C# .NET BOOTCAMP

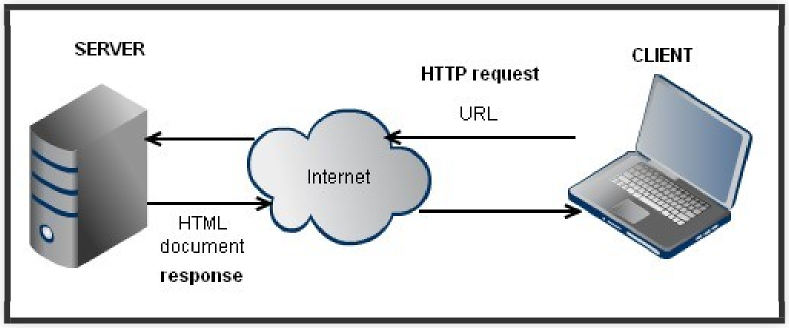
ASP.NET

INTRODUCTION TO ASP.NET

WHAT IS A DYNAMIC WEB APPLICATION?

- Contents of pages can change based on parameters.
- They run on application servers.
- Common technologies: : ASP.NET, JSP, php, Ruby on rails.

DYNAMIC WEB APPLICATIONS



WHAT IS ASP.NET?

- ASP stands for: Active Server Pages.
- ASP.NET is a server side web application framework that is used to develop dynamic web applications.

WHAT IS ASP.NET?

- First release was back in 2002. Current release is ASP.NET Core 1.0, but we will use ASP.NET 4.6 (more mature and tested)
- Can be programmed using any .NET framework programming language.

ASP.NET EXTENSIONS

ASP.NET SUPPORTS MANY WEB TECHNOLOGIES AND EXTENSIONS SUCH AS:

- Web Forms
- MVC
- WebAPI
- Razor

RUNING ASP.NET APPLICATIONS

Running web applications requires a web server.

We will use Micorsoft IIS to host ASP.NET applications.

Visual Studio has IIS Express, which can help developers program ASP.NET web applications without setting up a full IIS.

ASP.NET MVC

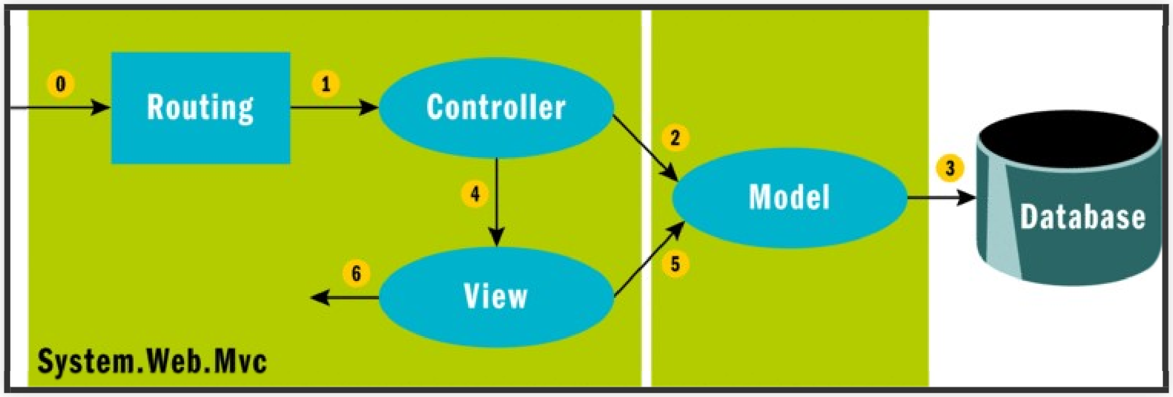
The ASP.NET MVC is ASP.NET's implementation of the MVC pattern.

The Model will implement the business layer.

The View will implement the display layer.

The Controller will implement the interactions and input control.

ASP.NET MVC ARCHITECTURE



WALK THROUGH

CREATE YOUR FIRST ASP.NET MVC APPLICATION!

ASP.NET MVC APPLICATION LIFE CYCLE

ROUTING

Routing is the first step in MVC request cycle.

The Routing engine looks in the Routing table to find matches for the requested URL.

When a match is found, the request is then sent to the corresponding IRouteHandler.

The default one calls the MvcHandler.

The routing engine returns a 404 HTTP status code against that request if the patterns is not found in the Route Table.

ASP.NET MVC APPLICATION LIFE CYCLE

ROUTING

When a match is found, the request is then sent to the corresponding IRouteHandler.

The default one calls the MvcHandler.

The routing engine returns a 404 HTTP status code against that request if the patterns is not found in the Route Table.

