a.k.a Virtual Machine**
- Multiple Virtual Machines may be present on the same physical infrastructure
- Each Virtual machine will get promised computation power ( CPU/Storage/RAM/Bandwidth)
- **A virtual machine may be replicated on multiple infrastructure located in different geographical regions**
- Each change may be synchronized across various instances of the same server
- This allows
  - Scaling of requirement
    - We can launch a copy of virtual machine within a short span of time. A real hardware installation requires longer
    - We can launch such VMs nearest to my consumer base reducing the turnaround time.
    - Proper Load Balancing among various instances
  - Robust
    - Even if one VM or geographical location is compromised your system is still available via other instance
    - You are not completely out of business because of geographical conditions.

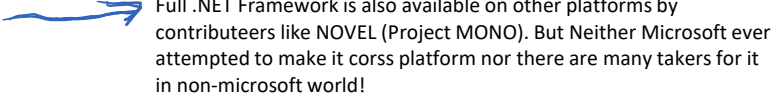
## Virtual Machine Requirement

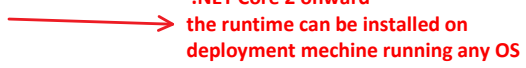
- Light weight software system
  - Easy to replicate
- Smaller virtual machines
- Design which can handle failures and crashes.

# .Net Core vs .NET Framework

Friday, May 15, 2020 11:23 AM

- .NET is
  - **Lightweight** (more like a subset) of full .NET Framework.
  - Doesn't support all features of .NET Framework like --
    - WinFrom
    - WebForms
    - WCF
    - WPF

**Introduced in .NET Core 3**
  - Make it universally available. unlike .NET Framework .NET core is easily deployed on
    - Linux
    - Mac
    - **High avialbility**

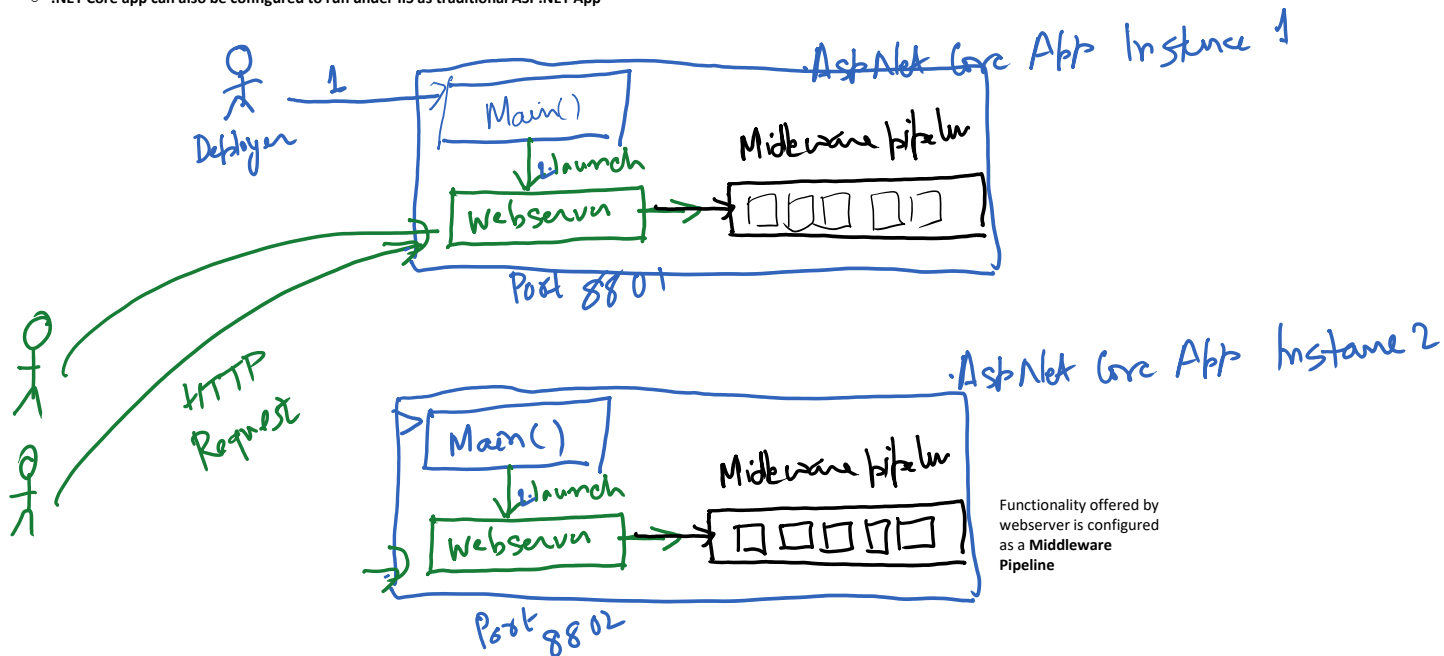
Full .NET Framework is also available on other platforms by contributeers like NOVEL (Project MONO). But Neither Microsoft ever attempted to make it corss platform nor there are many takers for it in non-microsoft world!
  - Restructure design to support scalable features for
    - MVC Application
    - WebAPI
    - Entity Framework
  - Highly configurable
  - Zero Installation
    - .NET core runtime may be embeded within your application
    - Application can run anywhere.
    - Less deployment problems!

**.NET Core 2 onward  
the runtime can be installed on  
deployment mechine running any OS**

# ASP.NET Core

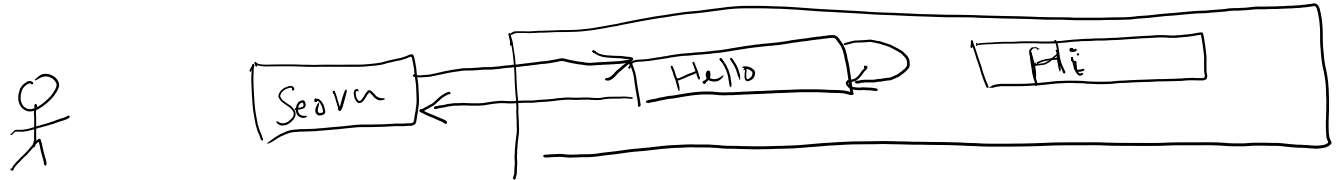
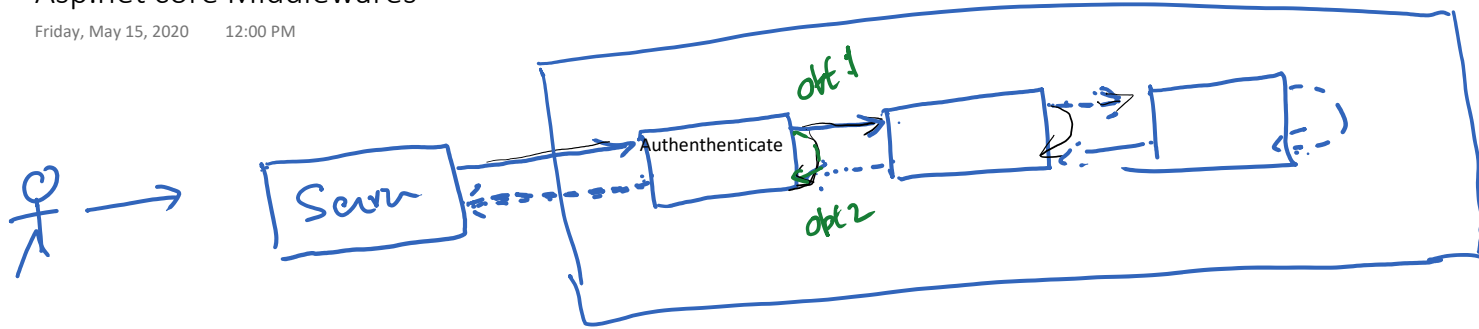
Friday, May 15, 2020 11:31 AM

- ASP.NET core is different from .NET Core in the sense
  - It is not container architecture
  - Your application doesn't run inside IIS (or a server)
  - Your application runs (lightweight) server inside it
  - Your application is more like a console application with its own Main
    - When you run Main(), it launches an IIS Server
      - Very easy to deploy. Just run Main
        - ♦ Remember a traditional ASP.NET Web Applications don't have their functions.
      - Since Server is embedded the underlying machine just need .NET Core framework on NOT IIS
      - Your machine listens to its own request and responds back on its
    - You may run multiple instances of your application on same machine (may be different port)
  - Your application contains
    - Server
    - Web Application
  - The server is not heavy weight structure like IIS
  - .NET Core app can also be configured to run under IIS as traditional ASP.NET App



# Asp.net core Middlewares

Friday, May 15, 2020 12:00 PM





<http://mmt.com/search/flights/blr-delhi/2020-06-06>

```
class SearchController{

    public async void Flights(String route, DateTime date)
    {
        //step 1. search flights with different airlines
        //step 1.1 --> AirIndia
        var result1=await airIndia.Search(route,date);
        //step 1.2 --> GoIndigo
        var result2=await goIndigo.Search(route,date);
        //step 1.3 --> Spicejet
        var result3=await spicejet.Search(route,date);

        //step 2. aggregate them together
        var mergedResult=ResultMerger.Merge(result1,result2,result3);

        //step 3. return the response

    }
}
```

what happens to current thread while result is being processed by other server?

