

RESTFUL WEB SERVICE VÀ WEB API

❖ Tóm tắt lý thuyết

GIỚI THIỆU VỀ WEB SERVICE



❖ Web Services là gì?

- Là một hệ thống phần mềm được thiết kế để hỗ trợ sự tương tác giữa các máy tính trên mạng.
- Các phần mềm được viết bởi các ngôn ngữ lập trình khác nhau và chạy trên các nền tảng khác nhau có thể sử dụng dịch vụ Web để chuyển đổi dữ liệu thông qua mạng dưới dạng XML/JSON.
- WS có khả năng chạy trên những hệ điều hành và framework khác nhau.

GIỚI THIỆU VỀ WEB SERVICE



❖ Web Services là gì?

- Một Web service được tạo nên bằng cách lấy các chức năng và đóng gói chúng sao cho các ứng dụng khác dễ dàng nhìn thấy và có thể truy cập đến những dịch vụ mà nó thực hiện, đồng thời có thể yêu cầu thông tin từ Web service khác.
- WS bao gồm các mô đun độc lập cho hoạt động của khách hàng và doanh nghiệp và bản thân nó được thực thi trên server.
- Mục tiêu của Web services (WS) là cung cấp một số chức năng của nó (có thể là của một cá nhân hay tổ chức)

❖ Ứng dụng của Web Services

- Các nhà hàng, khách sạn cung cấp các WS cho phép đặt phòng, đặt tiệc.
- Đường sắt Việt Nam cung cấp các WS cho phép đặt vé tàu.
- Hàng không Việt Nam cung cấp các WS cho phép đặt vé cho các chuyến bay.

🔷 Giới thiệu về RESTful Web Service

- ✓ RESTful Web Service là các Web Service sử dụng HTTP, được viết dựa trên kiến trúc REST.
- ✓ **REST** (**RE**presentational **S**tate **T**ransfer) đã được sử dụng rộng rãi thay thế cho các Web Service dựa trên SOAP.
- ✓ RESTful Web Service đơn giản hơn, dễ dàng mở rộng và bảo trì.
- ✓ REST là một bộ quy tắc để tạo ra một ứng dụng Web Service



Giới thiệu về Web API

✓ Web API là gì?

- Web API (API- Application Programming Interface) là các RESTful Web service được xây dựng dựa theo mô hình lập trình convention (như ASP.NET MVC).
- **Web API** là một phương thức dùng để cho phép các ứng dụng khác nhau có thể giao tiếp, trao đổi dữ liệu qua lại.
- Dữ liệu được Web API trả lại thường ở dạng JSON hoặc XML.

XÂY DỰNG WEB API

Bài thực hành mẫu

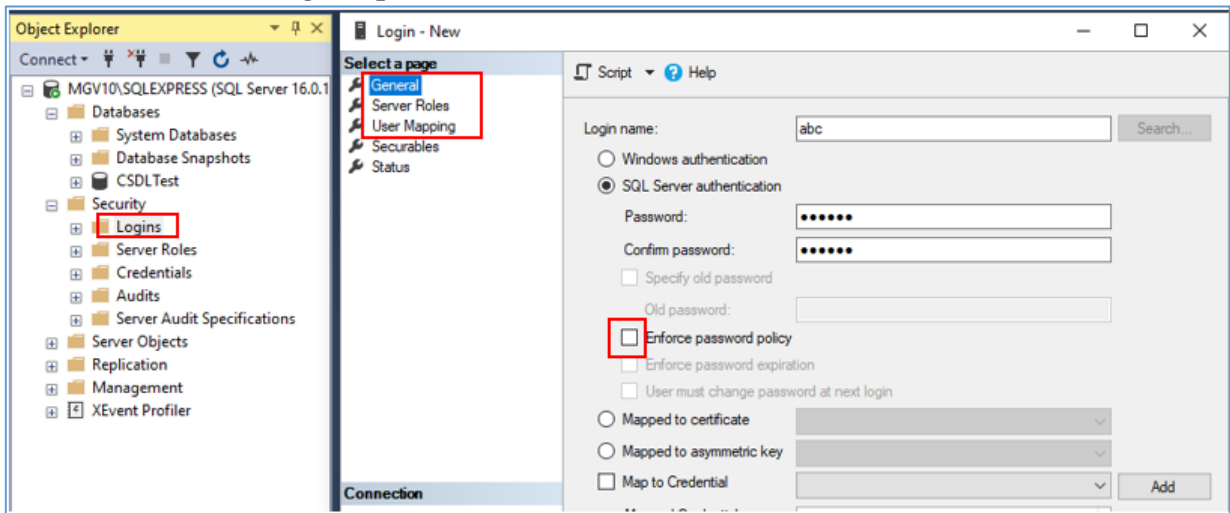
B1. Trước tiên ta tạo cơ sở dữ liệu **CSDLTest** gồm 2 bảng:

- + Đăng nhập Windows Authentication để tạo CSDLTest
 - DanhMuc(**MaDanhMuc** - int, TenDanhMuc-nvarchar(20))
 - SanPham(**Ma** - int, Ten – nvarchar(20), DonGia - int, MaDanhMuc - int)

+ Nhập dữ liệu cho 2 bảng

(Tạo các bảng bằng Click chuột – Nếu tạo các bảng bằng code query analys thì xem chú ý cuối bài).

B2. Tạo tài khoản đăng nhập SQL Server: SQL Server Authentication



+ Click chuột phải vào **logins**, chọn **New Login**

+ General:

- Login name: abc
- Chọn SQL Server Authentication: pass:123456, repass
- Bỏ chọn policy

+ Server Roles:

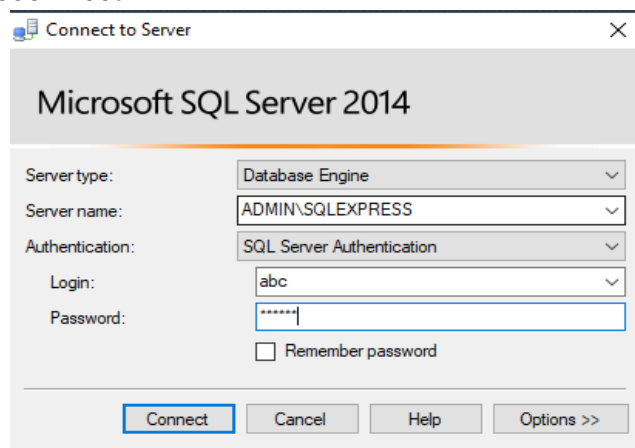
- Chọn: sysadmin

+ Use mapping: chọn CSDLTest

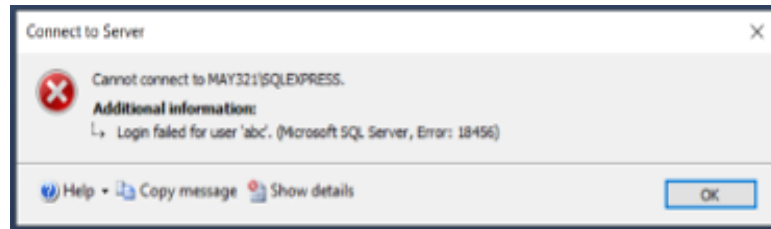
+ Sau khi tạo xong Login mới:

=>**Disconnect** với Windows Authentication

=>Đăng nhập lại: **reconnect**

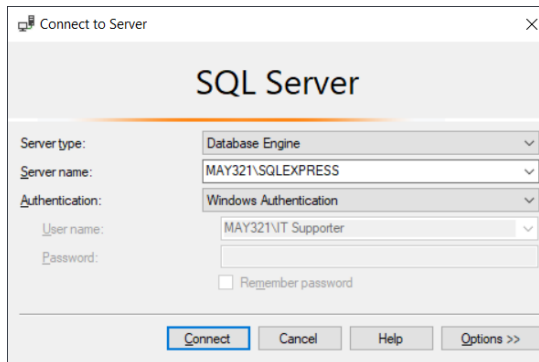


(Nếu có lỗi sau thì phải sửa lỗi theo hướng dẫn)

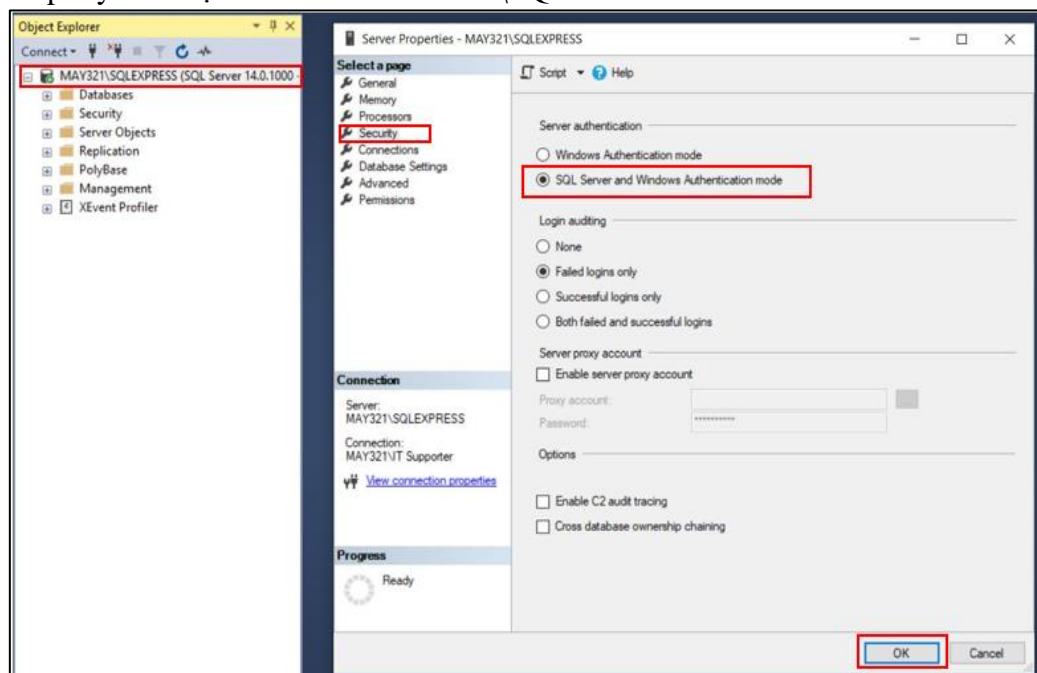


❖ Xử lý lỗi login với account Authentication SQL Server:

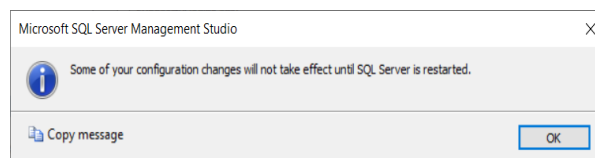
Đăng nhập lại với **Windows Authentication** mode như ban đầu:



Chọn Property cho mục tên Server MAY321\SQLEXPRESS

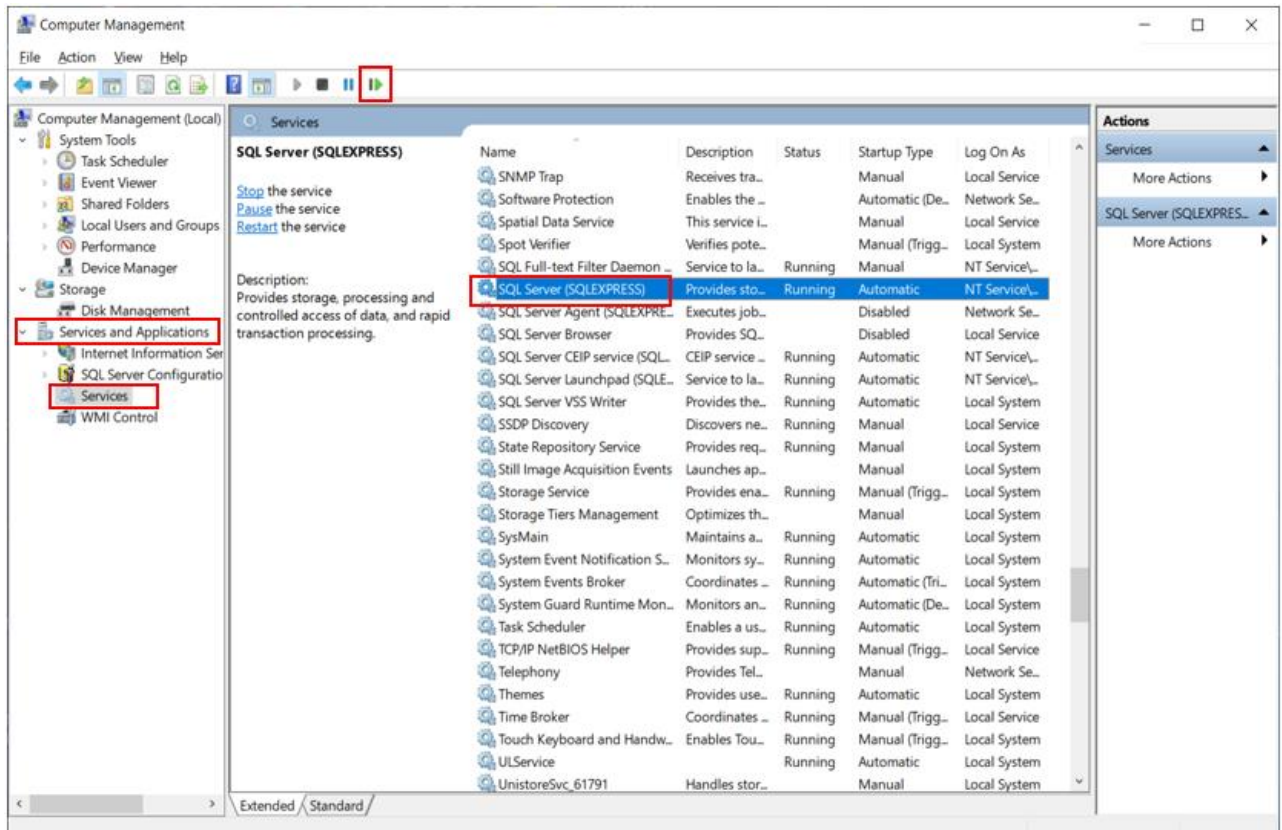


Click ok, chờ 1 lát:

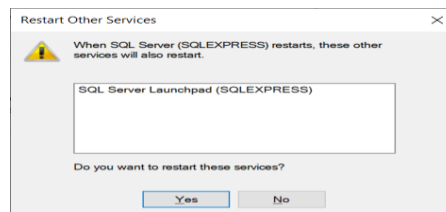


=>Click OK.

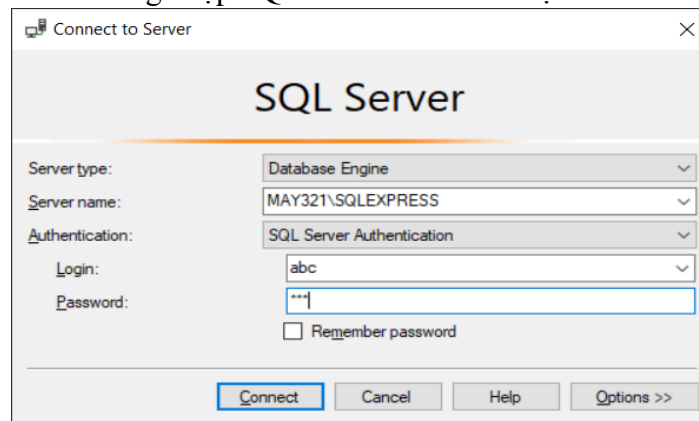
Tìm **This PC**, click chuột phải, Chọn thuộc tính **Manage**, Mở **Services**: Ấn S để tìm thấy SQL EXPRESS:



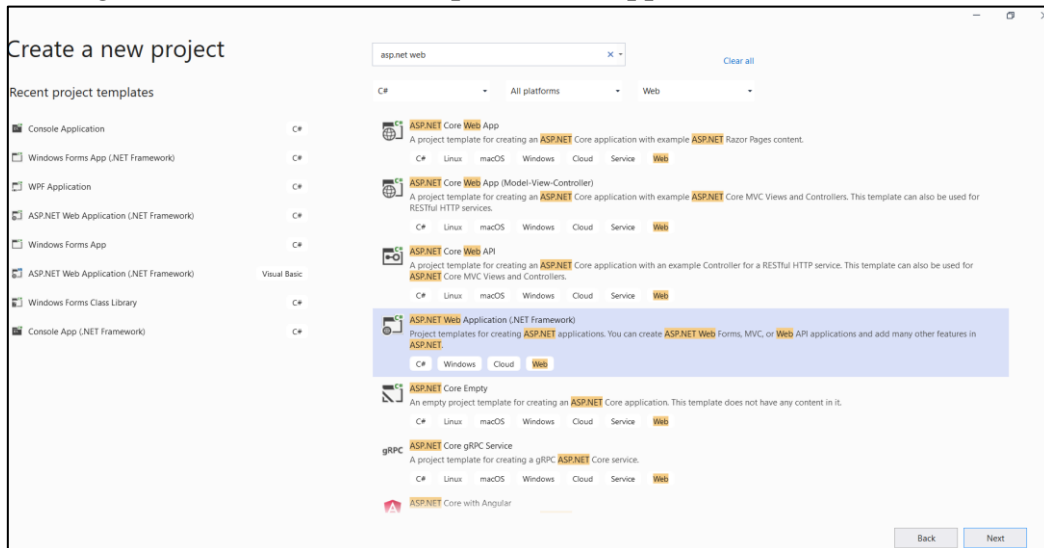
Click vào nút xanh phía trên để Restart.



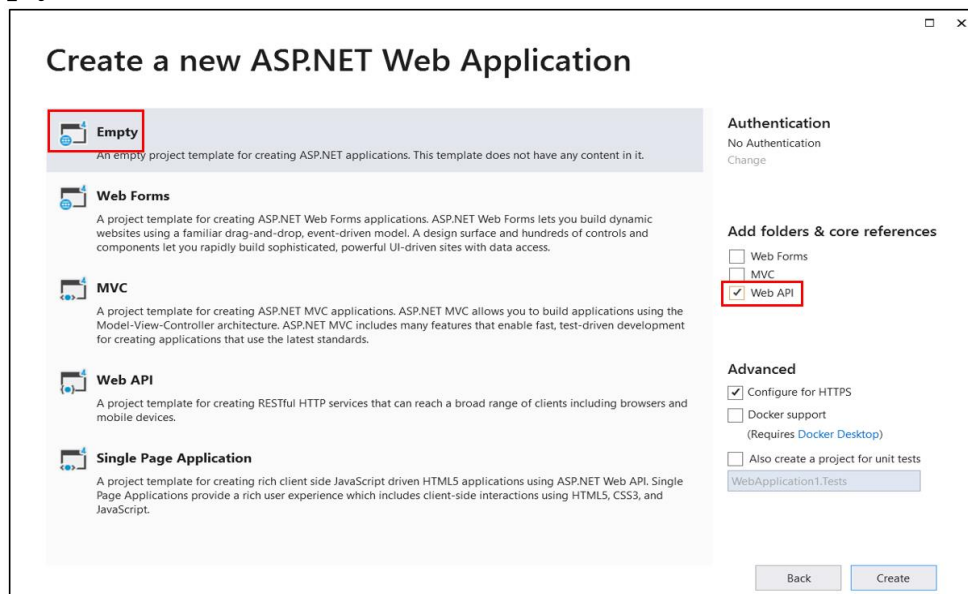
Disconnect SQL Server và đăng nhập SQL với user **abc** vừa tạo => OK



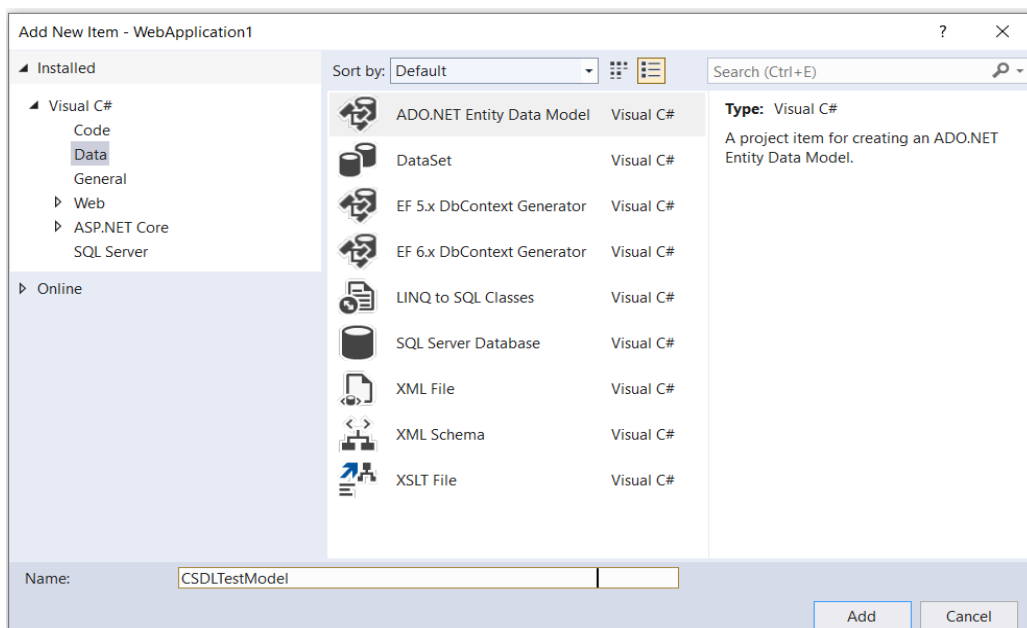
B3. Khởi động Visual Studio, Chọn Asp.net Web Application



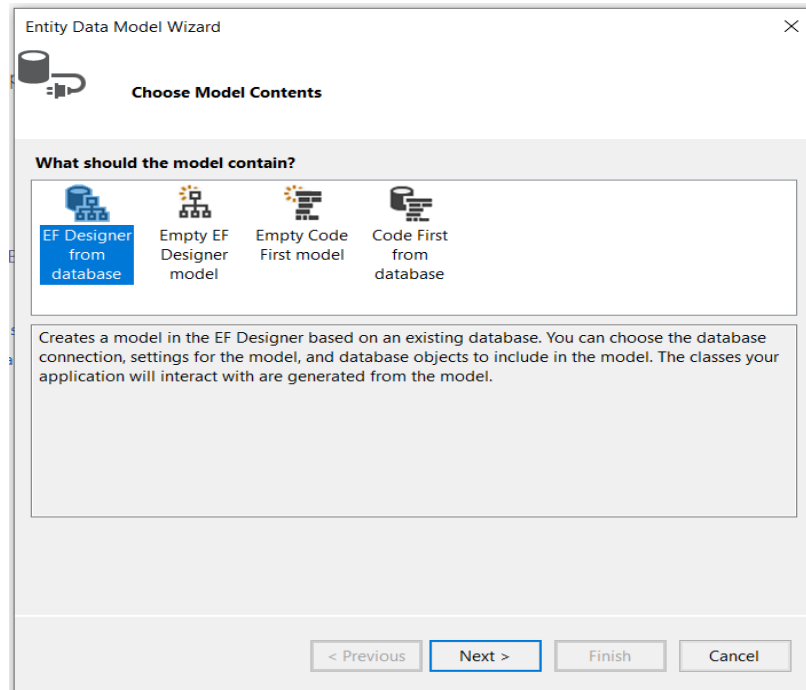
Chọn **Empty** và check **web API**



B4. Tạo model: Ở Models, Add ->New Item->Data->ADO.Net Entity Data Model, đặt tên cho Model:

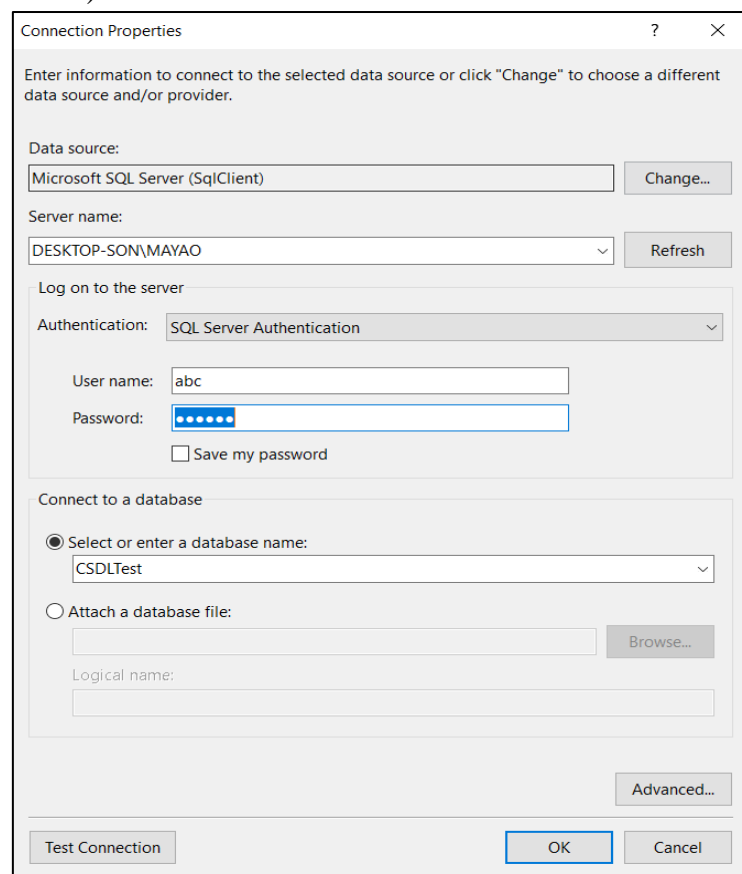


=> Kết nối Model tới CSDL:



- Chọn New connection

- Lấy server name từ SQL Server, kết nối dạng sql Server Authentication (với User name:abc; Pass:123456)



- Nó sinh ra 1 entities framework

Entity Data Model Wizard

Choose Your Data Connection

Which data connection should your application use to connect to the database?

desktop-son\mayao.CSDLTest.dbo1 New Connection...

This connection string appears to contain sensitive data (for example, a password) that is required to connect to the database. Storing sensitive data in the connection string can be a security risk. Do you want to include this sensitive data in the connection string?

☐ No, exclude sensitive data from the connection string. I will set it in my application code.

☒ Yes, include the sensitive data in the connection string.

Connection string:

```
metadata=res://*/Models.CSDLTestModel.csd|res://*/Models.CSDLTestModel.ssd|
res://*/Models.CSDLTestModel.msl;provider=System.Data.SqlClient;provider connection
string="data source=DESKTOP-SON\MAYAO;initial catalog=CSDLTest;user
id=abc;password=*****;MultipleActiveResultSets=True;App=EntityFramework"
```

☒ Save connection settings in Web.Config as:

CSDLTestEntities

< Previous **Next >** Finish Cancel

- Chọn hết các bảng

Entity Data Model Wizard

Choose Your Database Objects and Settings

Which database objects do you want to include in your model?

☒ Tables

☒ dbo

☒ DanhMuc

☒ SanPham

☐ Views

☐ Stored Procedures and Functions

☒ Pluralize or singularize generated object names

☒ Include foreign key columns in the model

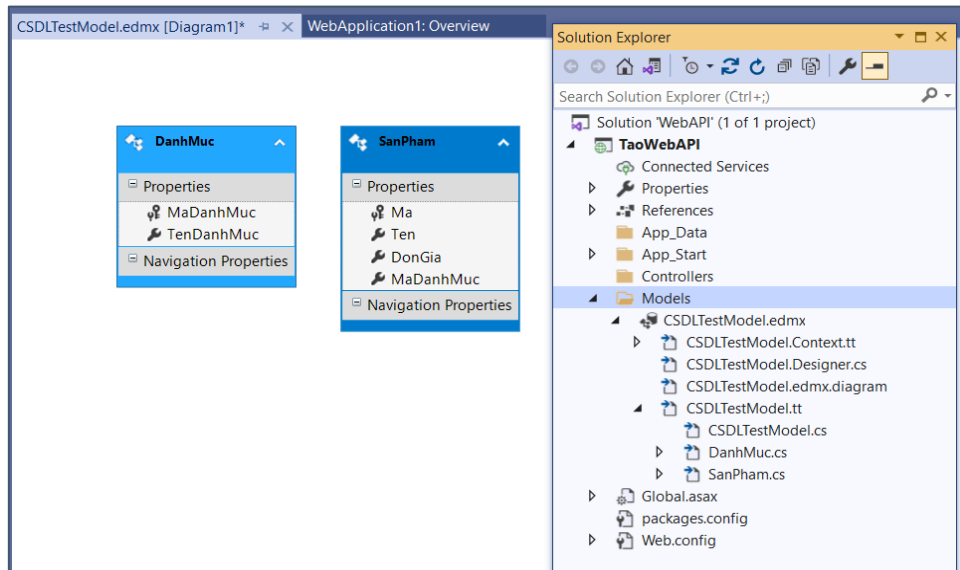
☐ Import selected stored procedures and functions into the entity model

Model Namespace:

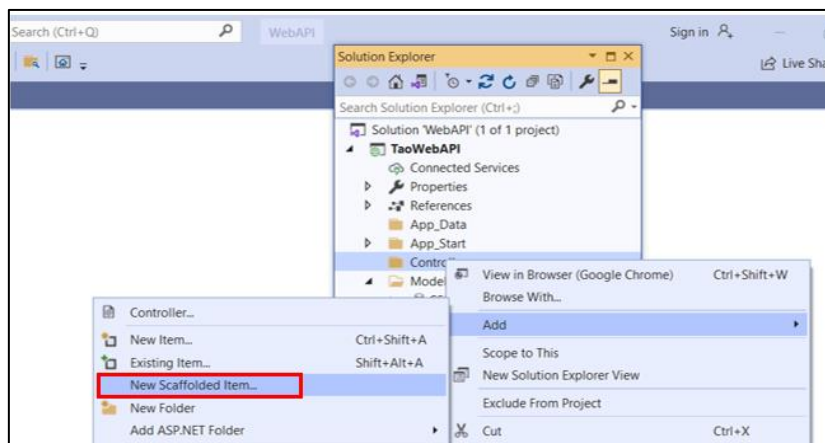
CSDLTestModel

< Previous **Next >** **Finish** Cancel

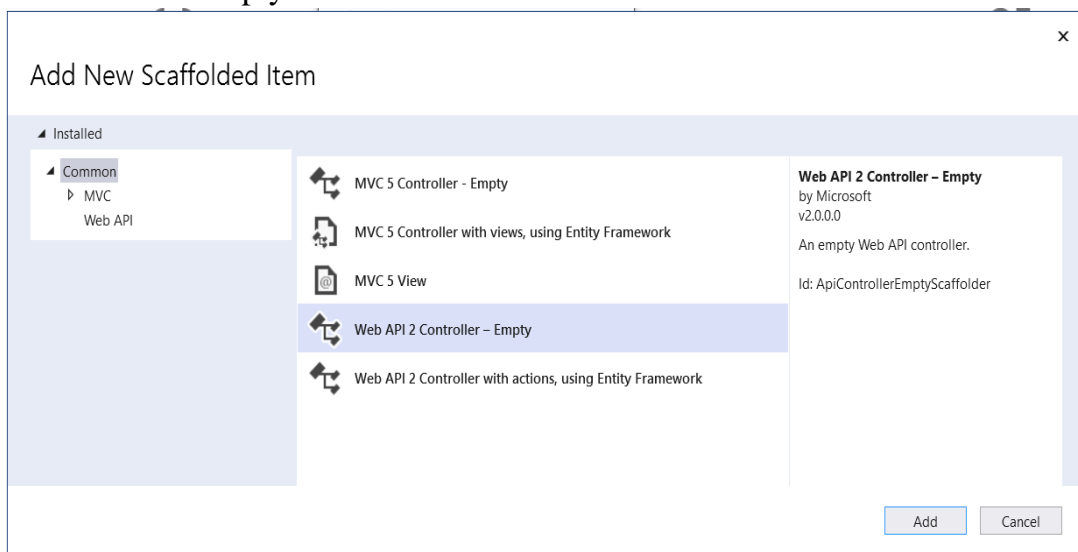
- Models sinh ra:



B5. Tạo các lớp controllers: SanPham, DanhMuc



- Chọn Web API 2 Empty



- Đặt tên cho các Controller

Add Controller

Controller name:
SanPhamController

Add
Cancel

- Tương tự tạo Controller cho bảng **DanhMuc**

B6. Tạo các Web API

```

7      using TaoWebAPI.Models;
8      namespace TaoWebAPI.Controllers
9      {
10         0 references
11         public class SanPhamController : ApiController
12         {
13             CSDLTestEntities db = new CSDLTestEntities();|
14             [HttpGet]//Lấy dữ liệu
15             0 references
16             public List<SanPham> LaySP()
17             {
18                 return db.SanPhams.ToList();
19             }
20             [HttpGet]
21             0 references
22             public List<SanPham> TimSPTheoDanhMuc(int madm)
23             {
24                 return db.SanPhams.Where(x => x.MaDanhMuc == madm).ToList();
25             }
26             [HttpGet]
27             0 references
28             public SanPham TimSPTheoMa(int ma)
29             {
30                 return db.SanPhams.FirstOrDefault(x => x.Ma == ma);
31             }
32             [HttpPost]//Thêm dữ liệu
33             0 references
34             public bool ThemMoi(int ma, string ten, int gia, int madm)
35             {
36                 SanPham sp = db.SanPhams.FirstOrDefault(x => x.Ma== ma);
37                 if (sp == null)
38                 {
39                     SanPham sp1 = new SanPham();|
40                     sp1.Ma = ma;
41                     sp1.Ten = ten;
42                     sp1.DonGia = gia;
43                     sp1.MaDanhMuc = madm;
44                     db.SanPhams.Add(sp1);
45                     db.SaveChanges();
46                     return true;
47                 }
48                 return false;
49             }
50         }
51     }

```

```

45 [HttpPut]//Sửa dữ liệu
    0 references
46 public bool CapNhat(int ma, string ten, int gia, int madm)
47 {
48     SanPham sp = db.SanPhams.FirstOrDefault(x => x.Ma == ma);
49     if (sp != null)
50     {
51         sp.Ma = ma;
52         sp.Ten = ten;
53         sp.DonGia = gia;
54         sp.MaDanhMuc = madm;
55         db.SaveChanges();
56         return true;
57     }
58     return false;
59 }

```

```

60 [HttpDelete]//Xóa dữ liệu
    0 references
61 public bool xoa(int id)
62 {
63     SanPham sp = db.SanPhams.FirstOrDefault(x => x.Ma == id);
64     if (sp != null)
65     {
66         db.SanPhams.Remove(sp);
67         db.SaveChanges();
68         return true;
69     }
70     return false;
71 }
72 }
73 }

