

## Contents

### Capitolul 4 - Servicii Web

1. Introducere .....	2
2. Crearea unei aplicatii Web Service.....	3
3. Crearea unui client de tip Console Application .....	8
4. Hostarea serviciului in IIS .....	10
Exercitii.....	15
Tema.....	16

## Capitolul 4

### Servicii Web

#### 1. Introducere

Un serviciu web este o colecție de protocoale și standarde utilizate pentru comunicarea între două dispozitive prin *Internet*. Acesta se bazează pe modelul client-server, astfel: serverul expune unele metode și clientul se conectează la server pentru a utiliza metodele. Ideea este de a crea componente, precum black box, care pot interacționa, fără a lua în considerare sistemul de operare sau limbajul de programare. Mai precis, serviciul web este o componentă sau un modul de cod executabil cu o interfață specială care face ca metodele să fie disponibile pentru alte programe utilizând o solicitare http. Cererea se face prin utilizarea http get, http post și simple Object Access Protocol (SOAP).

În figura de mai jos este prezentat modul de funcționare al aplicațiilor client - server.

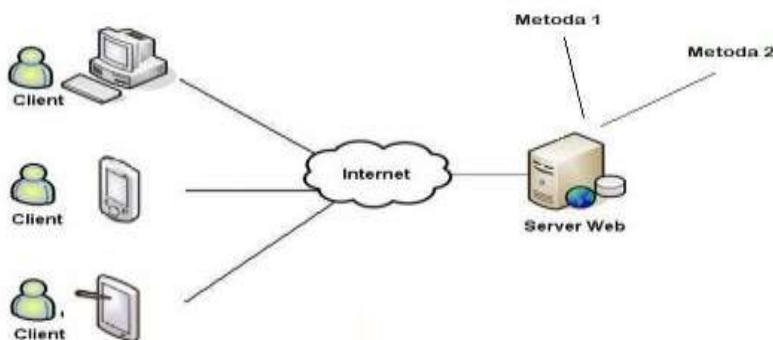


Fig. 1 Modul de funcționare al aplicațiilor client -server

Prin urmare, metodele sau mesajele pot fi publicate pentru a fi utilizate de pretutindeni. Pentru codarea datelor, serviciul web folosește Extensible Markup Language (XML) și pentru transportul acestuia, un protocol bazat pe XML numit SOAP. SOAP este folosit în aplicații care necesită schimb de informații prin HTTP. Un serviciu web poate fi localizat utilizând limbajul de definire a serviciului Web (WSDL). Arhitectura serviciilor web este prezentată în figura de mai jos.

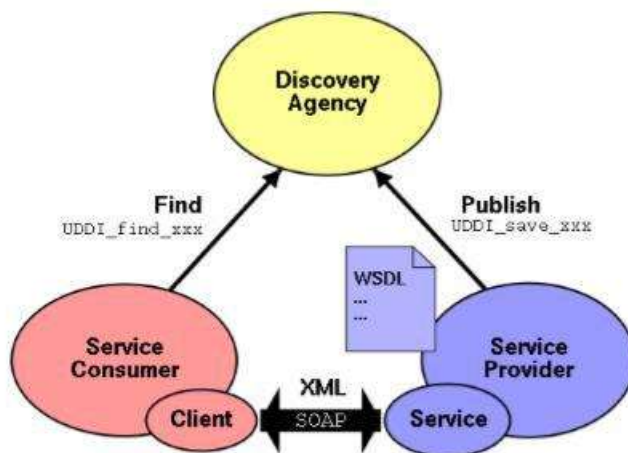


Fig. 2 Arhitectura serviciilor web

Serviciile Web aduc următoarele *avantaje* în cadrul unei aplicații distribuite:

- **Simplitate:** serviciile Web pot fi suportate de toate platformele întrucât folosesc protocoale cunoscute („open standards”) global;
- **Scalabilitate ridicată:** întrucât, un serviciu web poate răspunde la un număr mare de clienți;
- Clientul nu este afectat dacă definițiile funcțiilor din interfața expusă de serviciul Web sunt modificate, totuși metodele trebuie să păstreze aceiași parametri și același tip de returnat;
- **Interoperabilitatea:** un serviciu WEB poate fi folosit de către clienți implementați în limbaje de programare diferite.

În continuare sunt prezentate câteva *exemple* de utilizare a serviciilor WEB.

- Majoritatea serviciilor Web reale sunt de tip SOAP, dar există și câteva servicii de tip **REST notabile (RESTful Web Services)**: Amazon S3 (Simple Storage Service) pentru memorarea și regăsirea de obiecte, rețeaua Twitter și alte site-uri de blog, în care se descarcă fișiere XML în format RSS sau Atom cu liste de legături către alte resurse.
- Ca exemple de **servicii SOAP (Simple Object Access Protocol)** larg utilizate sunt servicii pentru rate de schimb între diferite valute, servicii bancare pentru verificare și operare în conturi (ex: de către aplicații Web de comerț electronic), servicii de memorare și regăsire de date, ș.a.

## 2. Crearea unei aplicații Web Service

În primul rând, pentru a putea crea aplicații de tip Web, trebuie să avem adăugat pachetul de servicii necesare din Visual Studio Installer, și anume: ASP.NET and web development.

**Construcția unui serviciu Web** Incepe de la o clasa ce contine una sau mai multe metode web si cel puțin una dintre aceste metode va fi publicata si expusa prin interfata Web a serviciului.

Pentru a crea un serviciu Web, utilizând mediul de dezvoltare Visual Studio, putem opta între cele două posibilități: fie selectăm din Start Page **Create new project...**, fie din bara de meniu, după cum urmează: File → New → Project → Visual C# → Web → **ASP.NET Web Application**, precum în figura 3 și 4. După crearea proiectului, toate fișierele din cadrul proiectului pot fi găsite în Solution Explorer.