1. It should remain idle.
2. **It may be used to process some other request by some other client --> async programming.**

# .NET Core USE

Monday, May 18, 2020 10:23 AM

## Request Delegate

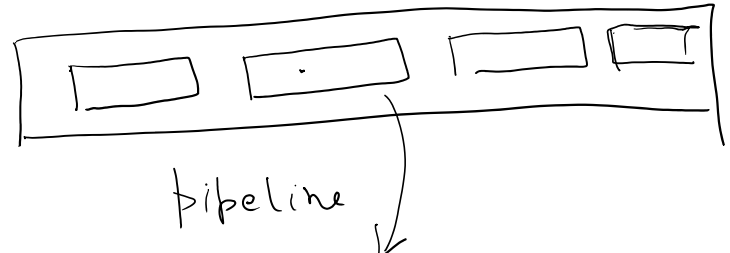
- A Request delegate is a handler or the actual object that can process request.
- It gets an HttpContext and it can process the request and return the response.

## Middleware

- Is a component in the request pipeline.
- It configures a piece of **RequestDelegate**
- Middleware can be considered as a wrapper over RequestDelegate
- The term Middleware/Request delegate is often be used interchangeably

```
public RequestDelegate Middleware(RequestDelegate next)
{
    if(some_condtion)
        return new_delegate; //action performed by this middleware
    else
        return next ; //pass the control to the next middleware
}
```

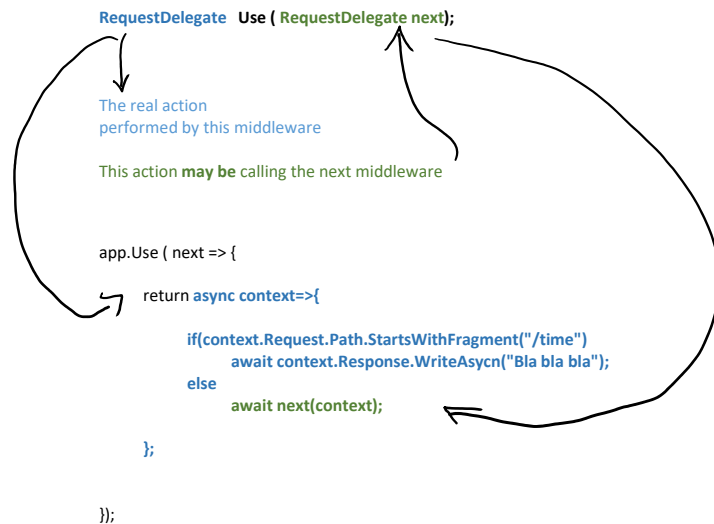
- Here next refers to next middleware in the pipeline
- The middleware conditionally **returns** either can --
  - pass request to the next element in the pipeline
  - or return a new delegate representing the action



Key role of a middleware is one of these two

1. Process the request
2. Pass it to middle ware

- Some middlewares may do both.
- But everyone must do at least one of the two



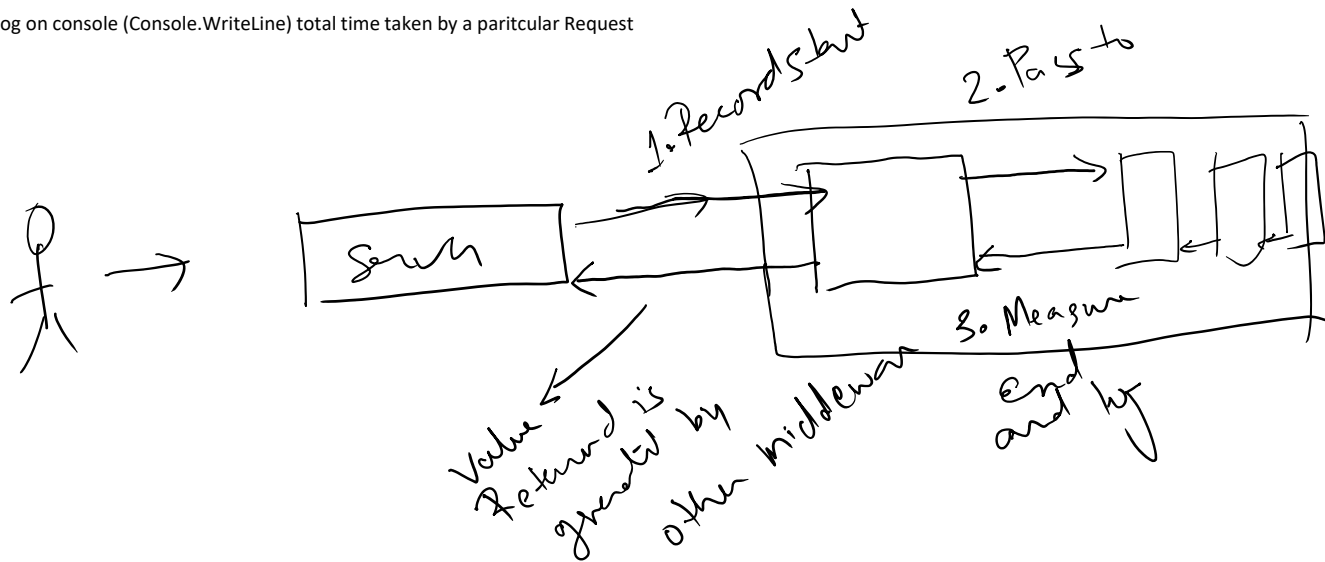
Incase you can't handle the request please pass to next middleware in pipeline and don't break the chain!

## Assignment -- Performance Middleware (Filter)

Monday, May 18, 2020 10:40 AM

Create a Performance Logging middleware

- should log on console (Console.WriteLine) total time taken by a particular Request



# Assignment2 --> Generic Helper to Configure A Url based Middleware

Monday, May 18, 2020 11:13 AM

```
app.Use(next =>
{
    //This is the middleware
    return async context =>
    {
        if (context.Request.Path.StartsWithSegments("/time"))
        {
            await Task.Delay(2500);
            var date = DateTime.Now;
            await context.Response.WriteAsync(date.ToLongTimeString());
        }
        else
        {
            await next(context);
        }
    };
});
```

In this code there are few common steps

1. check the URL
  - a. If Url matches then perform a **execute RequestDelegate**
  - b. **else** execute **next() delegate**

Write a helper

1. which can take URL and RequestDelegate as parameter
2. Invokes your RequestDelegate if parameter matches
3. Invokes Next delegate (not passed as parameter) if url is a mismatch

# Assignment UseBefore and UseAfter

Monday, May 18, 2020

11:45 AM

Follow the Idea of Middleware

public static class MiddleWares

```
{
    public static void UseMappedUrl(this IApplicationBuilder app,string url, RequestDelegate handler)
    {
        //This is the middleware
        app.Use(next =>
        {
            //This is the middleware
            return async context =>
            {
                if (context.Request.Path.StartsWithSegments(url))
                {
                    await handler(context);
                }
                else
                {
                    await next(context);
                }
            };
        });
    }
}
```

## Create Helpers for Filters that can run before after next in pipeline.

- The pipeline shall execute
- Before executes a code before passing to next pipeline
- After executes a code after executing next pipeline