```

```

using TaoWebAPI.Models;

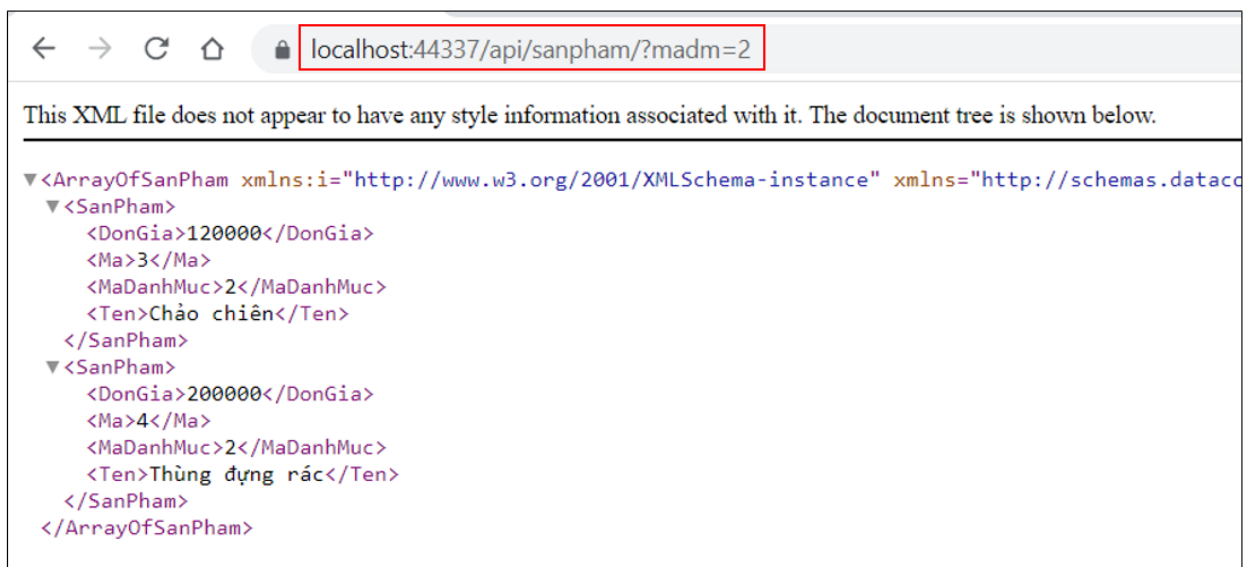
namespace TaoWebAPI.Controllers
{
    0 references
    public class DanhMucController : ApiController
    {
        CSDLTestEntities db = new CSDLTestEntities();
        [HttpGet]//Lấy dữ liệu
        0 references
        public List<DanhMuc> LayDM()
        {
            return db.DanhMucs.ToList();
        }
    }
}

```

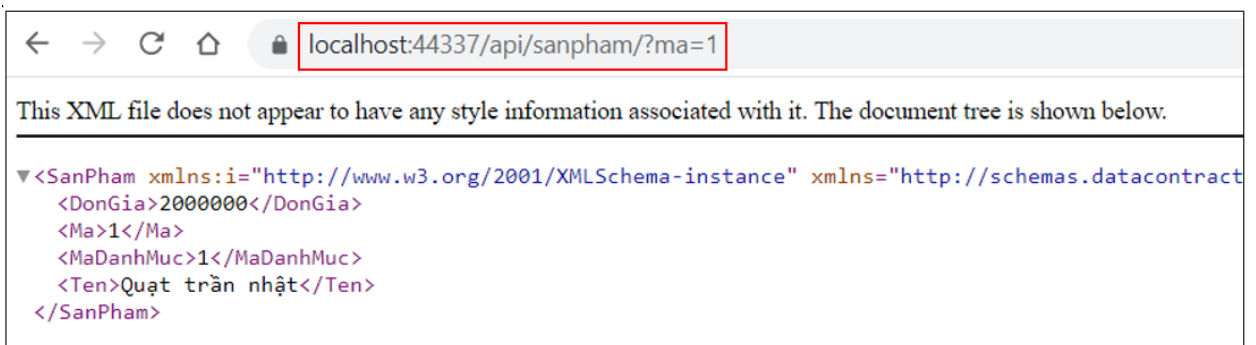
B7. Chạy thử các Web API (với [HttpGet])



```
<?xml version='1.0'>
<ArrayOfSanPham xmlns:i="http://www.w3.org/2001/XMLSchema-instance" xmlns="http://schemas.datacontract.org/2004/07/System.Data.DataContractSerializer">
  <SanPham>
    <DonGia>2000000</DonGia>
    <Ma>1</Ma>
    <MaDanhMuc>1</MaDanhMuc>
    <Ten>Quạt trần nhật</Ten>
  </SanPham>
  <SanPham>
    <DonGia>40000</DonGia>
    <Ma>2</Ma>
    <MaDanhMuc>1</MaDanhMuc>
    <Ten>Bóng đèn điện quang</Ten>
  </SanPham>
  <SanPham>
    <DonGia>120000</DonGia>
    <Ma>3</Ma>
    <MaDanhMuc>2</MaDanhMuc>
    <Ten>Chảo chiên</Ten>
  </SanPham>
  <SanPham>
    <DonGia>200000</DonGia>
    <Ma>4</Ma>
    <MaDanhMuc>2</MaDanhMuc>
    <Ten>Thùng đựng rác</Ten>
  </SanPham>
</ArrayOfSanPham>
```



```
<?xml version='1.0'>
<ArrayOfSanPham xmlns:i="http://www.w3.org/2001/XMLSchema-instance" xmlns="http://schemas.datacontract.org/2004/07/System.Data.DataContractSerializer">
  <SanPham>
    <DonGia>120000</DonGia>
    <Ma>3</Ma>
    <MaDanhMuc>2</MaDanhMuc>
    <Ten>Chảo chiên</Ten>
  </SanPham>
  <SanPham>
    <DonGia>200000</DonGia>
    <Ma>4</Ma>
    <MaDanhMuc>2</MaDanhMuc>
    <Ten>Thùng đựng rác</Ten>
  </SanPham>
</ArrayOfSanPham>
```

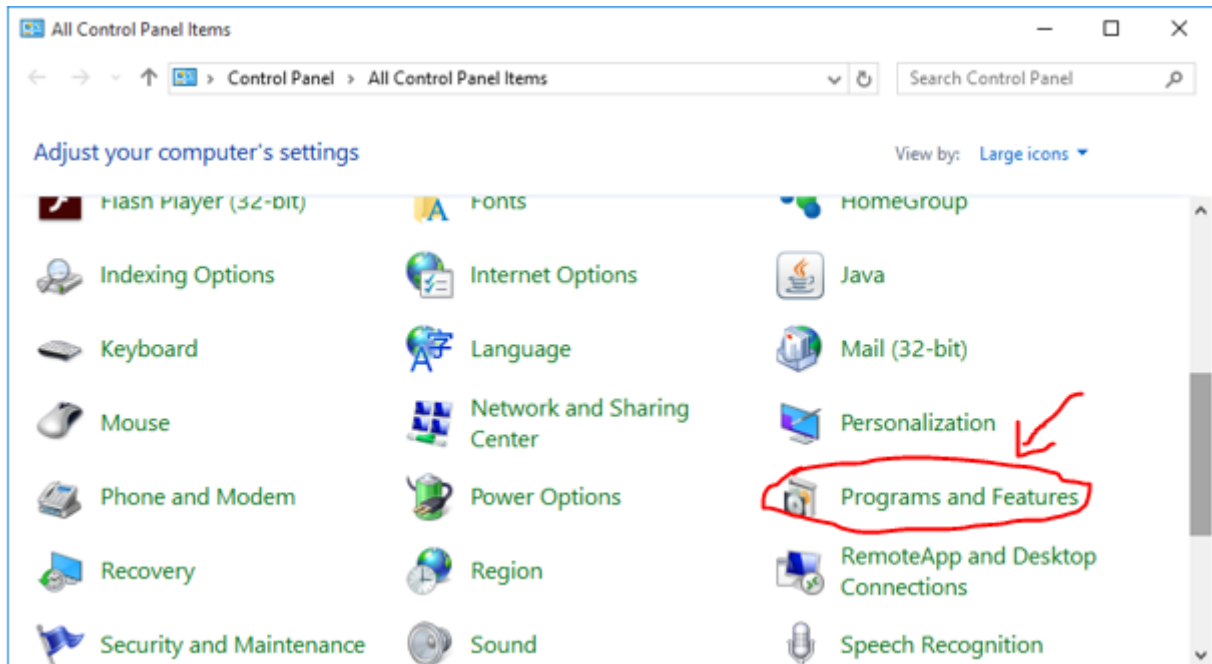


```
<?xml version='1.0'>
<SanPham xmlns:i="http://www.w3.org/2001/XMLSchema-instance" xmlns="http://schemas.datacontract.org/2004/07/System.Data.DataContractSerializer">
  <DonGia>2000000</DonGia>
  <Ma>1</Ma>
  <MaDanhMuc>1</MaDanhMuc>
  <Ten>Quạt trần nhật</Ten>
</SanPham>
```

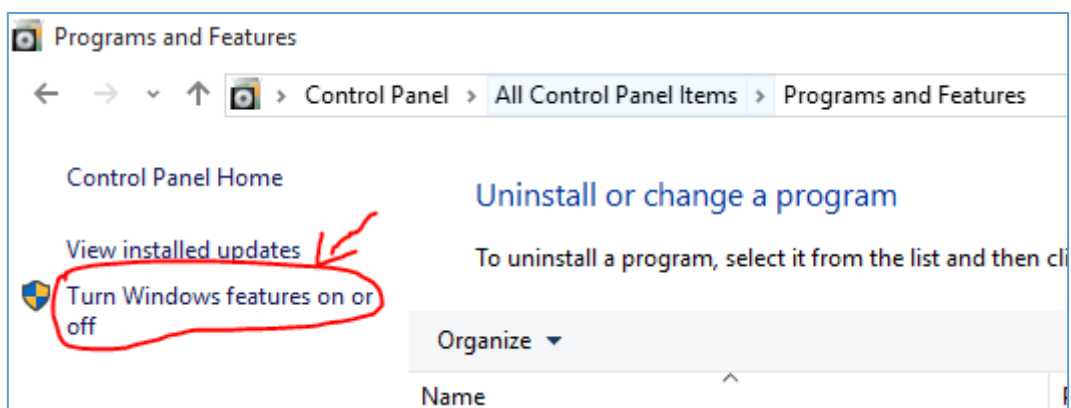
B8. Cài đặt các Web API lên IIS Web Server