Fig.3 Crearea unui proiect de tip Web Service

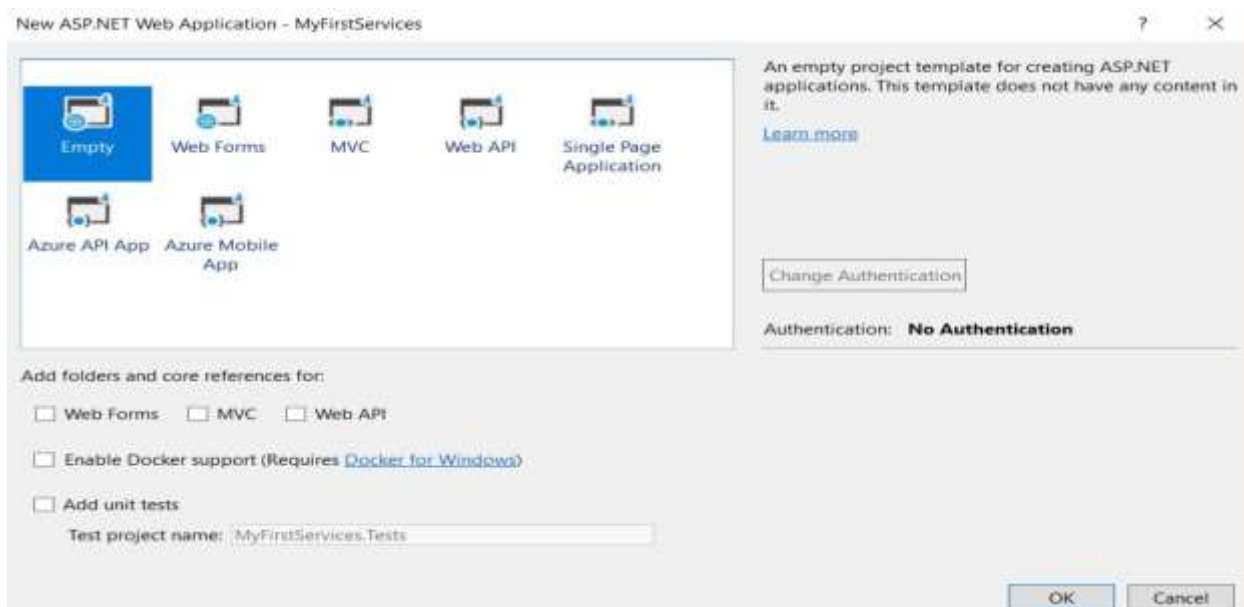


Fig.4 Crearea unui proiect de tip Web Service

Dupa crearea aplicatiei de tip Web trebuie sa ne adaugam serviciul pentru a putea crea metode necesare. Acest lucru se realizeaza precum In figurile ce urmeaza ( fig. 5, fig.6).

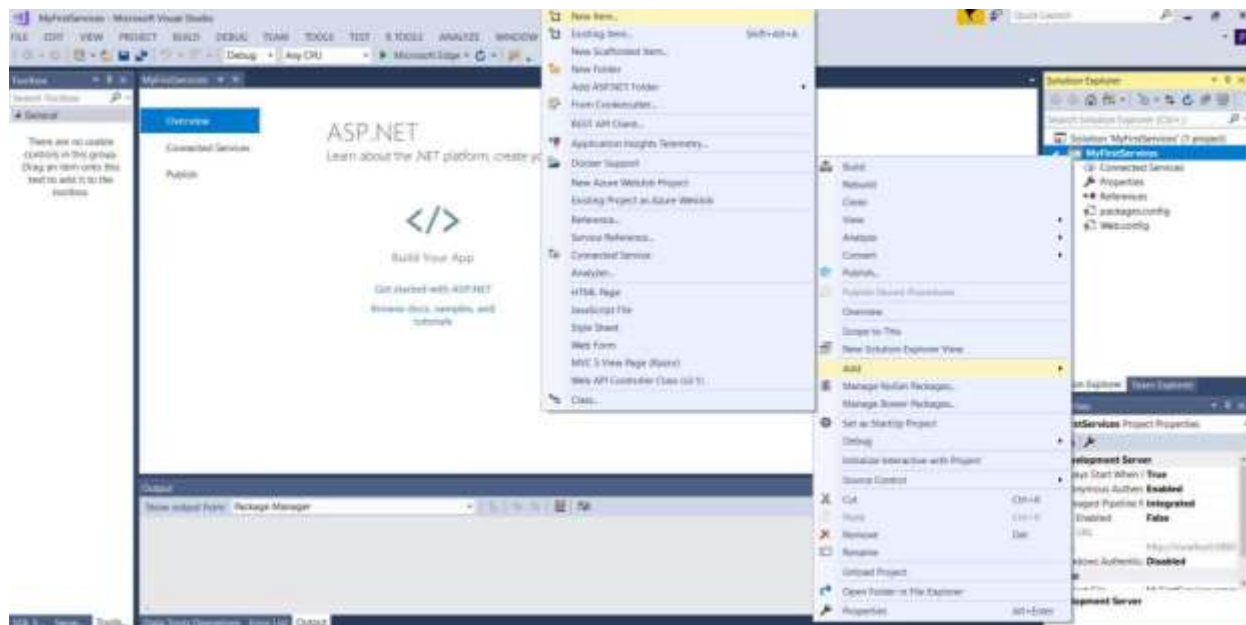



Fig.5 Adaugarea serviciului WEB



Fig.6 Adaugarea serviciului WEB

Dupa adaugarea serviciului, In clasa WebService1 este adaugata In mod automat prima metoda si anume, metoda **HelloWorld()**, dupa cum putem observa din figura 7. Urmatorul pas Il constituie expunerea serviciului In interfata Web apăsând butonul Run si selectând browser-ul dorit, In cazul nostru acesta este *Microsoft Edge* (  Microsoft Edge ).