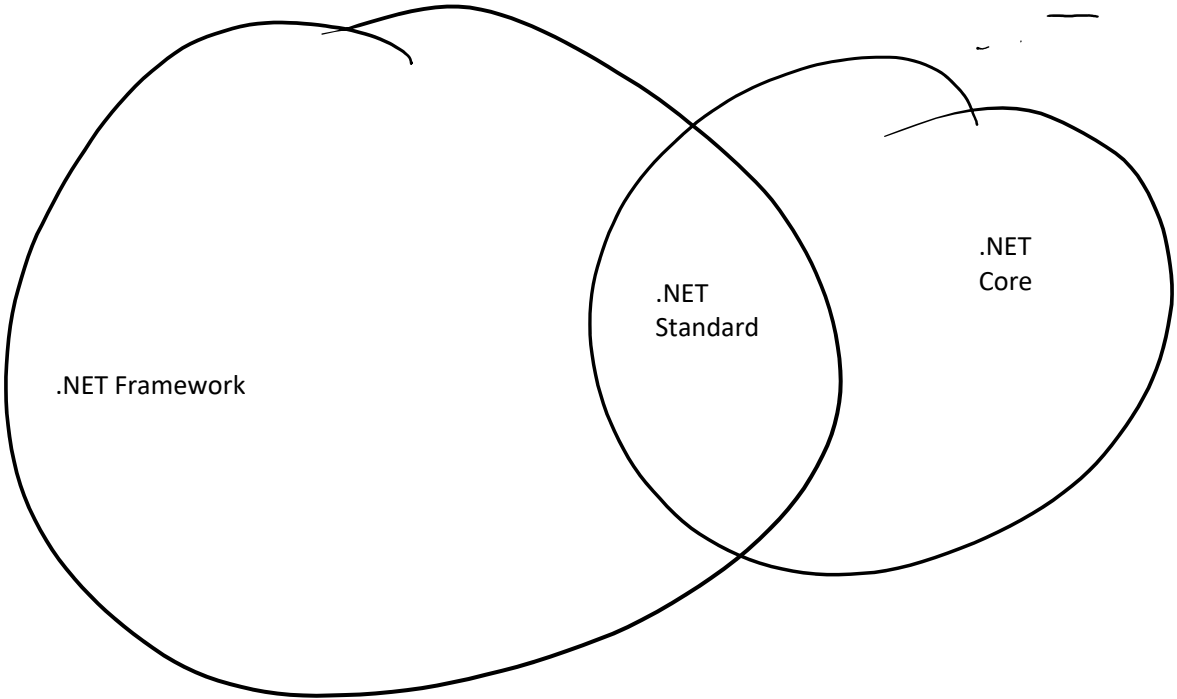
app.Use

Before(async context =>

```
{
    Console.WriteLine("Request: " + context.Request.Path);
    //next is called automatically
});
```

app.UseAfter(async context =>

```
{
    //next is called here
    Console.WriteLine("Requested Processed:" + context.Request.Path);
});
```



# MVC Model Binding

Tuesday, May 19, 2020 12:19 PM

```
ActionResult Create(Author author){  
  
}
```

Before invoking Action Method , base class Controller's execute method does the following

1. Identifies the function that represents Action
  - a. same as function name
  - b. ActionName attribute defined
2. Check the paramter required by the function (**Author**)
3. Creates the Author Object using default constructor
4. Check's each property of the Model Object at tries to find a matching value in
  - a. Route
  - b. QueryString
  - c. Form Fields
5. The property and request parameters are matched based on name in case insenitive form

# Resolving an Action Name

Tuesday, May 19, 2020 12:25 PM

Which of the below action shall be called for following URL

<http://books.org/author/create>

```
class AuthorController
{
    [HttpGet]
    public ActionResult Create()
    {
        var author = new Author();
        return View(author);
    }

    [HttpPost]
    public ActionResult Create(Author author) //model binding
    {
        return View("Details", author);
    }
}
```

MVC can call this method without passing any parameter

or

MVC can call this function by passing empty **new Author()** object as parameter

Since there is no clear answer It would raise an exception.

We can avoid ambiguity by specifying HTTP method associated with each Action

- specifying either HttpGet or HttpPost on one of the action will also work fine
- Other will server the non-specified request



# Dependency Management

Tuesday, May 19, 2020 12:51 PM

## Who should know which Repository Implementation to Use?

- **Model should not know**
  - The repository implementation may change tomorrow
  - As per design guidelines, Model (Service) should know only Repository interface Not actual implementation
  - This is dependency inversion
- **Controller should not know**
  - Controller never interacts with repository, it interacts with model
  - Why should controller know something it doesn't interact with?
  - Controller should know even the model with its interface and not its implementation
    - This is again dependency inversion
- **Controller should not know**
  - Remember View should be stupid
  - View gets data from Controller, who gets data from model who gets data from repository it doesn't know about

Mvc Requires Dependency Injection, but it is not the job of Model, View or Controller

We need some other entity or framework to do the dependency Injection

- .NET framework and .NET Core differ Here

### .Net Framework MVC

- doesn't have builtin support for dependency injection
- Some interface exist but implementation must be provided
- we generally use separate framework for dependency injection such as
  - Unity
  - nInject
  - MEF

### .Net Core MVC

- Dependency Injection is builtin in asp.net core
- It is one of the fundamental features of .net core
- no thirdparty dependency injector is required
- You may configure one if you need

# Middleware Assignments

Thursday, May 21, 2020 9:11 AM

## A Middleware to Block Unreferend or HotLinked Image Request

- Should test if request is for an image
- It should be rejected if
  - There is no referer mentioned
  - If mentioned referer is not in approved list
- **app.UseHotLinkImageBlocker()**

## A Middleware to record 404 Urls

- **Middleware should push all URLs resulted in 404 in the database**
- This could be a great feature for a product like website
- In booksworld 404 may refer to a book or author not present in database
  - This record can help us update our database based on client's need
- A website like Amazon can use this as wishlist for people searching for unavailable products

//should block any request for image files (.jpg, .png)

//that is unreffered or referred from a differnt domain

//app.UseImageHotlinkBlocker();

//inspect and log which Urls user is requesting and which is

//not found.

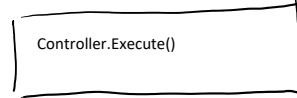
//You may add this URL to a database

//app.UseNotFoundUrlLogger();

# Asp.net mvc validation

Thursday, May 21, 2020 11:49 AM

- MVC executes a Validation on the submitted model using a ModelValidator
- The entire flow is handled by base class Controller Execute method



```
class Controller
{
    public void Execute(RequestContext context)
    {
        //1. find out which is the right action method
        var method= FindActionMethod();

        //2. create necessary model for the action
        // 2.1 create object using default constructor
        // 2.2. check each model property and try to findout a matching value in request
        // 2.3. assign the matching values and ignore the rest
        var model= modelBinder.CreateModel();

        //3. Execute ModelValidator to verify if Model is valid or Not. Generate a Result as ModelState
        var modelState=ModelValidator.ValidateModel(model);

        //4. Execute The Action Method with Model passed as parameter
        var result= ActionInvoker.Invoke(actionName, model)

        //5. Now process the result...
    }
}

class AuthorController
{
    [HttpPost]
    public ActionResult Create(Author author)
    {
    }
}
```