B8.1. Cài đặt IIS

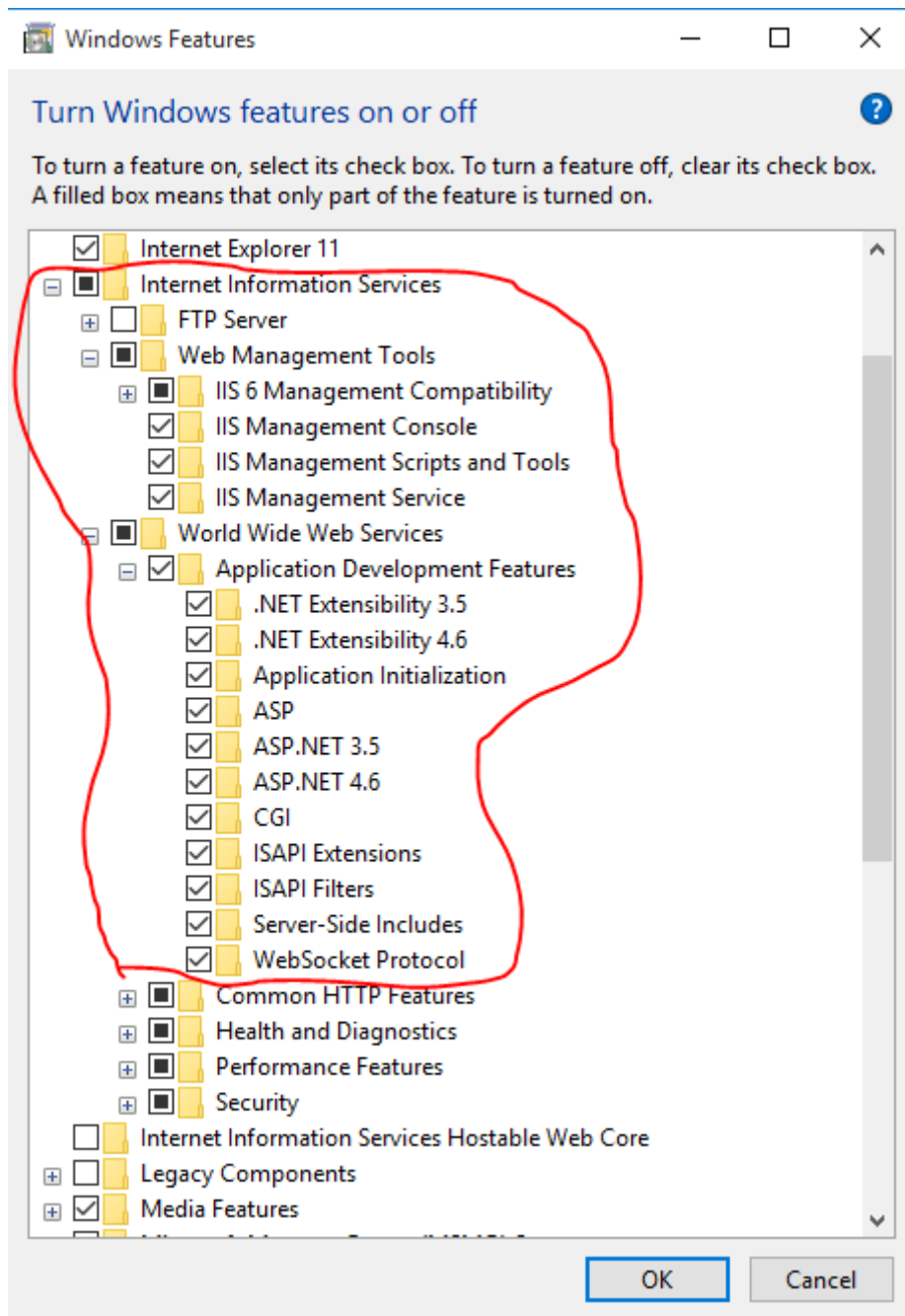
+ Vào Control panel chọn Programs and Features



+ Chọn Turn Windows features on or off:

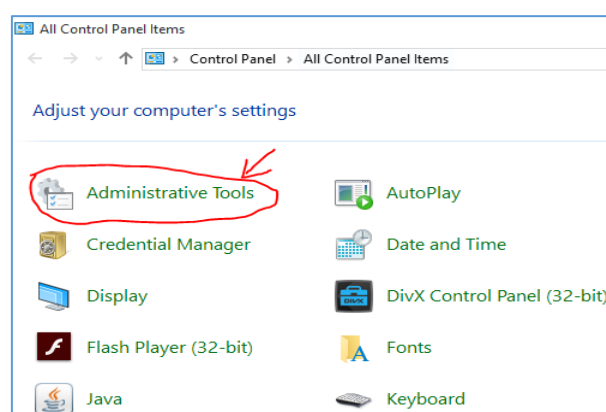


+ Tiến hành lựa chọn để cài IIS Web Server theo màn hình dưới đây:

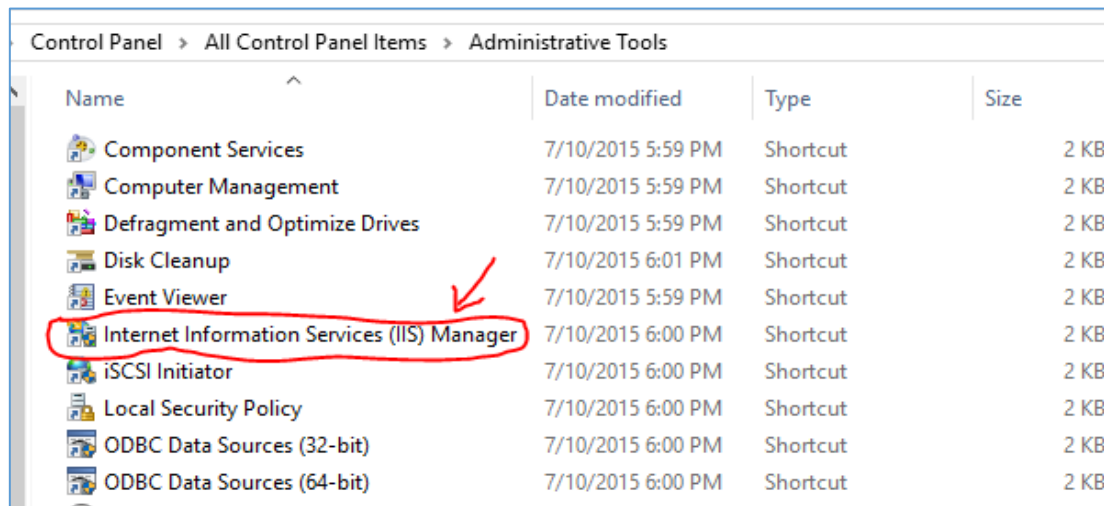


B8.2. Cài đặt Web API lên IIS Web Server

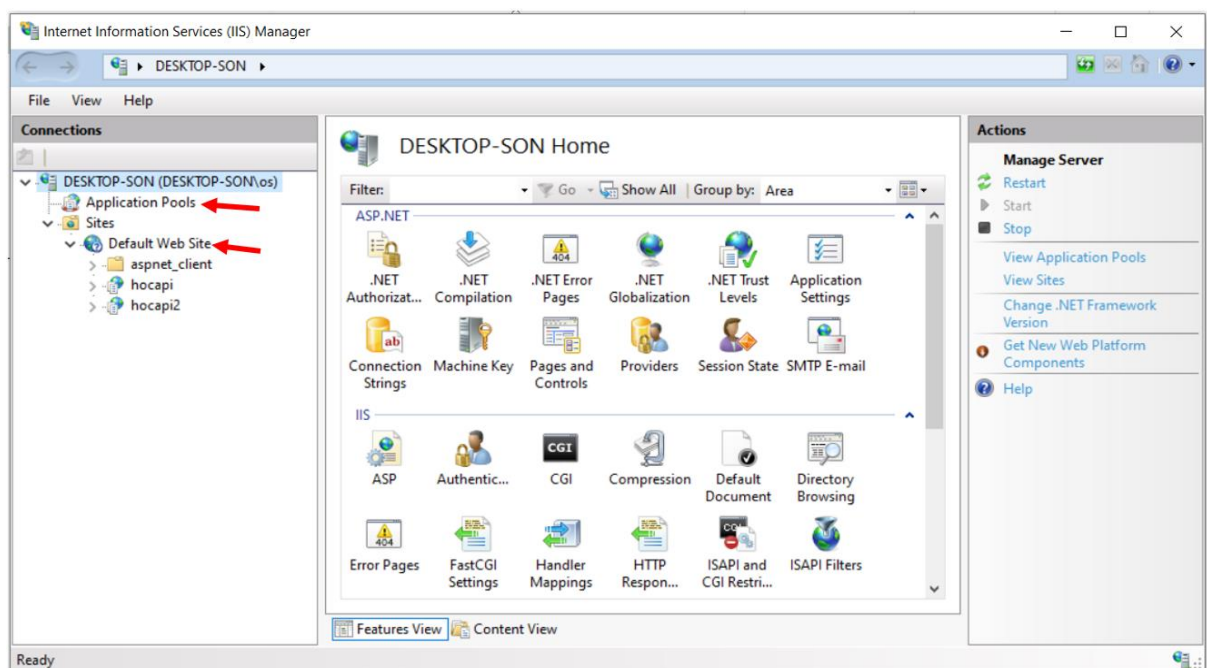
+ Sau khi cài xong IIS Web Server, bạn quay lại màn hình Control panel, bạn sẽ thấy Administrative Tools:



+ Ta click vào biểu tượng Administrative Tools, ta có giao diện tiếp theo:



+ Ta chọn “**Internet Information Services (IIS) Manager**” như hình trên:

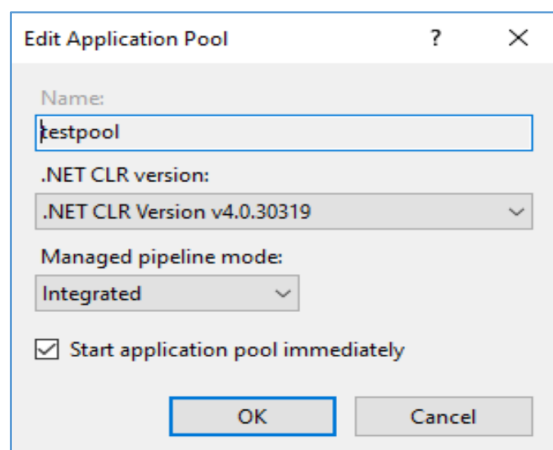


Trong màn hình quản trị bạn thấy có 2 vùng: Application Pools và Default Web site.

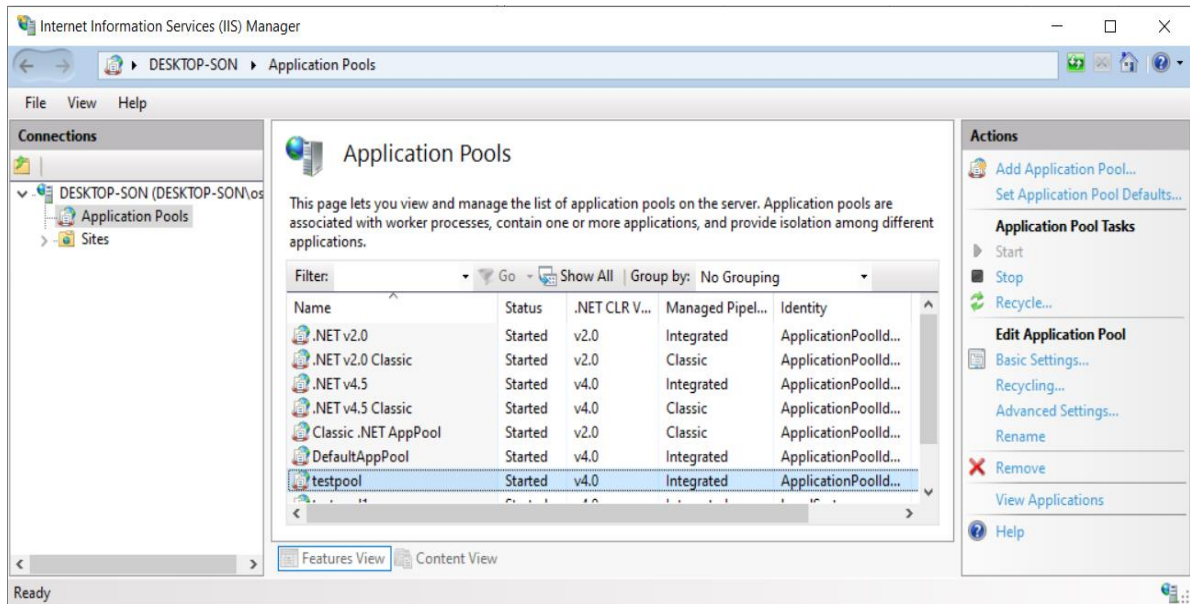
+ **Cấu hình Application Pools:**

Ta bấm chuột phải vào mục **Application Pools**/ chọn **Add Application Pool...**