ASP.NET MVC APPLICATION LIFE CYCLE

ROUTING

The routing engine returns a 404 HTTP status code if there are no matches.

ASP.NET MVC APPLICATION LIFE CYCLE

MVC HANDLER

The MvcHandler is where we will process the request.

MVC handler implements IHttpHandler interface.

ASP.NET MVC APPLICATION LIFE CYCLE

CONTROLLER

Here, the MvcHandler will try to get a IController instance.

After getting an instance, the Execute method will be called.

ASP.NET MVC APPLICATION LIFE CYCLE

ACTION EXECUTION

After creating the controller , the ActionInvoker chooses which action to invoke on the controller.

ASP.NET MVC APPLICATION LIFE CYCLE

VIEW RESULT

After getting the user input, the action method will prepare the appropriate response data, and then executes the result by returning a result type.

ASP.NET MVC APPLICATION LIFE CYCLE

VIEW RESULT

The result type can be ViewResult, RedirectToRouteResult, RedirectResult, ContentResult, JsonResult, FileResult, and EmptyResult.

ASP.NET MVC APPLICATION LIFE CYCLE

VIEW ENGINE

The IViewEngine selectes the appropriate View Engine to render the View Result.

ASP.NET MVC APPLICATION LIFE CYCLE

VIEW

The last step is to return string,a binary file or a Json data.

ASP.NET MVC PROJECT LAYOUT IN VISUAL STUDIO'S SOLUTION EXPLORER

Follow your instructor to go over the MVC project in Visual Studio

STATE MANAGEMENT

WHAT IS STATE MANAGEMENT?

HTTP is a stateless protocol, which means that the web server does not retain status about each user for the duration of multiple requests (each request is independent).

The idea is to enhance performance and make web applications scalable.

WHAT IS STATE MANAGEMENT?

However, we need some way to save infromation between requests.

Example: Moving contents from one page to the next.

Information can be stored on the server (sessions) or the client's machine (cookies).

PASSING DATA BETWEEN THE CONTROLLER AND THE VIEW:

VIEWDATA

ViewData is a dictionary object where we can put data in that will be available to the view

PASSING DATA BETWEEN THE CONTROLLER AND THE VIEW:

VIEWBAG

The ViewBag is a wrapper around the ViewData object that allows you to create dynamic properties.

PASSING DATA BETWEEN THE CONTROLLER AND THE VIEW:

TEMPDATA

TempData is also a dictionary. The difference is the life cycle of the object.

TempData keeps the information for the time of an HTTP Request.

This mean only from one page to another.

PASSING DATA BETWEEN THE CONTROLLER AND THE VIEW:

TEMPDATA

TempData keeps the information for the time of an HTTP Request.

SESSIONS

A Session is also used to pass data within the ASP.NET MVC application

It will reside for a long time compared to TempData.

It is valid for all requests.

COOKIES

Cookies composed of text and it is stored on the client's computer to be used only by the website setting that cookie.

Cookies allow web applications to save information for users, and then they can be used on each page if needed.

SESSION AND COOKIES DEMO

DEMO!

RECAP

What you should know at this point:

- What is State Management
- How can we move data between the controller and the View
- What is a Session, and how to use it.
- What is a cookie, and how to use it.

NUGET

NUGET

NuGET is a package manager that enables programmers to produce and consume packages.

It is free and open source.

Pre-installed on Visual Studio.

NUGET

Example packages:

- Entity Framework.
- Bootstrap css
- JQuery
- JSON .NET

NUGET DEMO

We will show how to install packages from NuGET

RECAP

What you should know at this point:

- What is NuGET?
- How to use NuGET.

ENTITY FRAMEWORK

ENTITY FRAMEWORK (EF)

Entity Framework (EF) is an open-source object-relational mapping framework that enables.

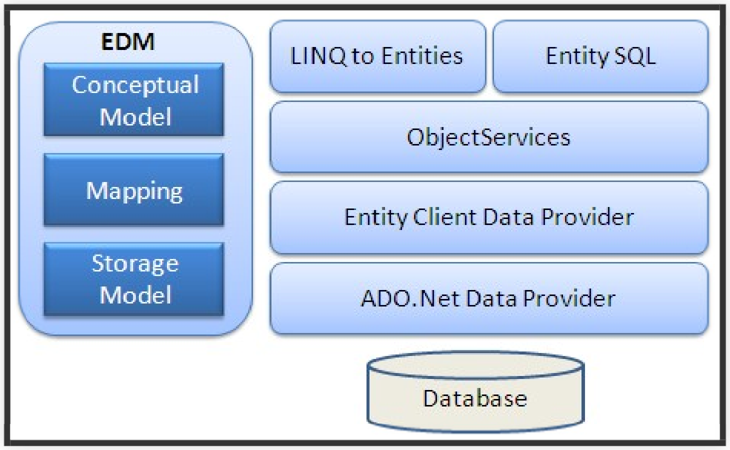
It is a part of ADO.NET.