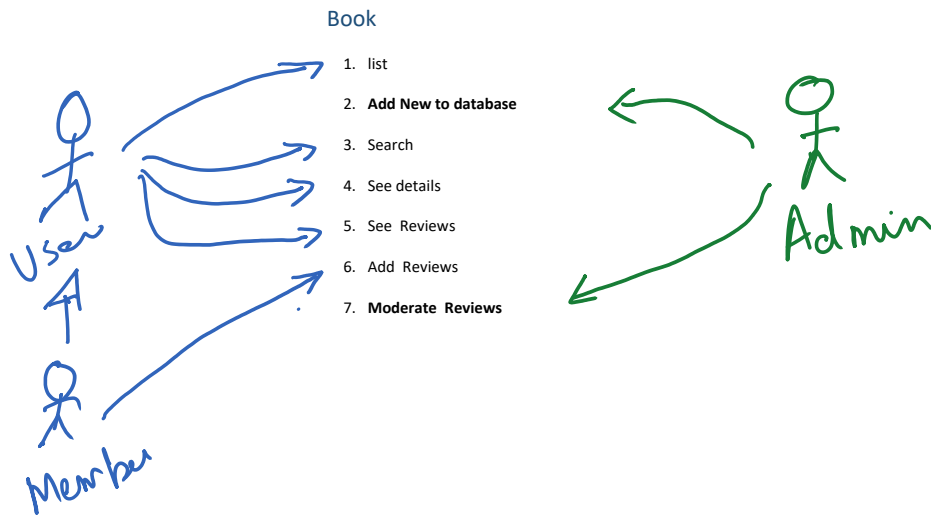
## Key Elements

Book Database Management

Author Database Management

Reviews

User and Authnetication



## Urls

### List all books

<https://books.org/book/list>

or

<https://books.org/books>

### Search Books

<https://books.org/books?q=TheAccursedGod>

### Create New Book

<https://books.org/books/create>

### Book By Id

<https://books.org/books/info/the-accursed-god>

### Edit Book

<https://books.org/books/edit/the-accursed-god>

### Edit Book

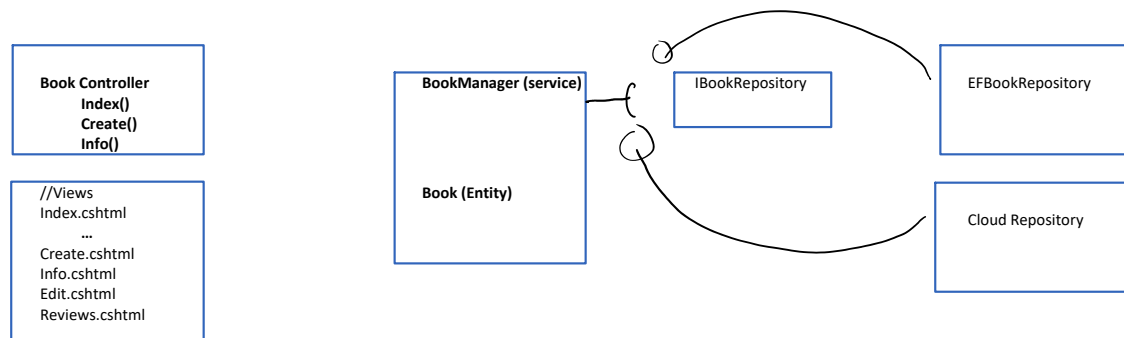
<https://books.org/books/delete/the-accursed-god>

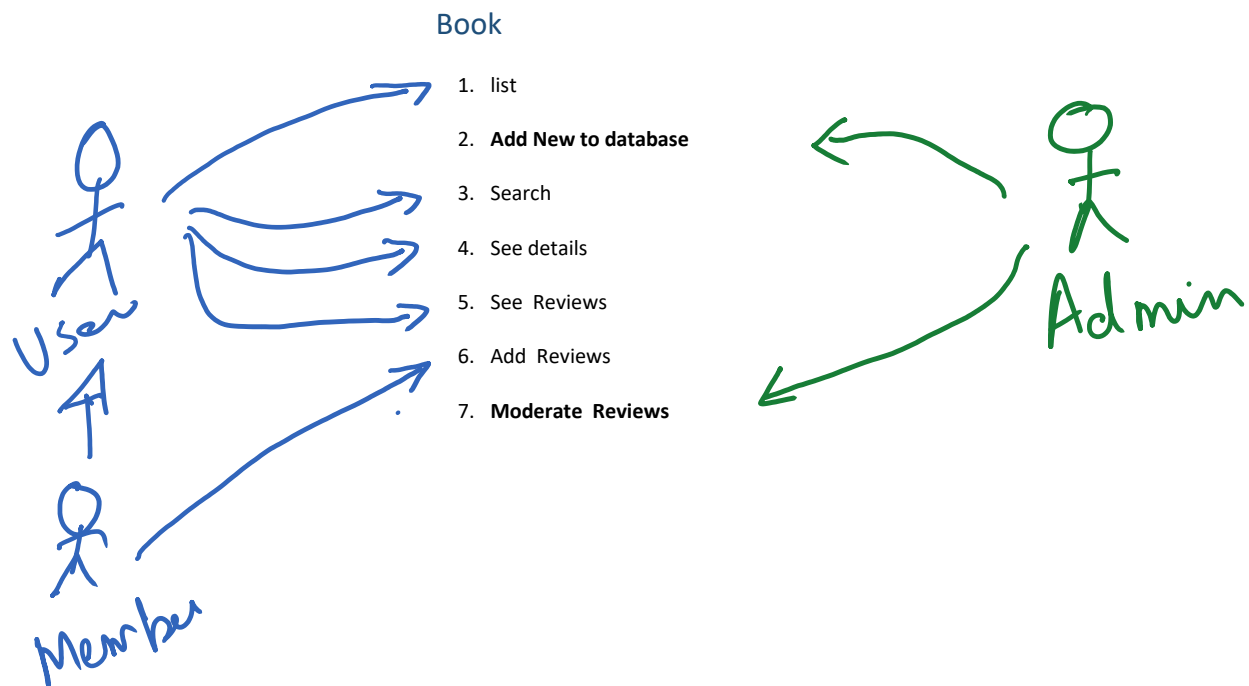
### List of Book Reviews

<https://books.org/books/reviews/the-accursed-god>

alt

<https://books.org/books/info/the-accursed-god/reviews> <--- Note requires a New Route

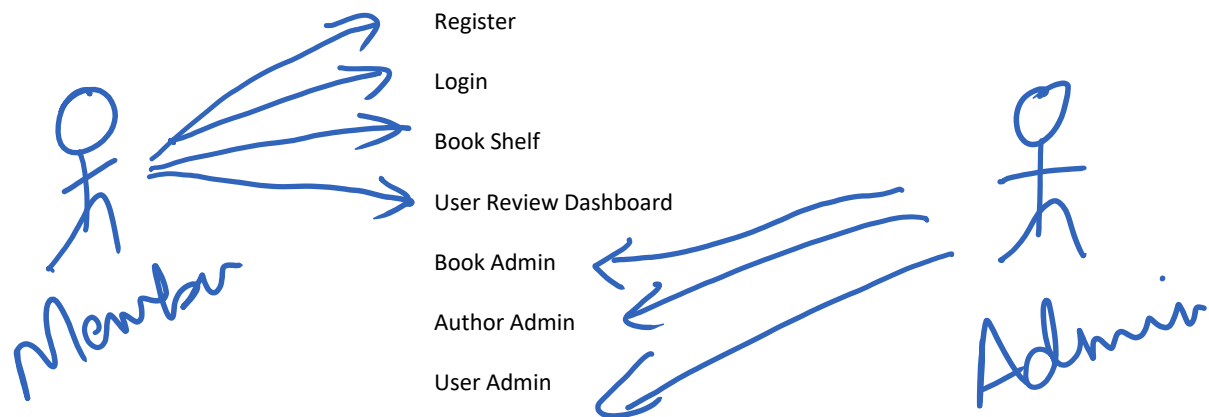




# User

Friday, May 15, 2020

8:45 AM



# Controller (Bad Design)

Monday, May 18, 2020 12:51 PM

```
class BookController
{
    public ActionResult BookList(){
    }
    public ActionResult AuthorList(){
    }
    public ActionResult UserRegister(){
    }
    public ActionResult UserLogin(){
    }
}
```

<https://books.org/book/booklist> <---- book is repeated twice

<http://books.org/book/authorlist> <--- book is not required in url

<http://books.org/book/userlogin> <----- book is not required in url

*Don't use this*



# Controller

Monday, May 18, 2020 12:51 PM

```
class class BookController
{
```

```
    public ActionResult List(){
```

```
    }
```

```
    public ActionResult Info(String id){
```

```
    }
```

```
}
```

```
public class AuthorController{
```

```
    public ActionResult List(){
```

```
    }
```

```
    public ActionResult Info(string id){
```

```
    }
```

```
}
```

```
public class UserController{
```

```
    public ActionResult Register(){
```

```
    }
```

```
    public ActionResult Login(){
```

```
    }
```

```
}
```