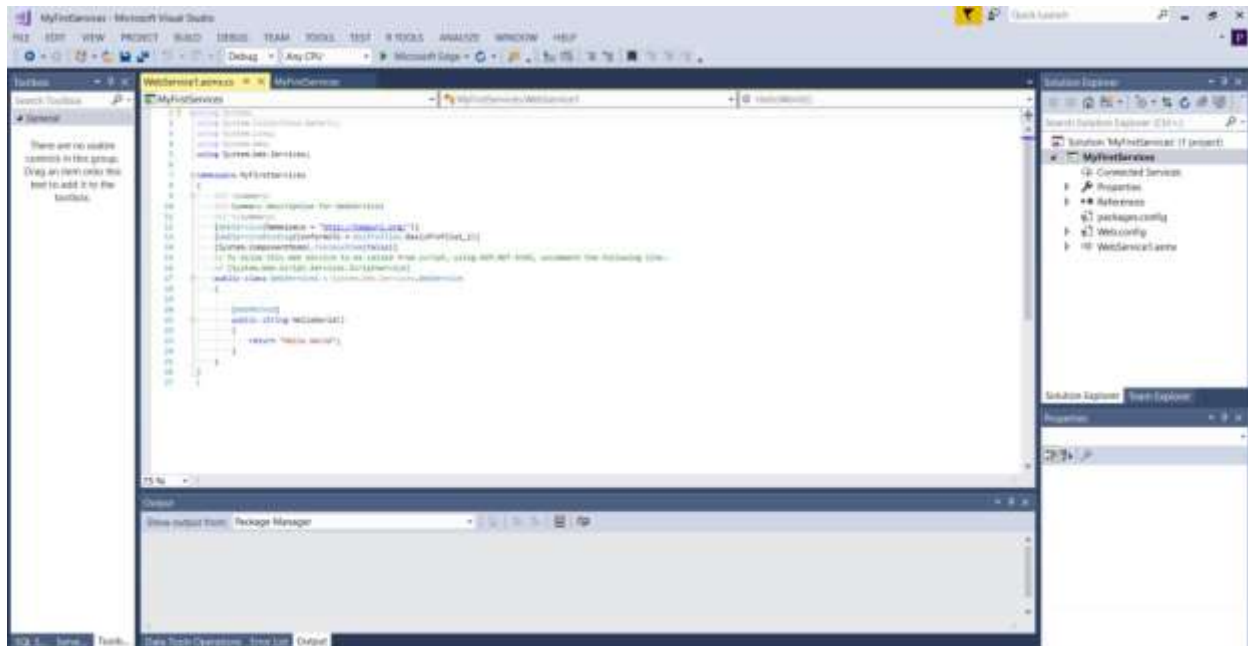


Fig.7 Codul sursa generat automat

**Nota!:** Testarea serviciului Web se face folosind un browser Web In care introducem adresa catre fisierul .asmx. In browser v-a aparea o pagina de test generata, de runtime-ul ASP.NET. Inaintea testarii aplicatiei In browser nu exista nici o alta modalitate de a executa codul metodei serviciului Web.

In urma rularii serviciului web obtinem:

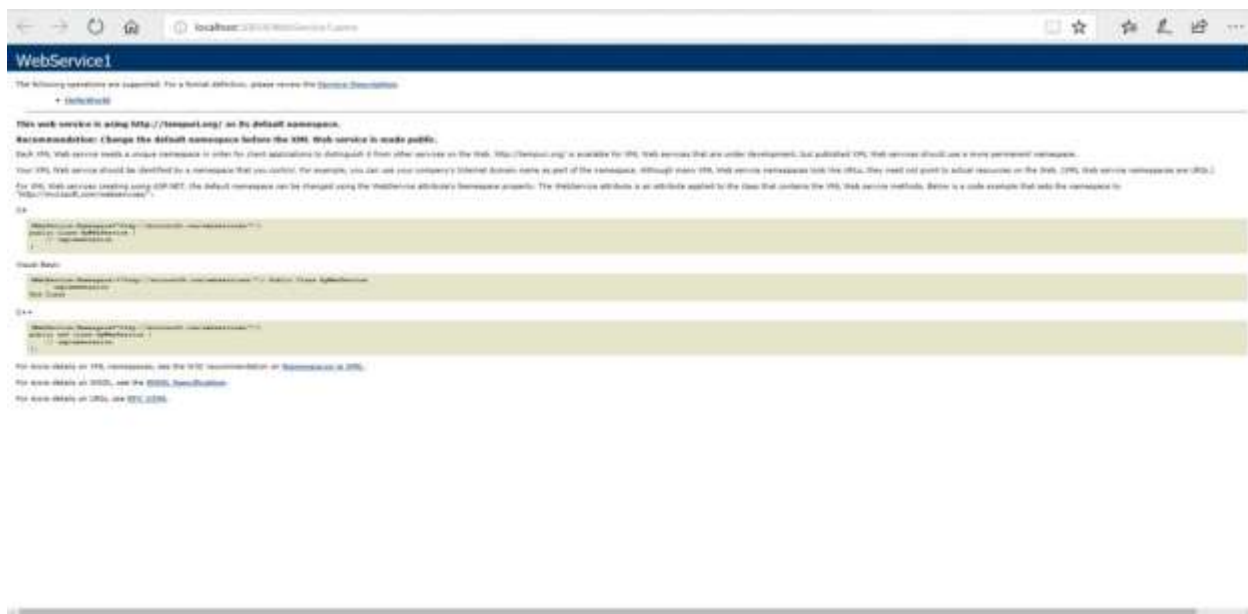


Fig. 8 Rularea serviciului Web

**Observatie!:** Pentru ca o metoda sa poata fi apelata (vizibila) In aplicatia client, aceasta trebuie declarata ca metoda web prin intermediul entitatii **[WebMethod]**. Pentru fiecare metoda In parte trebuie adaugata aceasta directiva.

### *Adaugarea unei descrieri pentru serviciul Web si pentru metodele acestuia*

ASP.NET ofera posibilitatea descrierii serviciului Web si a fiecărei metode In parte. Acest lucru se poate dovedi util pentru clientii noi ai unui serviciu Web. Documentarea se face prin decorarea cu attribute atât a clasei cât si a metodelor acesteia, dupa cum putem observa In cele ce urmeaza.

```
[WebService(Description="Serviciu Web pentru Cursul de C#",
    Name="My First Web Service" , Namespace = "Web Services")]
// [WebService(Namespace = "http://tempuri.org/")]

[WebMethod(Description = "This method displays a message")]
public string HelloWorld()
{
    return "Hello World";
}
```

Fig. 9 Adaugarea descrierii serviciului Web

Sa adaugam o alta metoda. De exemplu, vrem sa calculam suma a doua numere. Pentru aceasta, implementam o metoda dupa cum urmeaza:

```
[WebMethod]
public int Add(int a, int b)
{
    return a + b;
}
```

Prin construirea si derularea proiectului, vom avea primul nostru serviciu web, asa cum este prezentat In figura 10.