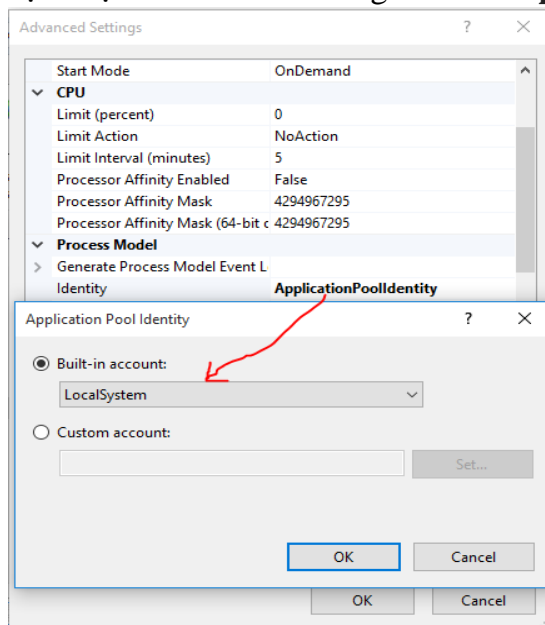
Màn hình tạo mới Application Pool hiển thị lên, ta nhập “testpool” rồi bấm OK:



Sau khi bấm OK, bạn quan sát:



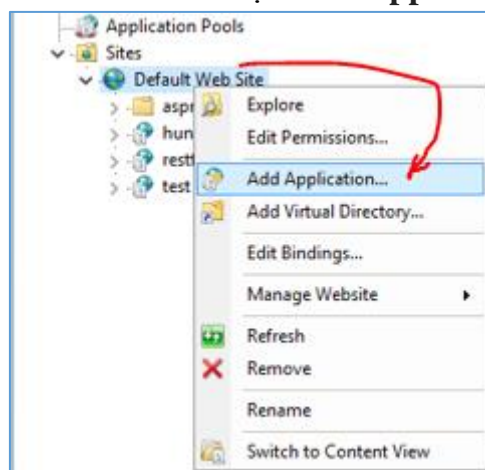
Ở màn hình trên bạn tiếp tục chọn Advanced Settings.. cho testpool:



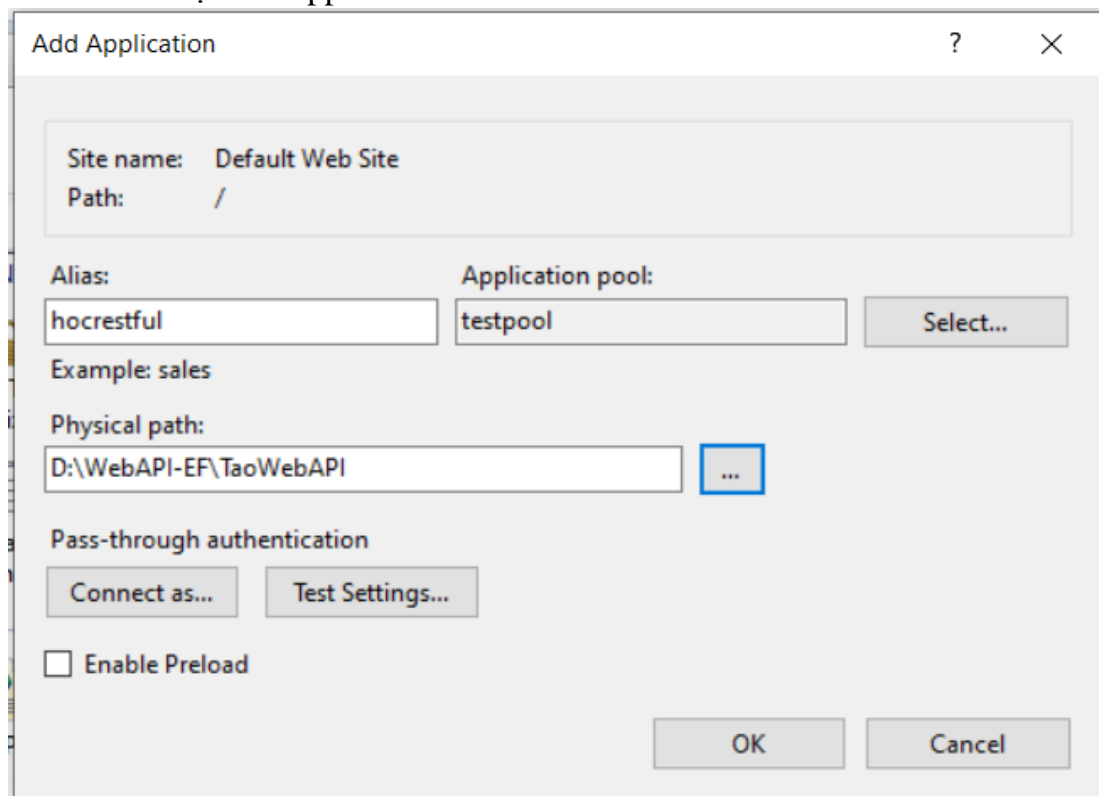
Bạn cần chỉnh ApplicationPoolIdentity qua **LocalSystem**.

+ Cấu hình WebService lên IIS Server

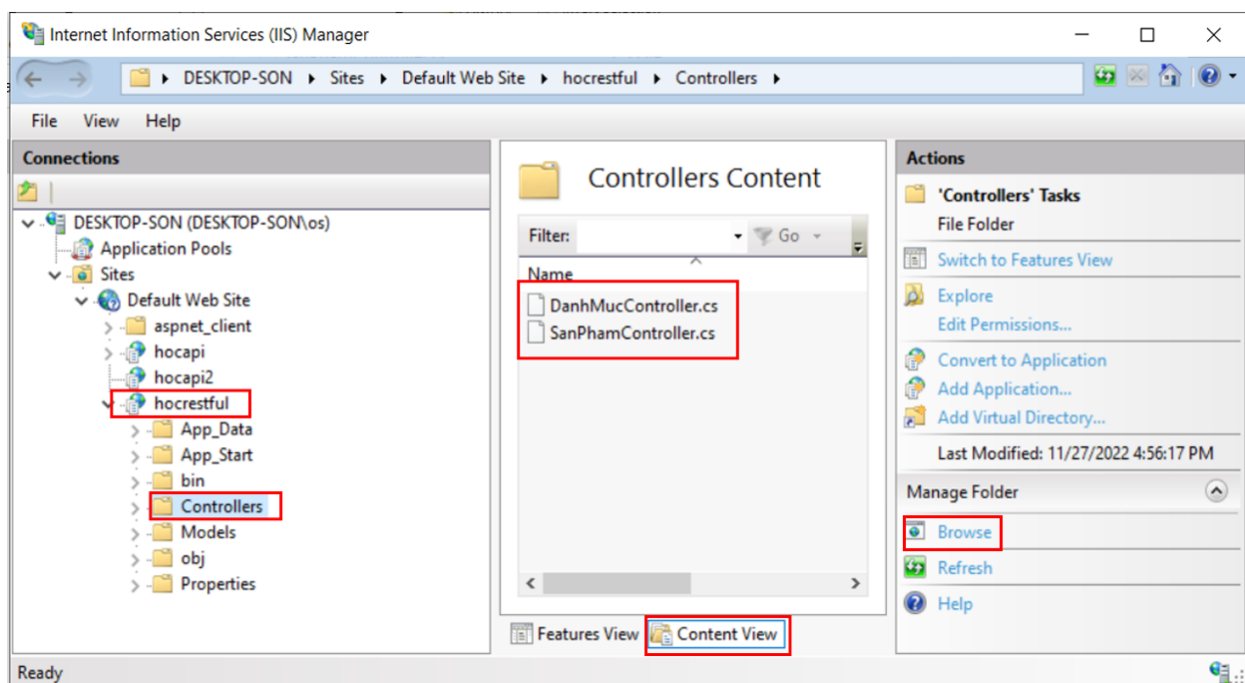
- Bấm chuột phải vào Default Web Site / chọn Add App




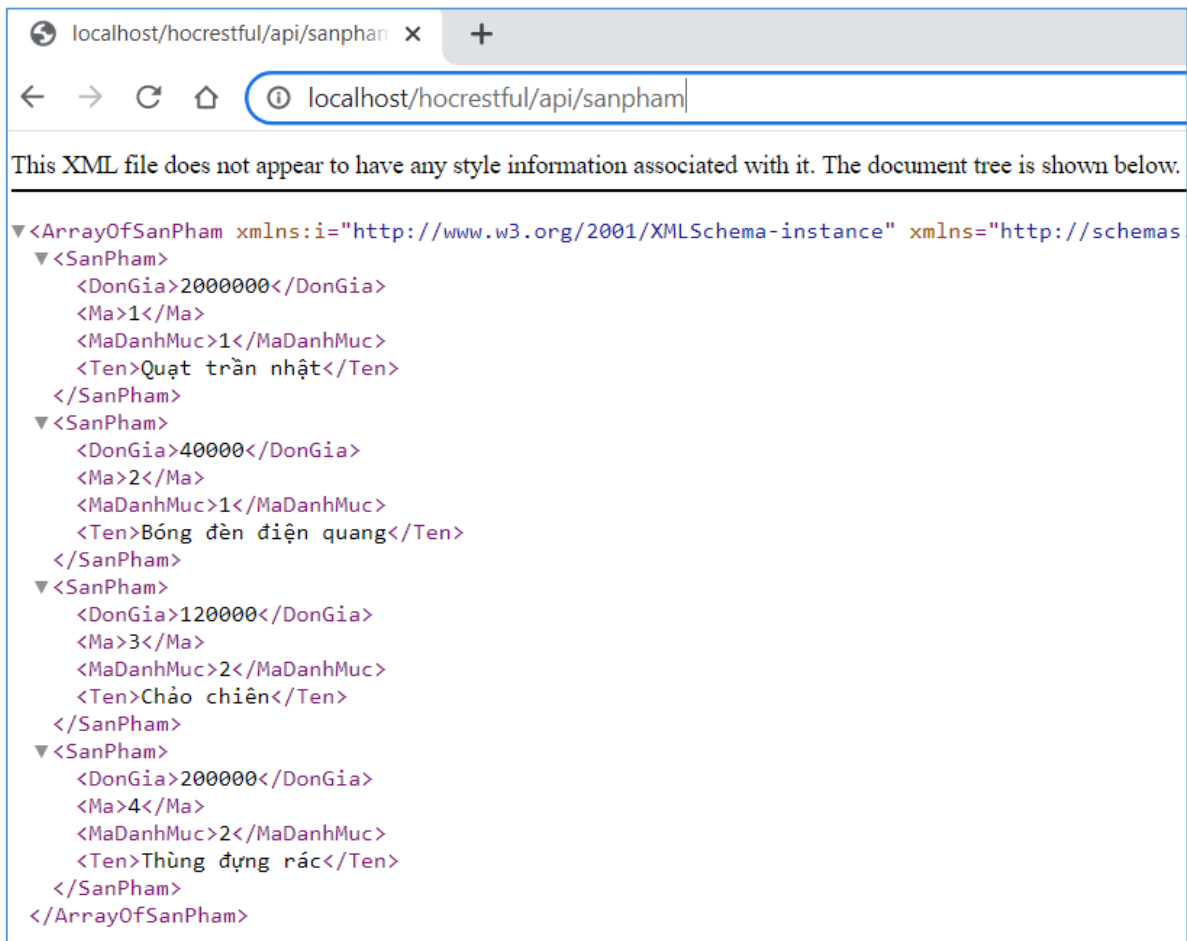
- Màn hình hiển thị Add Application:



- Mục Alias (1): Đặt tên cho Service, ở đây ta đặt **hocrestful**
 - Mục Application pool (2): Nhấn nút Select... và chọn đúng **testpool**
 - Mục Physical path (3): Trở tới đường dẫn chứa source code của project
- Sau đó bấm OK để tạo, ta quan sát kết quả:



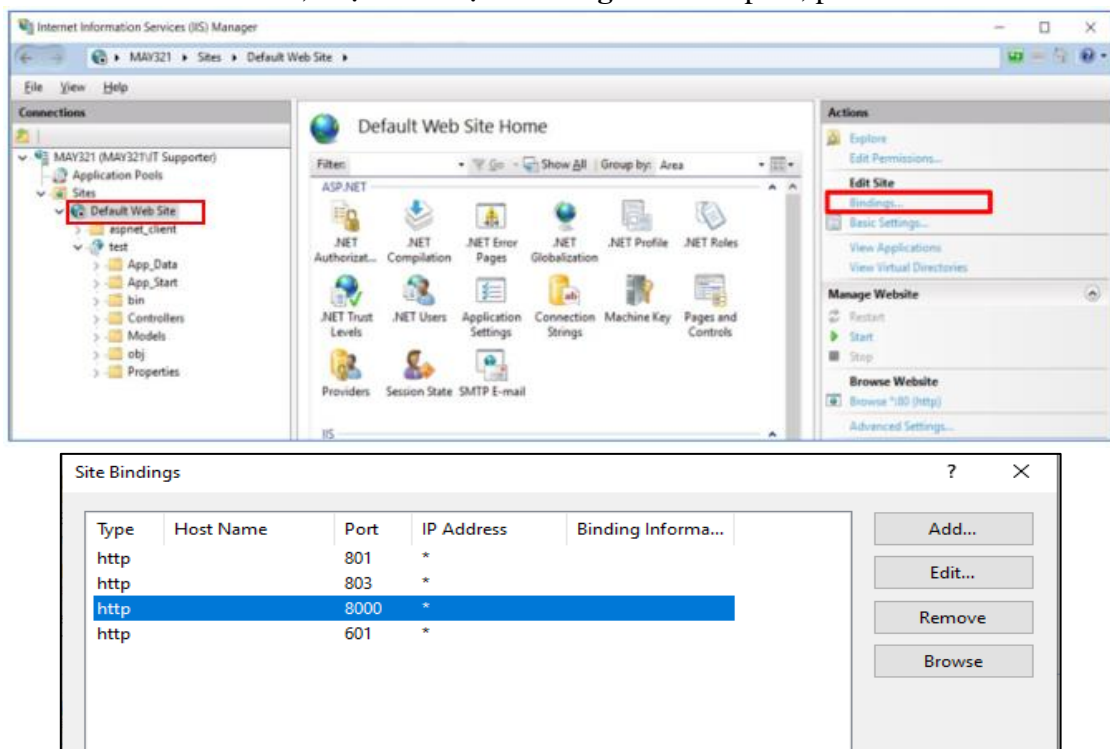
- Ta mở 1 trình duyệt bất kỳ, chạy dòng lệnh sau (hoặc chọn **hocrestful**, click vào  **Browse** <http://localhost/hocrestful/api/sanpham>
- Quan sát kết quả:



<http://localhost/hocrestful/api/sanpham> chính là kết quả sau khi cấu hình xong. Tới đây hệ thống đã chạy 24/24.

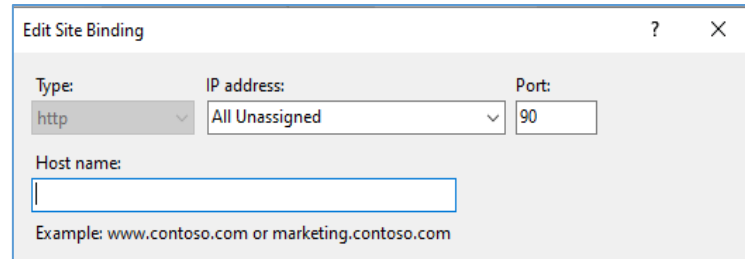
❖ Nếu có lỗi và không trả lại kết quả XML, ta cần đổi lại cổng:

Là do có một chương trình nào đó đã dùng cổng **80**. Chúng ta chỉnh lại cổng cho IIS như sau: Click vào **Default Web Site**, chọn vào mục **Bindings...** ở bên phải, phía trên:

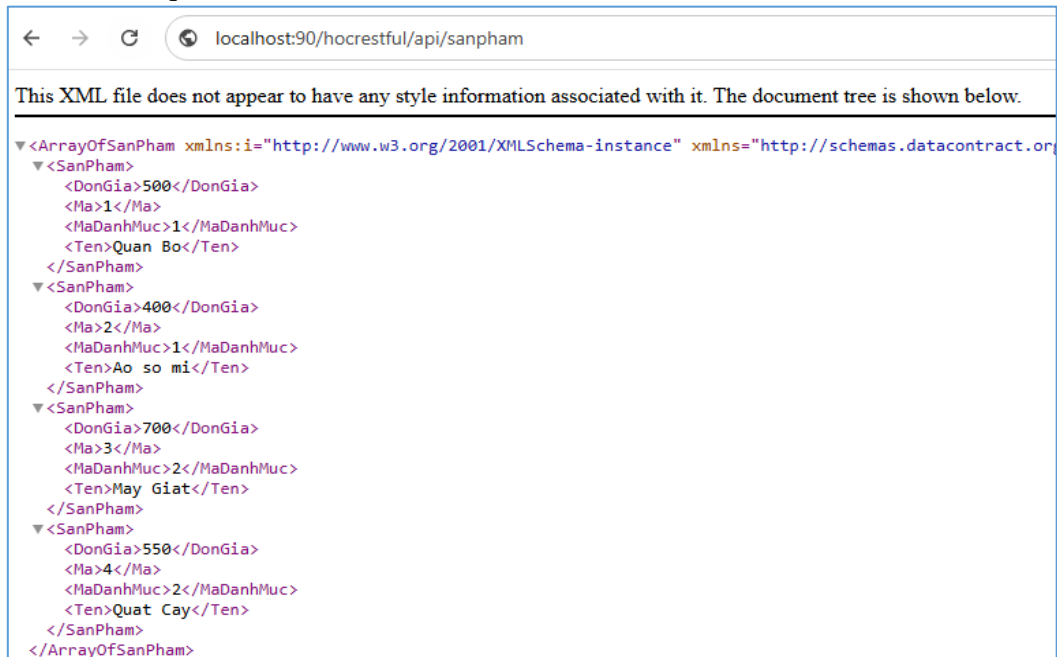


Remove tất cả các cổng và chỉ giữ lại 1 cổng **8000**

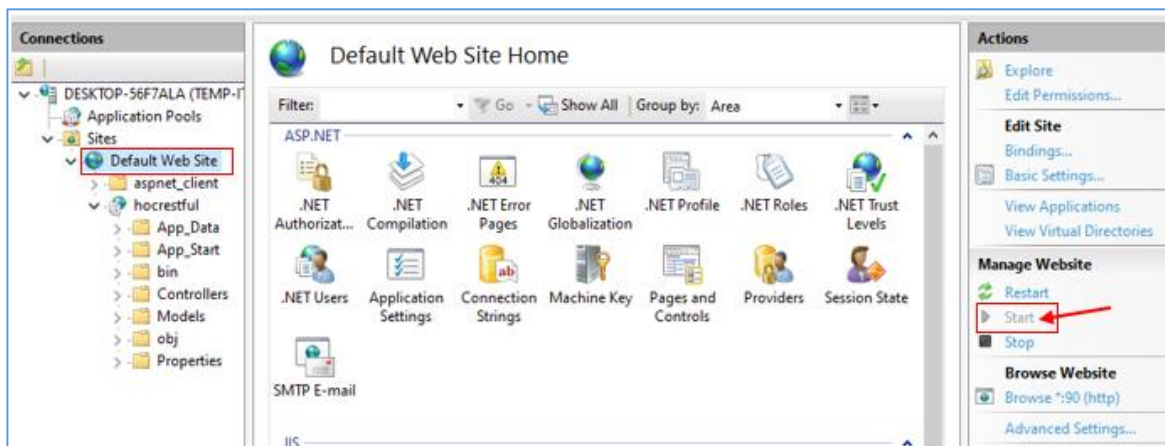
Sau đó Click vào **Edit**, sửa thành cổng **90**, OK:



Chạy lại Web API: quan sát link có **localhost:90**



❖ Nếu đổi lại cổng 90 rồi mà vẫn lỗi thì Click nút “Start” như sau:



B9. Kiểm tra Web API với công cụ Postman

+ Lấy sản phẩm theo madm

localhost/hocrestful/api/sanpham/?madm=2

GET localhost/hocrestful/api/sanpham/?madm=2

Send

Params Authorization Headers (6) Body Pre-request Script Tests Settings Cookies

Query Params

KEY	VALUE	DESCRIPTION
<input checked="" type="checkbox"/> madm	2	
Key	Value	Description

Body Cookies Headers (9) Test Results 200 OK 13 ms 389 B Save Response

Pretty Raw Preview Visualize JSON

```
1 {
2   "Ma": 3,
3   "Ten": "Chào chiến",
4   "DonGia": 120000,
5   "MaDanhMuc": 2
6 },
7 {
8   "Ma": 4,
9   "Ten": "Thùng đựng xác",
10  "DonGia": 200000,
11  "MaDanhMuc": 2
12 }
13 }
14
```

+ Thêm mới sản phẩm

localhost/hocrestful/api/sanpham/?ma=111&ten=abc&gia=400&madm=1

POST localhost/hocrestful/api/sanpham/?ma=111&ten=abc&gia=400&madm=1

Send

Params Authorization Headers (7) Body Pre-request Script Tests Settings Cookies

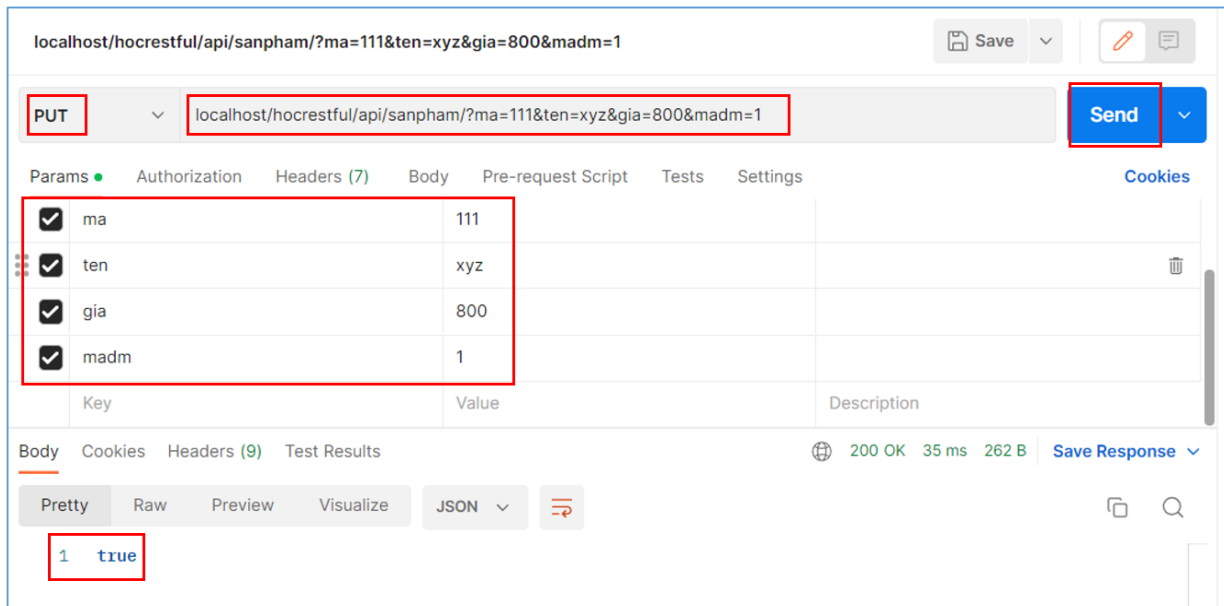
<input checked="" type="checkbox"/> ma	111	
<input checked="" type="checkbox"/> ten	abc	
<input checked="" type="checkbox"/> gia	400	
<input checked="" type="checkbox"/> madm	1	
Key	Value	Description