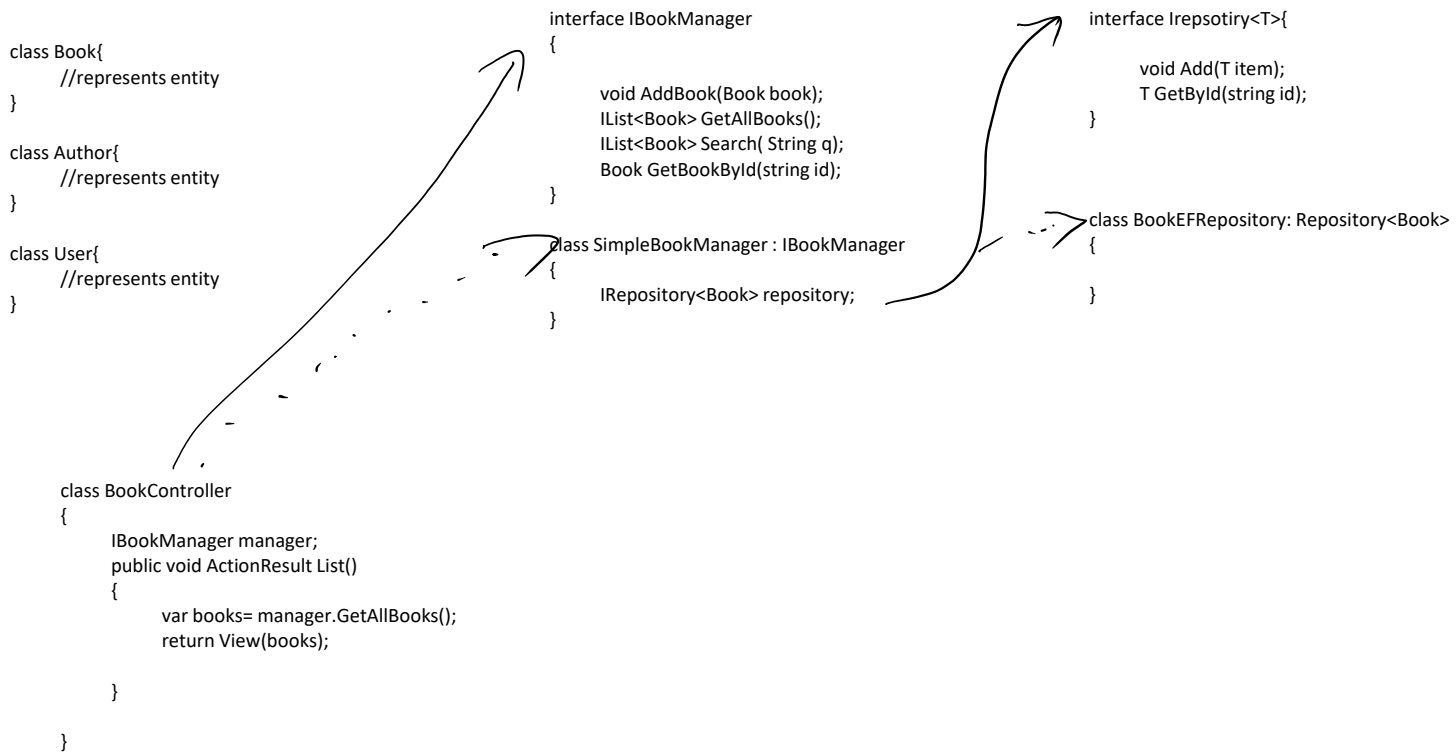
<https://books.org/book/list> <---- book is repeated twice  
<https://books.org/book/info/the-accursed-god>

<https://books.org/author/list>  
<https://books.org/author/info/jeffrey-archer>

<https://books.org/user/register>  
<https://books.org/user/login>

# Architecture

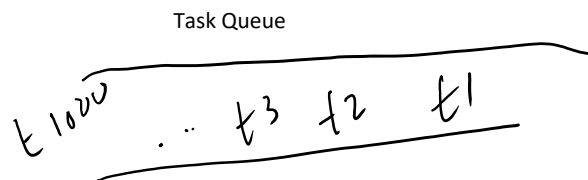
Monday, May 18, 2020 12:57 PM



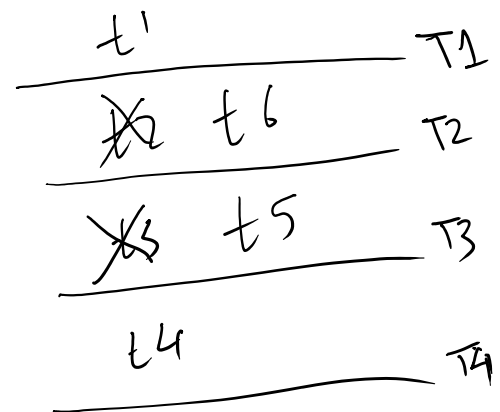
# Task

Tuesday, May 19, 2020 9:16 AM

- Lightweight abstraction of Asynchronous programming
- Represents a piece of job that should work asynchronously
- Tasks executes on a Threadpool



## Thread Pool



# Task and async programming

Tuesday, May 19, 2020 9:22 AM

- A Task is a piece of long running job whose result may come later
- We are no sure when the result will come!

```
int CountPrimes(int min,int max){  
  
    int count=0;  
    for(int i=min;i<max;i++){  
        if(IsPrime(i))  
            count++;  
    }  
  
    return count;  
}
```

counting primes between 2-5lack may take upto 40 seconds.

//Scenario #1

```
void Main(){  
  
    int r1= CountPrimes(2,500000);  
  
    int r2=CountPrimes(2,500);  
    print(r1);  
    print(r2);  
  
}
```

second calculation is smaller than the first one and are unrelated. There is no need for second to wait for first to finish. But we wait on first line to be over before we move to second line.

**This is synchronous programming**

//Task based design  
void Main(){

Task<int> t1=Task.Factory.StartNew( ()=> CountPrimes(2,500000));

Task<int> t2= Task.Factory.StartNew( ()=> CountPrimes(2,100));

Task sleep=LongSleep();

//both tasks have started together and second may end before first  
t2.Wait();  
print(t2.Result);

//t1.Wait(); //----> blocks current execution till Task is complete. Main **waits** for task to complete  
print(t1.Result);

}

public Task LongSleep(){

return Task.Factory.StartNew(()=> Thread.Sleep(10000));

}

# Task vs async await

Tuesday, May 19, 2020 9:32 AM

```
int CountPrimes(int min, int max)
{
    int count=0;
    for(int i=min;i<max;i++){
        if(IsPrime(i))
            count++;
    }

    return count;
}
```

```
Task<int> CountPrimesAsync(int min, int max)
{
    return Task.Factory.StartNew(()=> CountPrime(min,max));
}
```

```
public Task LongSleep(){
    return Task.Factory.StartNew(()=> Thread.Sleep(10000));
}
```

```
//This is a sync call around a Task
int FindPrimes(int min, int max)
{
    var pt=CountPrimesAsync();
    pt.Wait(); //this function waits. and this function is sync
    return pt.Result;
}
```

```
void Main(){
    var result=FindPrimes(2,50000)
    //you reach here only when the job is over.
    print(result);
}
```

```
Task<int> FindPrimesAsync(int min, int max)
{
    var pt=CountPrimesAsync();

    return pt;
}
```

```
void Main(){
    Task<int> result=FindPrimesAsync(2,50000)
    //you reach here before the job is over.

    //you should wait for the job to be over
    result.Wait();
    print(result.Result);
}
```

```
async int FindPrimesAsync(int min, int max)
{
    var pt=CountPrimesAsync();

    var result = await pt; //someone else will wait for it
    return result;
}
```

async  
automatically  
wraps  
result  
in a Task

the function  
is actually  
returning

Task<int>

because you  
await Task<int>

```
void Main(){
    Task<int> result=FindPrimesAsync(2,50000)
    //you reach here before the job is over.

    //you should wait for the job to be over
    result.Wait();
    print(result.Result);
}
```