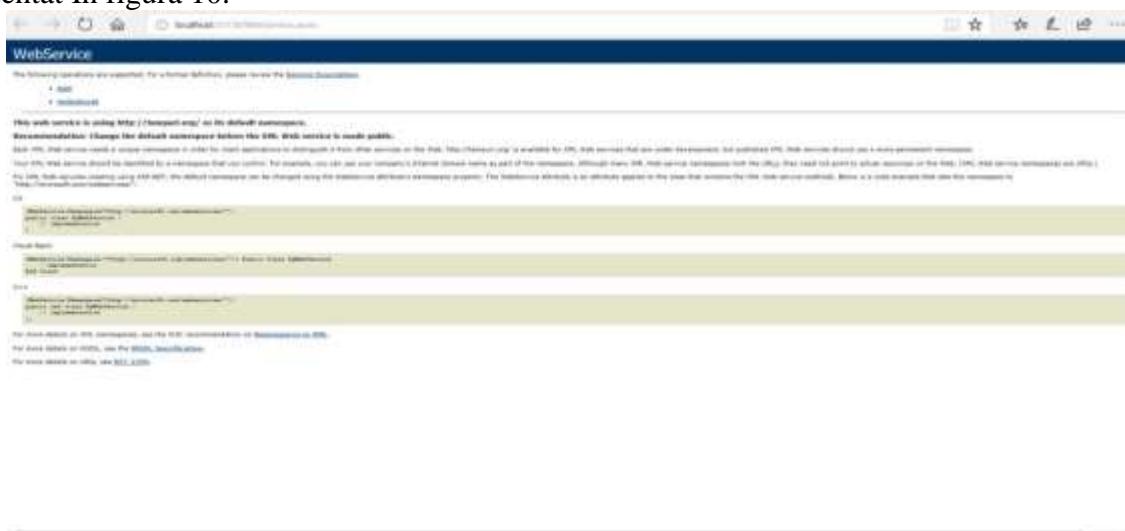


Fig. 10 Primul serviciu web



Dupa dezvoltarea serviciului cu metodele necesare, pentru a le putea utiliza este nevoie sa construim si aplicatiile client. In continuare este prezentam modul prin care ne putem dezvolta un client de tip Console Application In mediul de dezvoltare Visual Studio.

### 3. Crearea unui client de tip Console Application

Pentru a crea un client, deschidem o noua fereastră Visual Studio si cream un nou proiect de tip Console Application, precum In figura 11 si denumim proiectul myFirstClient.

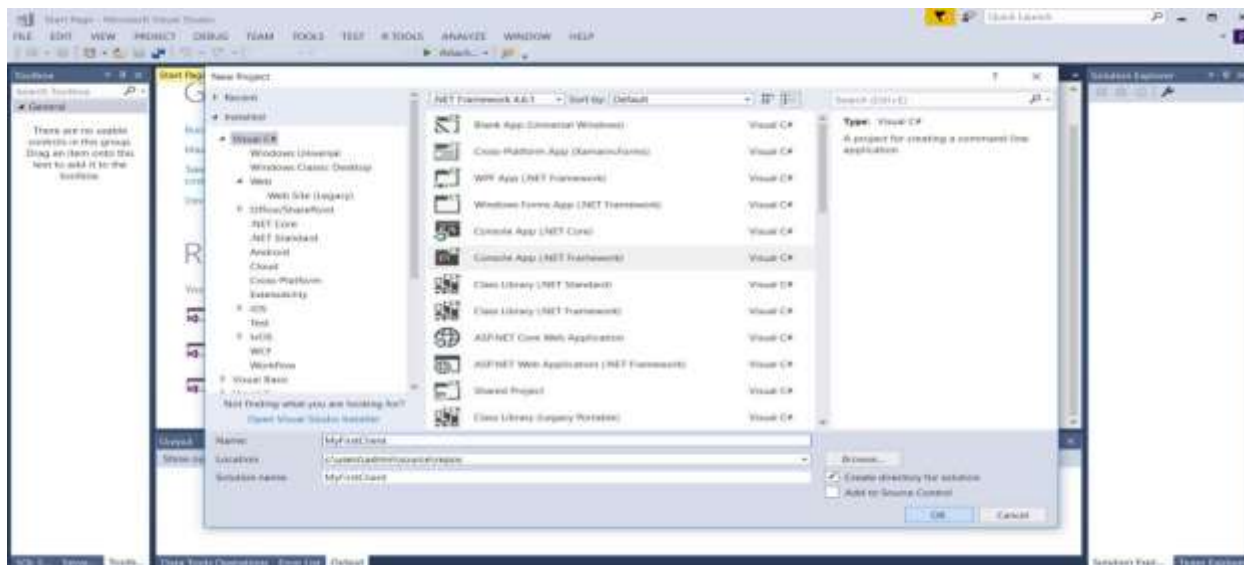


Fig. 11 Crearea clientului de tip Console Application

Dupa crearea proiectului, asa cum spuneam, trebuie sa **adaugam referinta serviciului**, mai exact sa conectam aplicatia Web la client prin intermediul adresei URL a serviciului. Acest lucru In putem observa In figura 12. Dupa ce am obtinut o referinta la un serviciu Web al doilea pas, pentru un client este de a cunoaste modul de a interactiona cu serviciul adica: cunoasterea metodelor disponibile, a parametrilor si tipul acestora.