Body Cookies Headers (9) Test Results 200 OK 311 ms 262 B Save Response

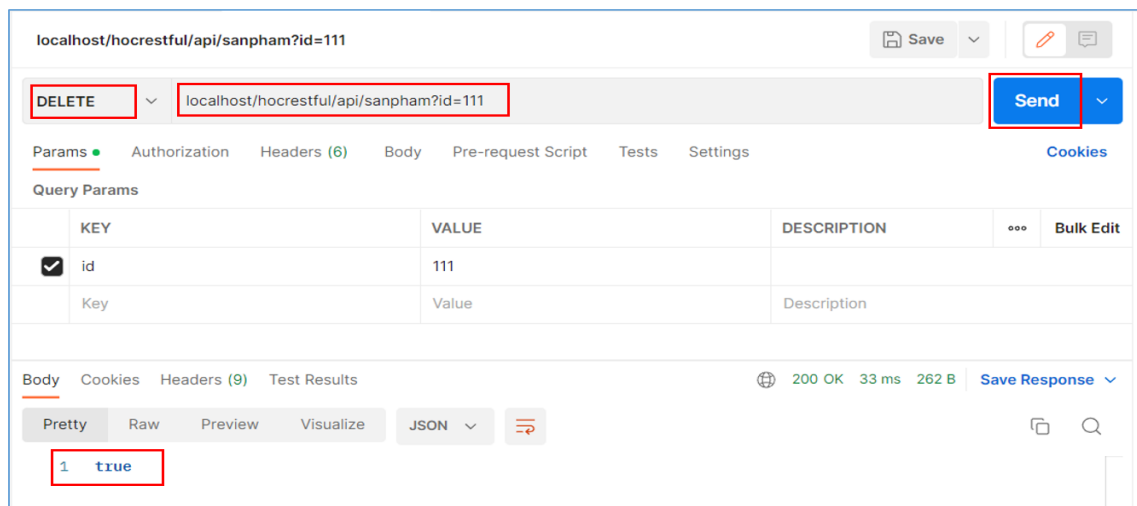
Pretty Raw Preview Visualize JSON

```
1 true
```

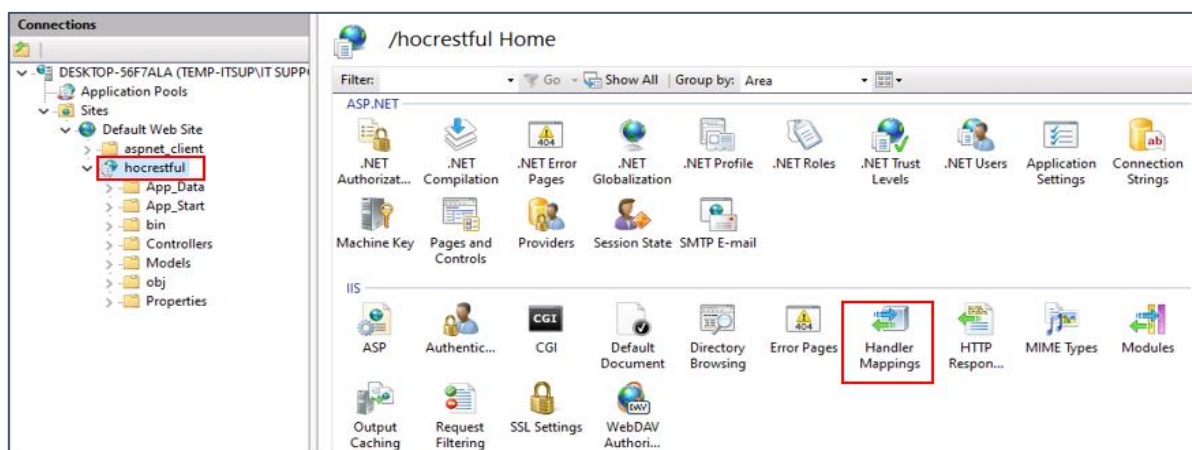
+ Sửa sản phẩm



+ Xóa sản phẩm



*** Lưu ý: Nếu dùng Postman Test các WebAPI sửa, xóa mà bị lỗi thì sửa như sau:**

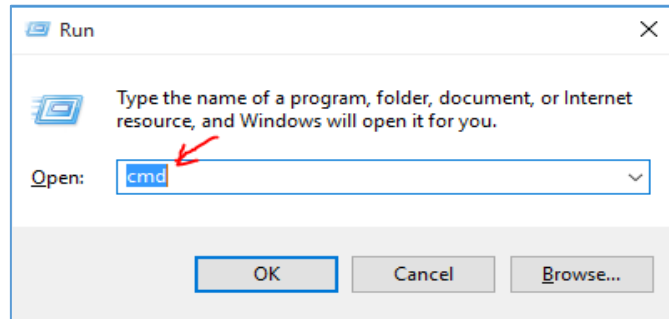


Click đúp chuột vào mục **Handler Mappings**, click chuột phải vào **WebDAV**, chọn **Edit**
Click vào nút **Request Restrictions...**, chọn **Web**, chọn **All Web**, OK

* **Các bạn chú ý rằng khi từ Mobile kết nối tới Web Service này thì không thể dùng localhost (vì nó làm tưởng là của chính mobile), do đó bạn cần đưa localhost về 1 địa chỉ IP để test.**

Cách lấy địa chỉ IP của máy local của ta:

– Nhấn tổ hợp phím có biểu tượng Windows + R để mở cửa sổ Run:



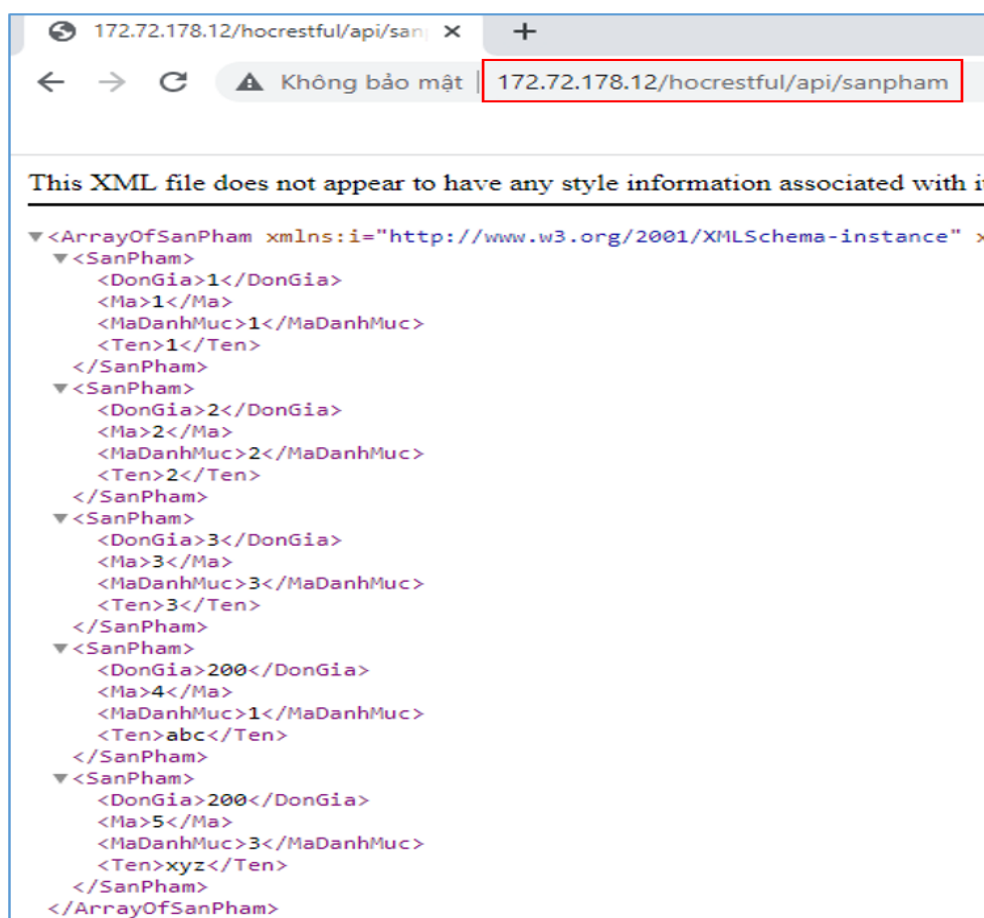
– gõ lệnh **cmd**, màn hình command line hiển thị lên, tiếp tục gõ lệnh **ipconfig**

```
Wireless LAN adapter Wi-Fi:

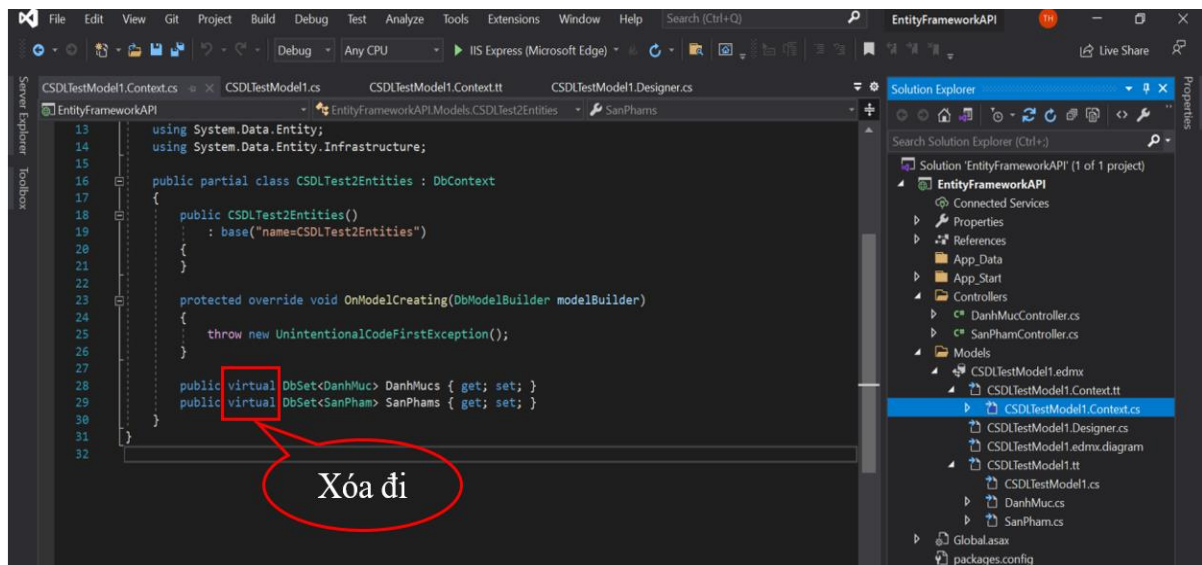
    Connection-specific DNS Suffix  . : 
    Link-local IPv6 Address . . . . . : fe80::4013:5ed:d990:cd64%9
    IPv4 Address. . . . . : 172.72.178.12
    Subnet Mask . . . . . : 255.255.0.0
    Default Gateway . . . . . : 172.72.0.1
```

Ở trên máy local có địa chỉ IP: **172.72.178.12**

Bây giờ thay thế cho localhost:



Lưu ý: Nếu tạo các bảng bằng câu lệnh, thì ta sẽ sửa đổi ở model sinh ra như sau:



Trong CSDL TestModel.context.cs => Xóa **Virtual** của DanhMucs, và SanPhams là OK