//result will be an int. so I am actually returning Task<int>

- Any function that has an await anywhere in its body will always be a task as there is something awaited
- await** need not be part of return statement
- if a function that has await returns a value of type **double** it means it actually returns a **Task<double>**
- if a function that has await returns **no value** it actually returns **Task**
- await can be a part of the function that has async in the signature!**
  - Main()** can't have **async await**
  - Main()** must wait **sync**

```
async int FindPrimesAsync(int min, int max)
{
    var pt=CountPrimesAsync(min,max);

    var result = await pt; //someone else will wait for it

    return result; //returns Task<int>
}
```

```
async void PrintPrimeAsync(int min, int max)
{
    var result= await CountPrimesAsync(min,max);

    Console.WriteLine("Total primes between {0}-{1} is {2}",min,max,result);
}

void Main()
{
    Task t=PrintPrimeAsync(2,50000); //this work will finish later
    //I can't await in Main

    t.Wait(); //must actually wait

    Console.WriteLine("End of Program");
}
```

```
Task PrintPrimeAsync(int min, int max)
{
    var task1=CountPrimesAsync(min,max);

    var task2= task1.ContinueWith( r=>{
        Console.WriteLine("Total primes between {0}-{1} is {2}",min,max,result);
    });

    returns task2
}
```

# Model should be POCO

Wednesday, May 20, 2020 9:11 AM

## Plain Old CLR (C#) Object

- You object should be Plain
- It shouldn't have dependency on third party or non business (architectural) frameworks
  - Entity Framework
  - WPF
  - WCF
- Your Model represents a business and the business may not be presented using a particular framework
  - We may not always use MVC or WPF or such features.
- Avoid dependencies such as
  - Inheriting from framework specific base classes or implementing such interfaces
    - **DependencyObject in WPF**
    - **EntityObject**
      - Earlier version of EntityFramework expected you to inherit from this base class
  - Shouldn't use components from such framework directly
    - Shouldn't contain SqlConnection class inside Business Model
    - Shouldn't use EntityCollection as a part of the business Object
    - **You should use dependency inversion!**

## What doesn't break POCO

- Dependency on fundamental domain or core .net library not associated with any architecture doesn't break POCO
- Dependency on namespaces **System, System.Collection, System.ComponentModel** is considered safe.
- Use of **Attributes even from third party library/framework is not bad.**
- **Why Attributes are not bad**
  1. **Attributes are non-obstrusive.**
    - i. **They do nothing on their own. Some API/Framework must process them**
    - ii. **If not used they can be ignored. They have no performance or otherwise issue**
  2. **They don't force you to carry extra steps like interface or base class**
  3. **If the dll where attribute is defined is not included in the runtime environment, there is no runtime error for missing references. Attributes are simply ignored.**

# Cross Reference/Cyclic Reference

Wednesday, May 20, 2020 9:35 AM

[Serializable]

```
public class Author
```

```
{  
    public string Id { get; set; }
```

```
    public string Name { get; set; }
```

```
    public IList<Book> Books { get; set; } = new List<Book>(); //sql world has no place for this list. It must be a reverse query
```

```
}
```

[Serializable]

```
public class Book
```

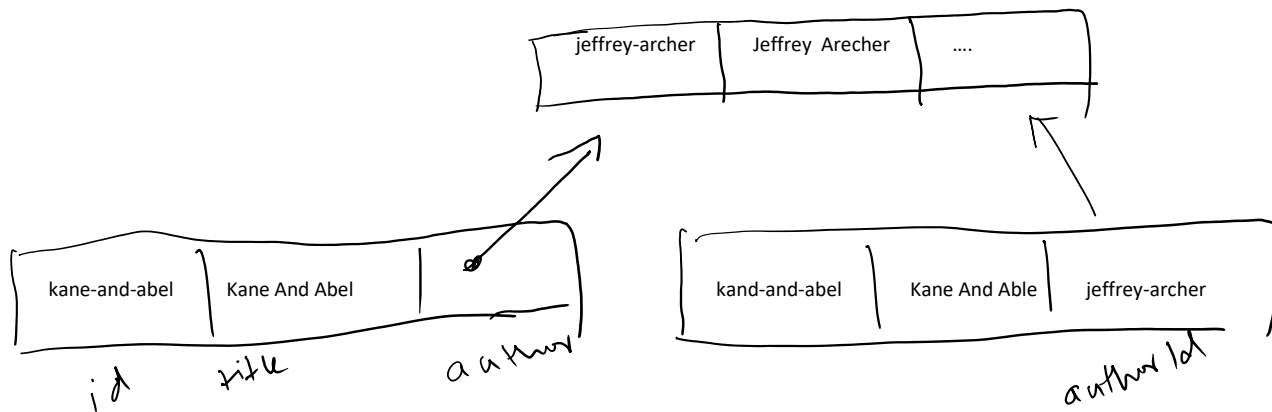
```
{  
    public string Id { get; set; }
```

```
    public string Title { get; set; }
```

```
    public Author Author { get; set; } //or the next line
```

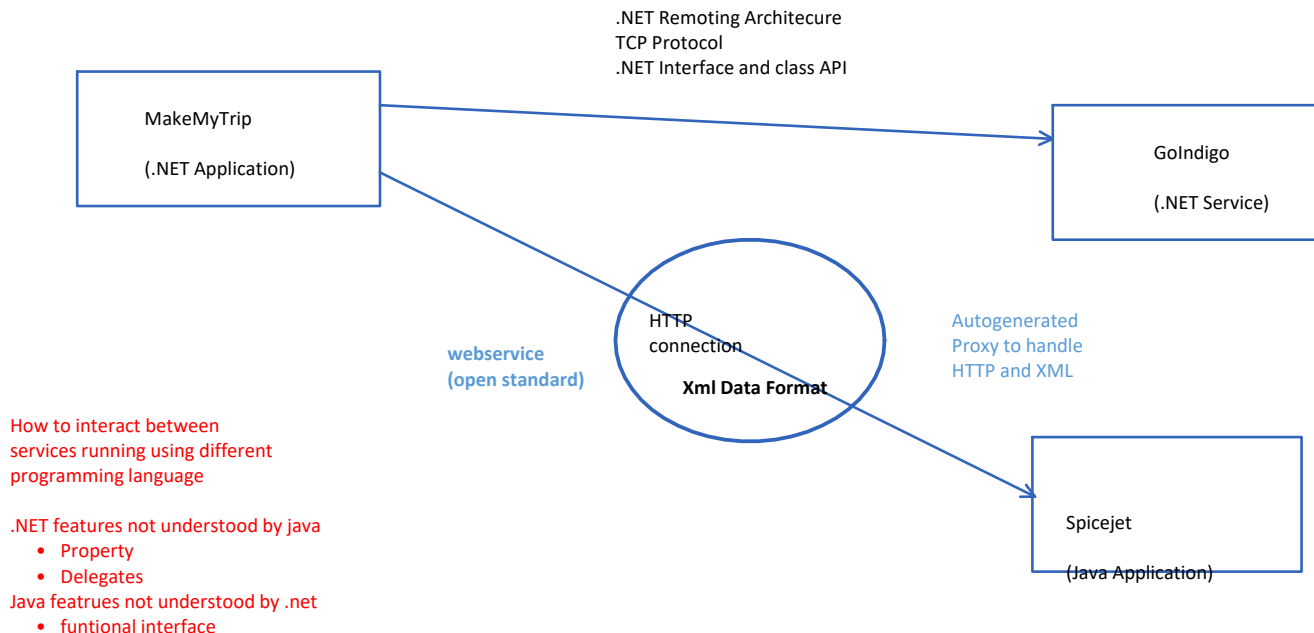
```
    public string AuthorId { get; set; } //RDBMS or SQL Related concept. Not OO --> force a join query to fetch records.
```

```
}
```



# Web Services

Friday, May 22, 2020 11:17 AM



## We Need common elements for communication

### Web Service (a.k.a SOAP Service)

- Most Acceptable Technology Stack
    1. **HTTP Protocol**
      - i. most whitelisted protocol
      - ii. most popular protocol
      - iii. every language has a lot of support feature
  - Data Transfer Format
    1. **XML**
      - i. Open Standard <-- we need to define the exact structure of data to be passed
1. **Xml Schema 1 ----> WSDL (Web Service Description Language)**
    - An Xml document to explain
      - what are the services (operations) available
      - what are the parameter each method takes
      - what are the parameter each method is expected to return
      - what security/authentication etc is needed
    - This document helps language create their own proxies to hide the actual communication
  2. **Xml Schema 2 ----> SOAP (Simple Object Access Protocol)**
    - A predefined XML format to transfer the runtime info between client and server
    - The details included here is
      - Name of operation (e.g. GetFlightList)
      - Required arguments (e.g. itinerary)
        - date
        - from
        - to
    - Result of the operation (e.g. List of Flights)