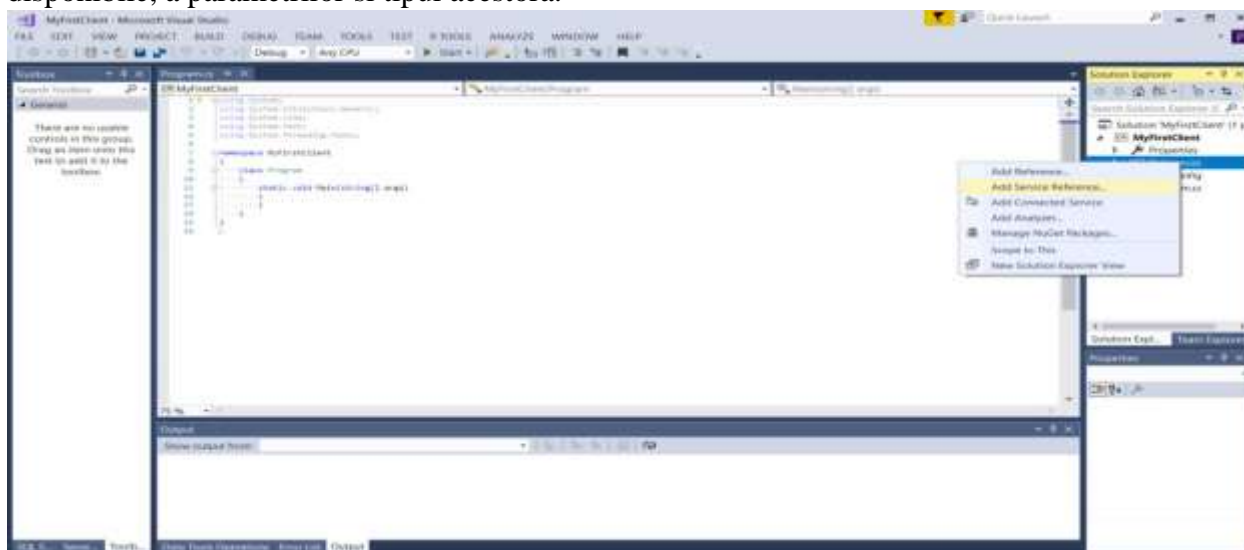


Fig. 12 Adaugarea unei referinte la serviciul WEB



**Nota!:** Pentru a putea adauga serviciul In aplicatia client, acesta trebuie sa fie pornit (sa ruleze) si mai trebuie sa copiem adresa URL a acestuia din browser pentru a o introduce la pasul urmator (figura 13), dupa care se apasa butonul Go. Daca serviciul este gasit, acesta v-a aparea sub adresa URL, si vom putea vizualiza inclusiv metodele serviciului, apăsând dublu click pe numele serviciului.

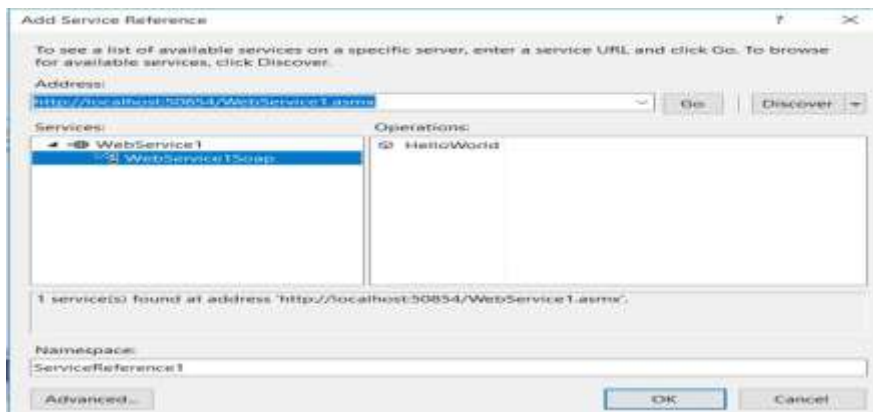


Fig. 13 Adaugarea unei referinte la serviciu WEB

**Observatie!:** Atunci când adaugam metode noi serviciului, pentru ca acestea sa fie accesibile (vizibile) In aplicatiile client trebuie sa facem update la referinta serviciului. Acest lucru se poate vedea In figura ce urmeaza.

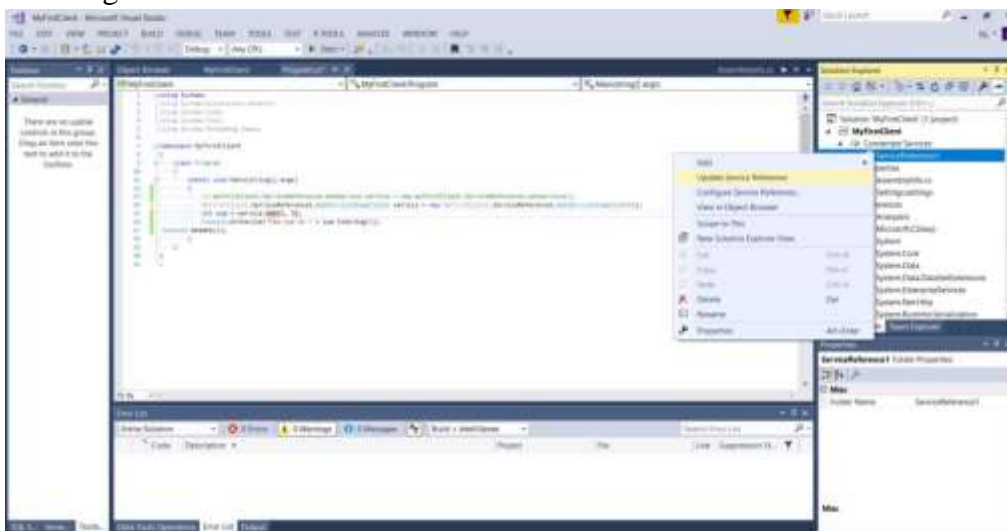


Fig. 14 Updatarea serviciului web In aplicatia client

Acum, clientul este conectat la serviciu si poate utiliza metodele expuse de serviciu. In aplicatia client trebuie sa creem un obiect, cu ajutorul caruia apelam metodele aplicatiei Web.

```
static void Main(string[] args)
{
    MyFirstClient.ServiceReference1.WebService1SoapClient service = new
    MyFirstClient.ServiceReference1.WebService1SoapClient();

    int sum = service.Add(2, 3);
    Console.WriteLine("The sum is " +
    sum.ToString()); Console.ReadKey();
}
```

Dupa implementarea codului, putem sa testam aplicatia, apăsând butonul Run. Rezultatul rularii programului se poate observa In figura de mai jos.



Fig. 15 Rezultatul rularii aplicatiei client

#### 4. Hostarea serviciului Web in IIS

IIS este un server WEB care ruleaza in platforma Microsoft .NET pe sistemele de operare Windows. Ce este un Web Server? Internetul si paginile web nu pot exista fara servere web. Un server web este un process pentru gazduirea aplicatiilor web. Un astfel de server permite aplicatiilor sa proceseze mesaje primite prin canalele specific TCP. De exemplu portul pentru HTTP este 80 iar pentru HTTPS este 443. Cand un utilizator acceseaza un website din browser, de obicei nu specifica si portul, decat in cazurile in care serverul web este configurat astfel incat sa suporte traffic pe alte porturi decat cele implicite. Vizitand <http://www.exemplu.ro> o sa trimita o cerere catre portul default 80. De altfel si adresa <http://www.exemplu.ro:80> este corecta precum si adresa <https://www.exemplu.ro:443> in cazul in care avem conexiune securizata.