ENTITY FRAMEWORK (EF)

EF enables programmers to model data entities and the relationships between them, and also model the business logic.

EF also provides abstraction, where developers don't need to worry about the underlying database tables and columns where this data is stored.

ENTITY FRAMEWORK (EF) ARCHITECTURE



ENTITY FRAMEWORK (EF) SCENARIOS

- If the DB already exists, or you want to start first with DB design.
- If the focus is on the domain classes and then the DB will be created from those classes.
- Use the DB schema designer to design your DB, and then create the DB and classes.

ENTITY FRAMEWORK (EF) SETUP

Use NuGET to setup EF.

Follow your instructor to setup EF.

CODE FIRST APPROACH

Write your classes first and then create database from these classes.

If you want to follow Domain-Driven Design (DDD) principles, start by coding the domain classes first and then generate the database.

CODE FIRST APPROACH

DEMO!

MODEL FIRST APPROACH

Here, you need to create Entities, relationships, and inheritance hierarchies directly using the model designer, and then that model to generate your database.

MODEL FIRST APPROACH

DEMO!

RECAP

What you should know at this point:

- What is Entity Framework?
- How to setup EF.
- EF Scenarios

LINQ

WHAT IS LINQ?

LINQ stands for Language Integrated Query.

This Microsoft query language offer an easy way to access data in in-memory objects, databases, XML documents and many more.

SYNTAX OF LINQ

There are two main ways to write a LINQ query:

- Query Syntax or Query Expression Syntax
- Method Syntax or Method extension syntax

SYNTAX OF LINQ

QUERY SYNTAX OR QUERY EXPRESSION SYNTAX

```
ange variable}in {collection}
, joining, grouping, aggregate operators etc.} {lambda

 or groupBy operator} {formulate the result}
```

SYNTAX OF LINQ

QUERY SYNTAX OR QUERY EXPRESSION SYNTAX EXAMPLE

```
enAgerStudent = from s in studentList
        where s.Age > 12 && s.Age < 20
        select s;
```

SYNTAX OF LINQ

METHOD SYNTAX

Method syntax (also known as fluent syntax) uses extension methods

Example: Where, ToList

SYNTAX OF LINQ

METHOD SYNTAX EXAMPLE

```
enAgerStudents =
tList.Where(s => s.Age > 12 && s.Age < 20)
                    .ToList();
```

MORE LINQ EXAMPLES

Follow your instructor to do more examples!

RECAP

What you should know at this point:

- What is Linq?
- Ways to write Linq.

ASP.NET IDENTITY

WHAT IS ASP.NET IDENTITY?

ASP.NET Identity is a membership system that makes it easy to implement profiles and add Login/ LogOut functionality to the application.

ASP.NET Identity is can be downloaded as a NuGet package.

WHAT IS ASP.NET IDENTITY?

ASP.NET Identity uses OWIN Authentication for log-in/log-out of users in the web site instead of using Forms Authentication.

ASP.NET IDENTITY PACKAGES

Microsoft.AspNet.Identity.EntityFramework

Microsoft.AspNet.Identity.Core

Microsoft.AspNet.Identity.OWIN

DEMO

Follow your instructor to create a website that uses Identity to secure the application.

WEB API

WHAT IS ASP.NET WEB API?

ASP.NET Web API is used to develop and consume HTTP services.

They use mainly XML or JSON to send and recieve information.

CREATING ASP.NET WEB APIS

We can use MVC to build Web APIs.

Web APIs can be downloaded from NuGET.

Visual Studio has project templates to create Web APIs

WHY DO WE NEED WEB APIS?

Can be used as a service layer for different kinds of applications.

WEB API DEMO

Follow your instructor to create and use a Web API using Visual Studio.

MICROSOFT AZURE

MICROSOFT AZURE PLATFORM

Microsoft Azure is a cloud computing platform that is used to host and manage applications.

Microsoft Azure can be used to host not only .NET applications.

Visual Studio can be used to host applications on the Azure cloud.

MICROSOFT AZURE PLATFORM

Microsoft Azure competes with Amazon's AWS in the cloud computing market.

It also provides many services including data storage, streaming, and analytics.

DEMO

Follow Your instructor to host an ASP.NET MVC web application on Azure.