# Web Service Limitation

Friday, May 22, 2020 11:32 AM

1. Web Service has two important components

## Http Protocol

- Good
  - Popular
  - Lots of Support
  - Platform/Language/Vendor Independent
- Bad
  - Request-Response Mode
  - No Support for
    - Full duplex communication
    - Push communication
- Web Service Problem
  - Web service while making a request sends SOAP (XML)
  - Xml can't be send as query string (they are large)
  - It must be passed in the body of the request
  - **Webservice uses only POST method for all communication**
    - **No support for GET**
    - **POST doesn't support**
      - **Bookmarking**
      - **Caching**
    - **POST request need special infrastructure**
      - **At least an html form submission**
      - **You can't send POST request by typing an URL**
- URL Problem
  - **All operations which are part of same web service they are available on same URL**
  - **URL in web service is**
    - **NOT Unique to an operation or a resource**
    - **The actual operation is specified in SOAP data**
    - **Or using special HTTP Header**

**Remember!**  
**Web applications support**  
**GET and POST**

**Web Service supports just**  
**POST even for retrieve request!**

Web service uses limited aspect  
of HTTP and Not full and proper HTTP  
(Even Web Application is guilty of it)

## SOAP data

- Good
  - SOAP is XML
  - XML is popular
  - XML is open standard
  - Lots of support
- Bad
  - Xml is bulky
  - SOAP is bulkier
    - Lots of additional tags to represent SOAP headers
  - SOAP requires parsing and parsers are slow
  - **No body needs SOAP**

## Why No body needs SOAP?

- There are formats which are popular and many applications exist for them
- Examples
  - VCARD --> outlook, android, iphone
  - CSV --> excel, gmail , many other app
  - RSS --> new reader apps
  - PDF --> popular standard
- **There is no app that actually needs SOAP**
  - **SOAP is specifically for Webservice**
  - **Everyone who received SOAP must convert it something before using**
  - **Conversion is expensive**

## PROXY

- Good
  - Proxy makes the method call easy
    - You invoke a c# method which will do the heavyweight lifting in the background
    - You need not worry about HTTP or SOAP
- Bad
  - Proxy creation in scripting languages may be a trouble
  - It hides the HTTP element which may be important

# SOAP Free Service

Friday, May 22, 2020 11:45 AM

## Objective

To create a **Lightweight, True HTTP, SOAP Free Service**

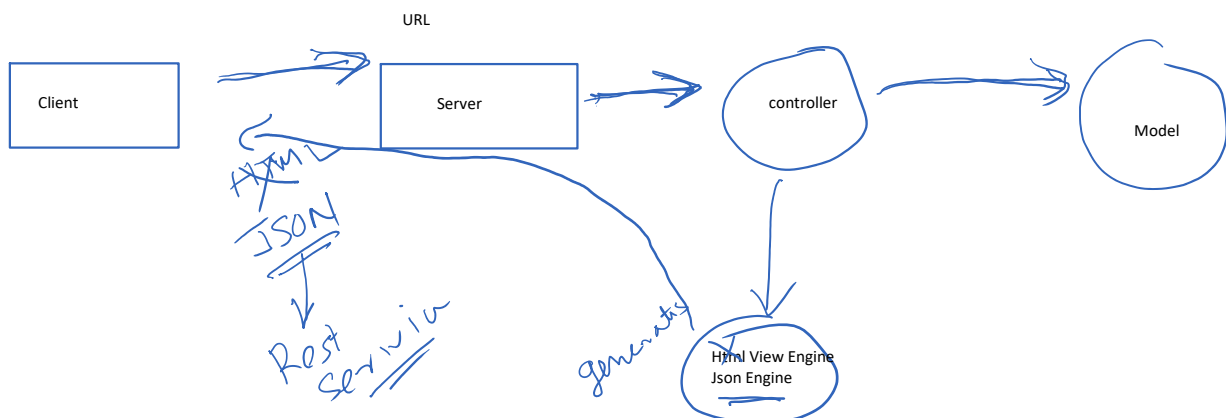
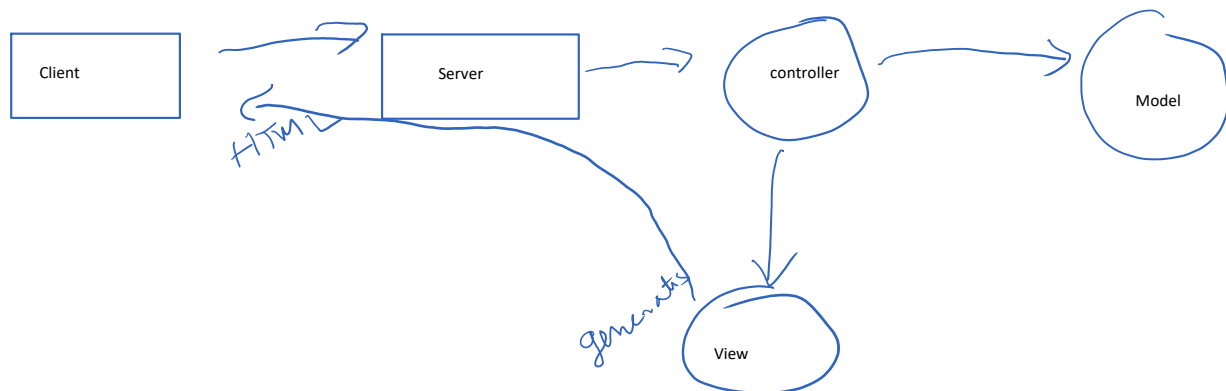
## Lightweight

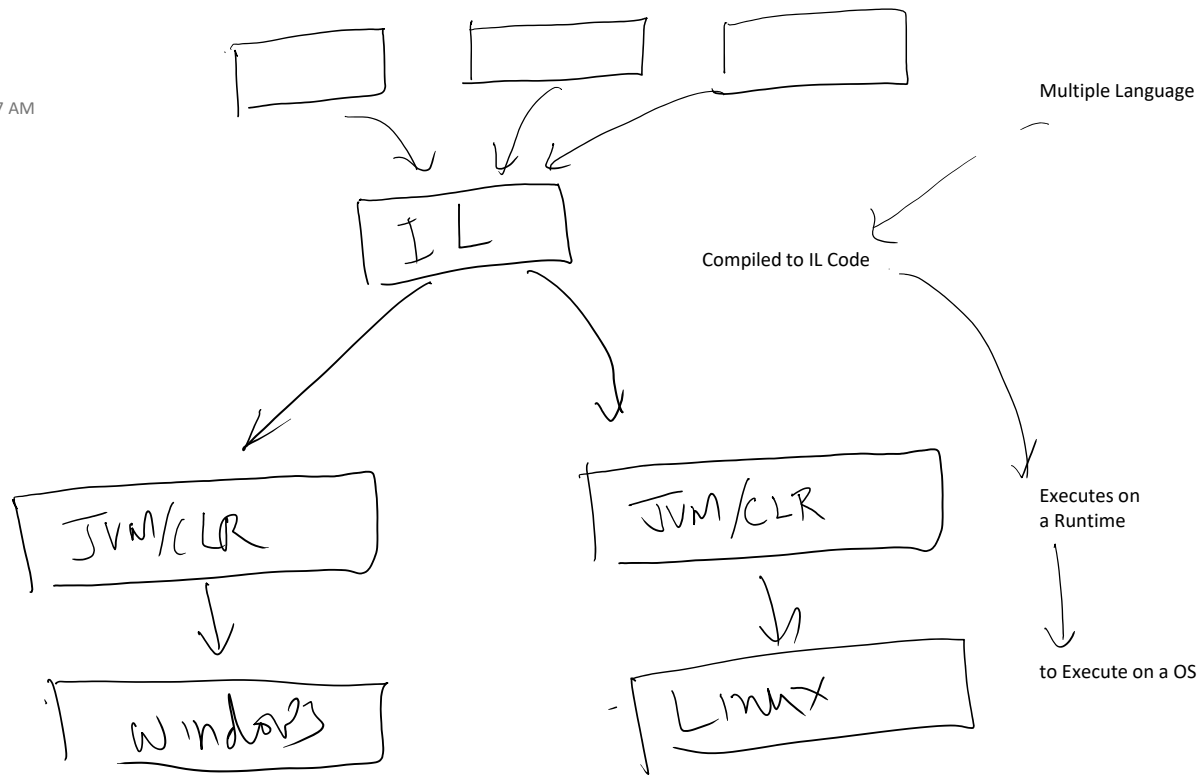
- No need of heavy weight architecture like need to generate proxy
- avoid heavy data transfer like SOAP