Astfel, presupunand ca avem configuratiile de baza, serverul web o sa primeasca tot traficul pe canalele 80/443. Ce face mai departe aplicatia noastra cu aceste date depinde de ce avem scris in aplicatie. Gazduirea in IIS este necesare deoarece ne

Pentru a "hosta"/gazdui serviciul creat in IIS trebuie executati urmatoorii pasi:

1. Publicarea aplicatiei: Efectuati click dreapta pe proiectul creat si selectati optiunea Publish

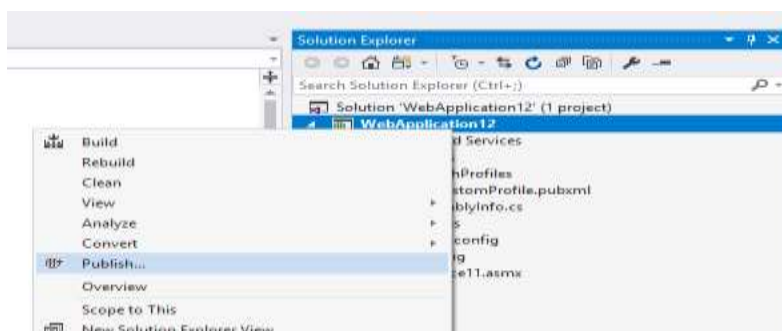


Fig. 16 Publicarea rezultatelor

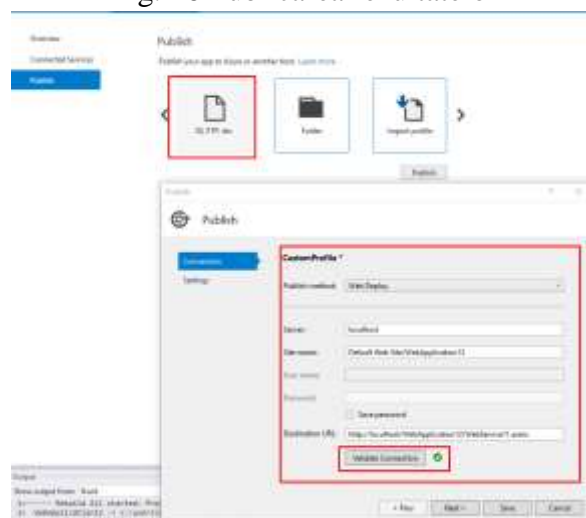


Fig. 17 Configurarea profilului in vederea publicarii

2. Selectati mai apoi optiunea IIS,FTP,etc si completati datele legate de profil precum in Fig17.
3. Click NEXT si bifati optiunea prezentata in imaginea de mai jos dupa care apasati SAVE

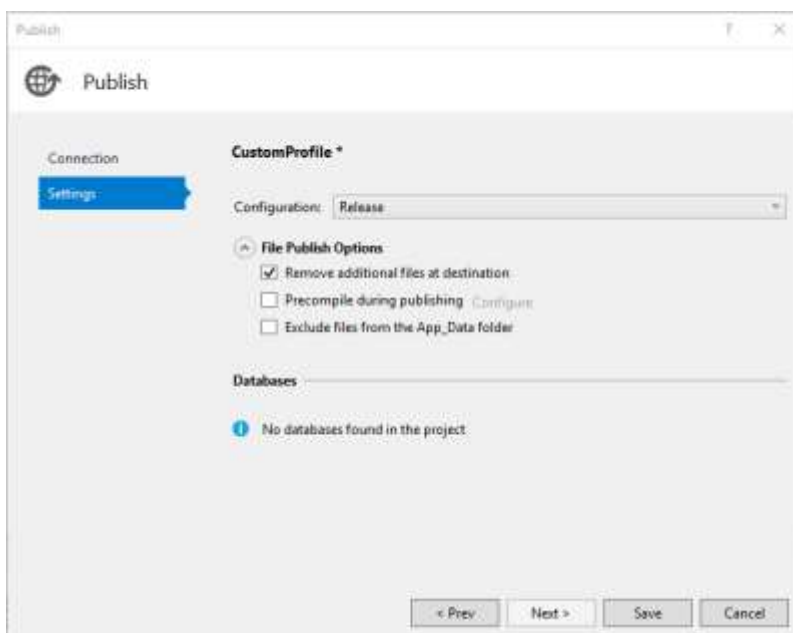


Fig. 18 Configurarea opțiunilor de publicare

Rezultatul acestui lucru se poate observa în imaginea de mai jos. Serviciul este hostat sub adresa localhost/WebApplication12/WebService11.asmx. Va rugăm să aveți în vedere faptul că adresa poate să difere pentru fiecare utilizator, depinzând de parametrii completați în pași 2, 3. Se poate observa în imaginea de mai jos că

serviciul ne pune la dispoziție o operație, și anume “HelloWorld”. Dacă se va efectua click pe aceasta putem să o invocăm și să vedem rezultatul rulării acestei metode.

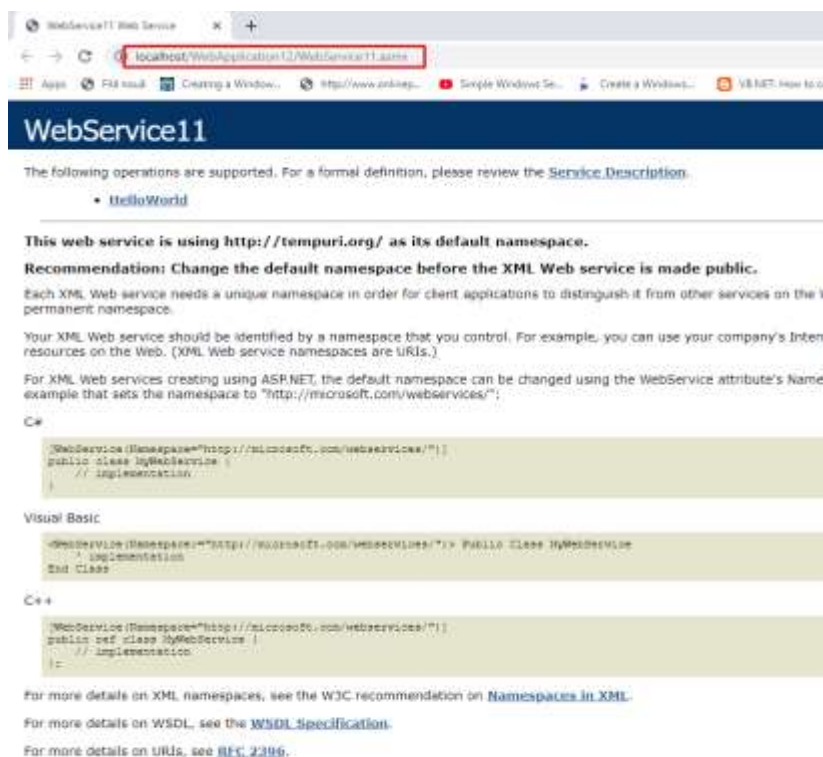


Fig. 19 Rezultatul publicării serviciului

## WebService1

Click [here](#) for a complete list of operations.