## True HTTP

- Follow HTTP recommendation property
- Use of proper methods
  - GET
  - POST
  - PUT
  - DELETE
  - ..
- Use of proper headers
  - Accept
  - ContentType
  - Status Codes
- **Use of Proper URL**

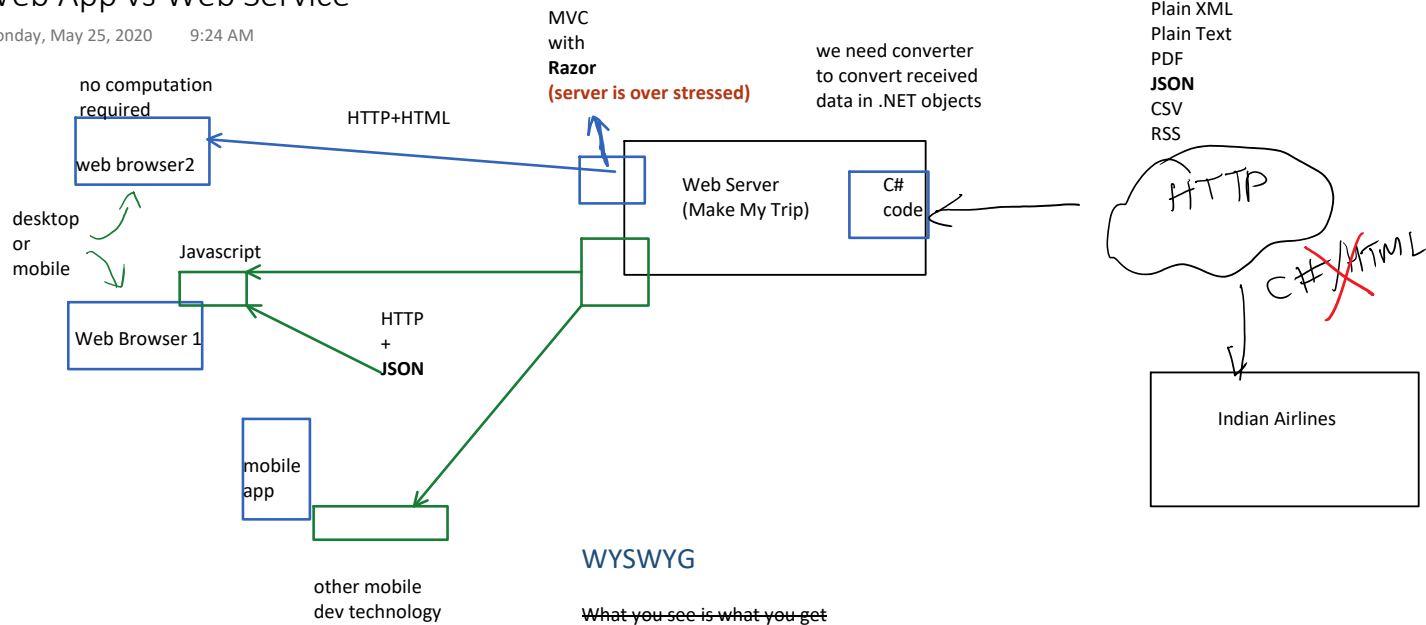
## Implementing REST Service





# Web App vs Web Service

Monday, May 25, 2020 9:24 AM



## WYSWYG

What you see is what you get

Where you see and when you get?

- you develop on a 19 inch wide screen
- user may consume it on
  - 15"
  - 10"
  - 7"
  - 5"
  - 4.5"

# URL Formats

Monday, May 25, 2020 9:57 AM

## Asp.net MVC Url structure

<http://books.org/book/detail/rashmirathi> <----- detail is an action here not a resource

<http://books.org/book/create> <----- create is an action

<http://books.org/delete/rashmirathi> <---- our urls are representing (including) actions

## Rest URL Recommendation

- URL should be
  - Short
  - Simple (Easy to remember)
  - Permanent (Technology independent)
  - Representation of a **Resource**
    - or one of states (In C# term state means object property)
- URL Should not
  - Represent an **action** (create/delete/list)
    - we should use HTTP methods GET/POST/PUT/DELETE
  - Include technology or platform id
    - <https://books.org/details.aspx?author=jeffrey-Archer>
    - <https://m.books.org> <--- mobile device
  - Avoid Including formats
    - <https://books.org/author/list/json> or <https://books.org/author/list?type=json>
      - Many services actually include this type element
  - Avoid localization formats
    - <https://books.org/author/list?lang=hi>

## Some Examples

- To Create an Author
  - **POST /authors**
  - The new crated author should have its own url
  - the url should be returned in **location** header
  - response status code should be 201
- To get a list of all Authors
  - **GET /authors**
  - **ACCEPT: application/json** <---specifiy the format of expected data
  - **UserAgent: Android Browser** <---- speicfiy the mobile format if required
  - **ACCEPT-LANGUAGE: hi** <----- speicfy the localization parameter
- To get a single author info
  - **GET /authors/jeffrey-archer**
  - user headers as above
- To update a single author info
  - **PUT /authors/jeffrey-archer**
  - **CONTENT-TYPE: application/json**
  - `\r\n`
  - `{ "Author": "John Grisham", "Biography": "...", ... }`
  - -----
  - Should return 202 on success
  -
- To delete a single author
  - **DELETE /authors/jeffrey-archer**
  - should delete the author info and the corresponding url
  - should return 204 on success.

Note while PUT/DELETE doesn't work with plain browser communication (when browser sends the request directly)

it works well with other clients such as

- C# client making a http request
- Javascript using XMLHttpRequest object to make Ajax call from within browser

XmlHttpRequest object is consider a brower within the browser and it knows how to send HTTP PUT/DELETE request.

## More REST URL examples (All below class are GET)

- <https://books.org/authors> <----- list of all authors
- <https://books.org/authors/jeffrey-archer> <--- details of jeffrey (no need of action word details)
- <https://books.org/authors/jeffrey-archer/books> <---- books by jeffrey archer
- <https://books.org/authors/jeffrey-archer/biography> <---- detailed biography of jeffrey-archer
- <https://books.org/authors/jeffrey-archer/reviews> <---- reviews about jeffrey archer

Urls that may not return Json but some data that can be used any where

- <https://books.org/authors/jeffrey-archer/photograp> <---- returns a image/png not JSON
- <https://books.org/authors/jeffrey-archer/email> <----- returns a text/plain containing just email id

``

`<a href="mailto:https://books.org/authors/jeffrey-archer/email">Mail Author</a>`

### Note

- If you image Author to be a class
  - Each URL may correspond to entire author object
  - or may be to one of the property (state) of Jeffrey Archer

```
class Author{
    String Name;
    String Email;
    String Photograph;
    IList<Book> books;
    IList<Review> reviews;
```

Representational State Transfer

}

### Why not MVC api

- MVC api depends on Action, REST has no Action
- REST needs to depend on
  - HTTP verbs
  - Accept header
- We may need to generate different data type based on ACCEPT header
  - MVC isn't designed for this out of box
  - I can use MVC to read header and take action, but MVC isn't designed for it
  - We don't want to manually decide and return JSON/XML
- We need api that can
  - Process HTTP header and methods to deliver result in desired format.

### Microsoft Approach to WebApi (.NET Framework)

- Create a new **Controller Hierarchy called ApiController**
- Created new ControllerFactory, ControllerActivator, ActionFilter etc
- The new namespace is System.Web.Http
- And we have an entire parallel hierarchy with same idea but different implementation
- So we have two sets of
  - Controller base class
  - Controller Factory
  - Controller Activator
  - ActionFilterAttribute

### Asp.net Core

- Has a unified api to serve both MVC and WebApi request
- Both (optionally) inherit same Controller class
- **A single api to serve all the requests**