### HelloWorld

#### Test

To test the operation using the HTTP POST protocol, click the 'Invoke' button.

Invoke

#### SOAP 1.1

Fig. 20 Opțiunea de invocare a metodei HelloWorld

This XML file does not appear to have any style information associated with it. The document tree is shown below.

```
<string xmlns="http://tempuri.org/">Hello World 2</string>
```

Fig. 21 Rezultatul invocării metodei HelloWorld

Pentru a putea accesa cu succes serviciul este nevoie ca serviciul World Wide Web Publishing Service să fie pornit. În meniul Start tastati Services și accesați opțiunea care o să vă deschidă o fereastră ca și cea prezentată în imaginea de mai jos. În lista de servicii disponibile cautați serviciul menționat

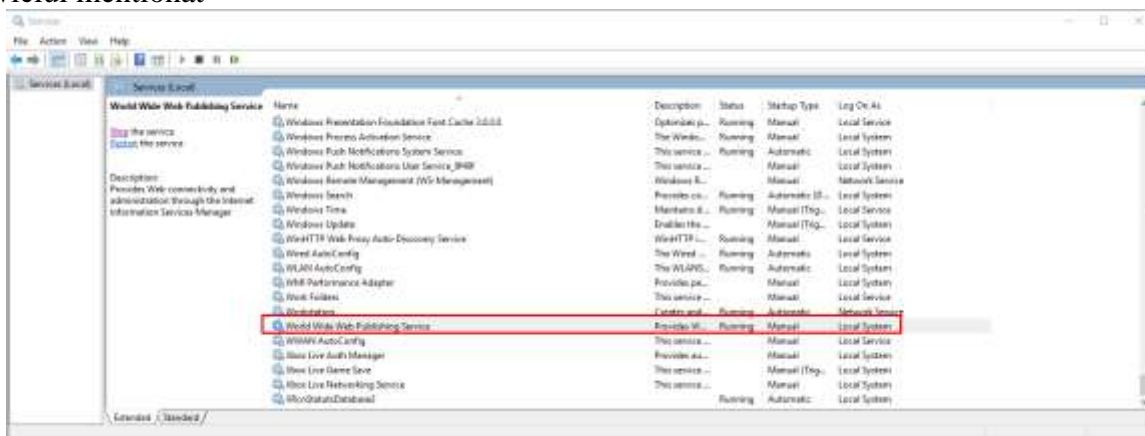


Fig. 22 Accesarea serviciului WWWPS

Click dreapta pe numele acestuia și selectați Start. Aceasta opțiune va porni acest serviciu.

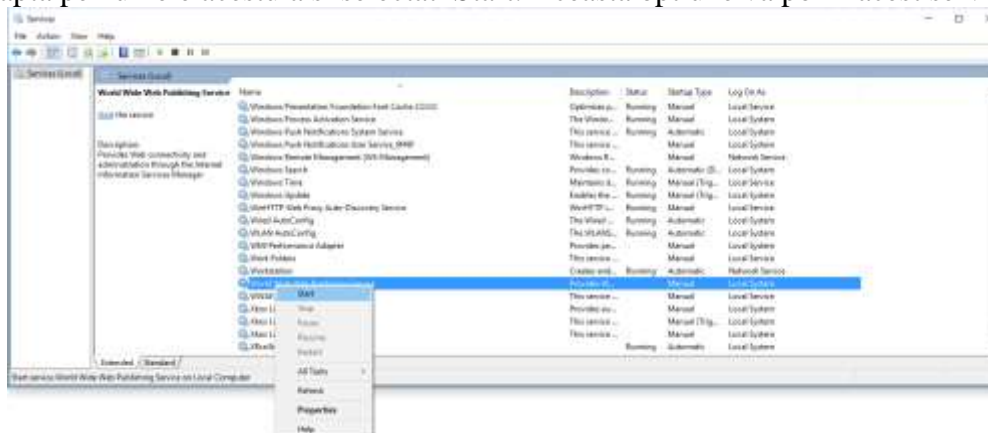


Fig. 23 Pornirea serviciului

Daca optiunea start nu este disponibila atunci o sa fie necesar sa accesati optiunea Properties, iar din fereastra General sa selectati Startup type ca fiind Manual. Dupa efectuarea acestui pas o sa puteti porni serviciul.

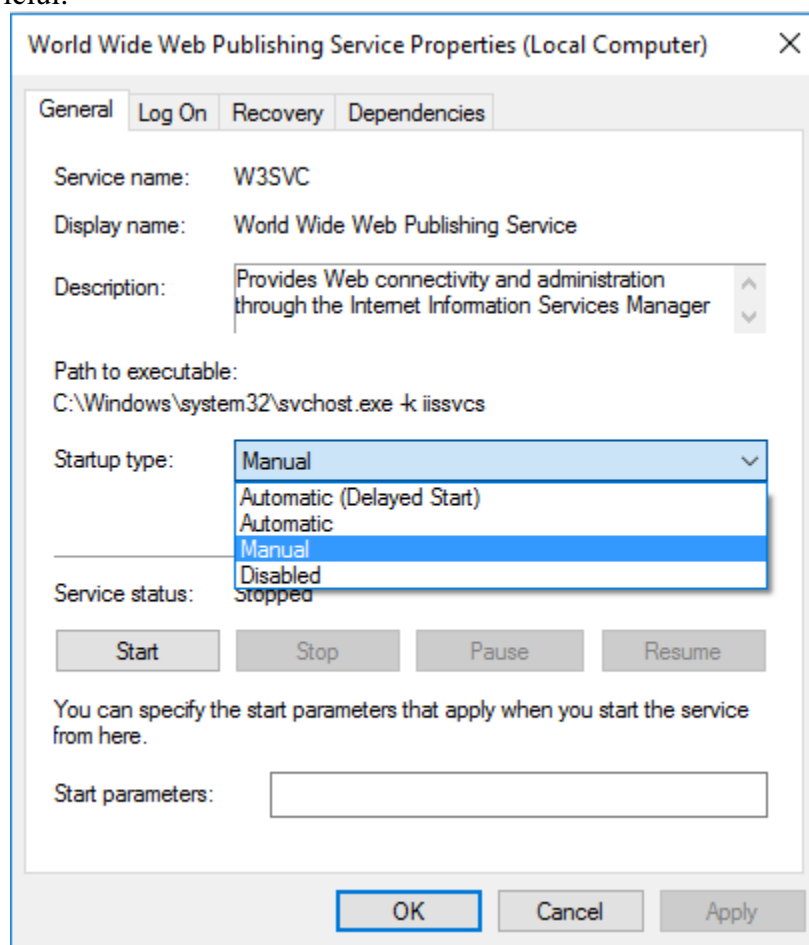
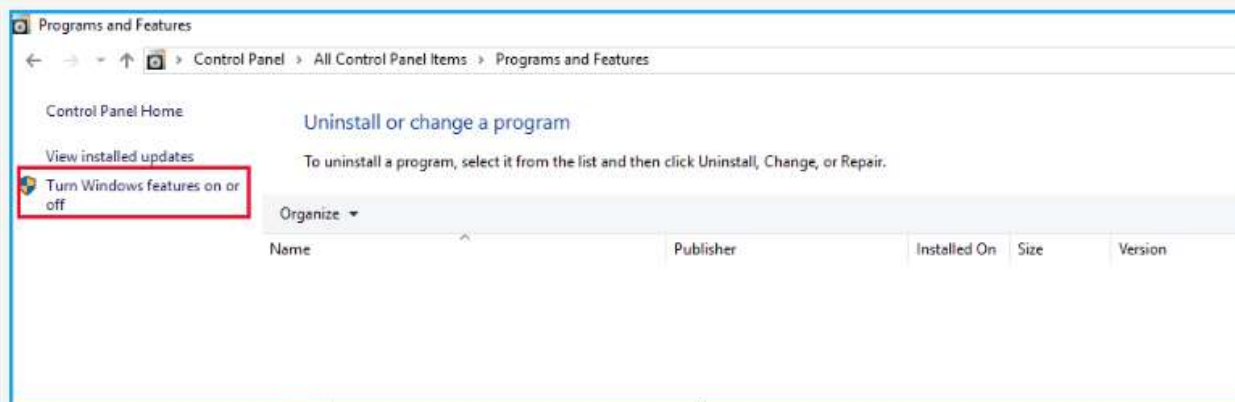
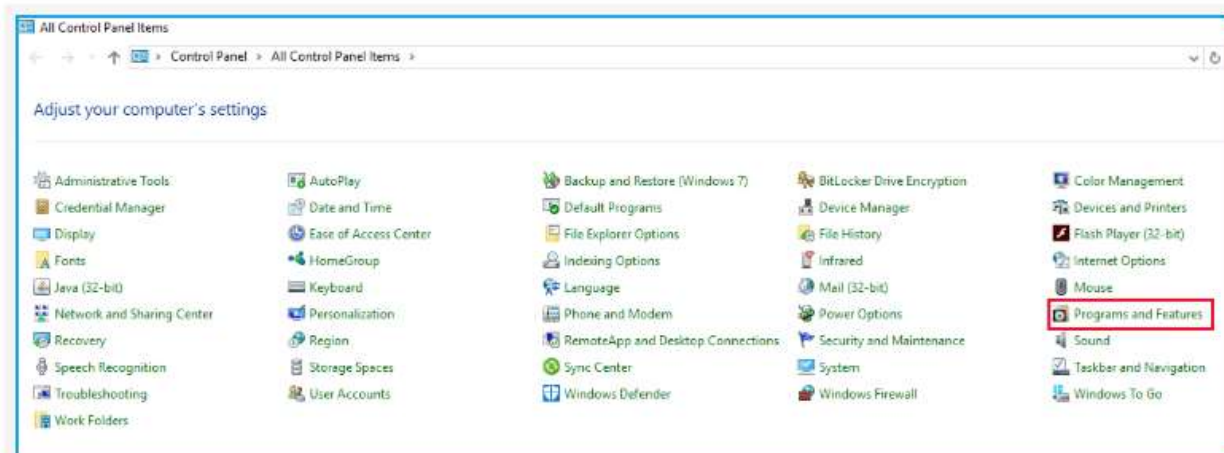


Fig. 24 Setarea modului de pornire al serviciului

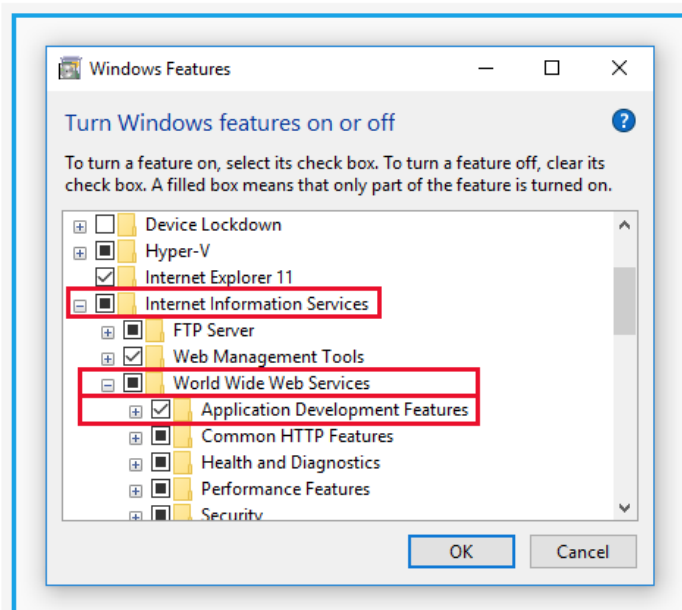
## Rezolvare erorii 404.3 – Not Found

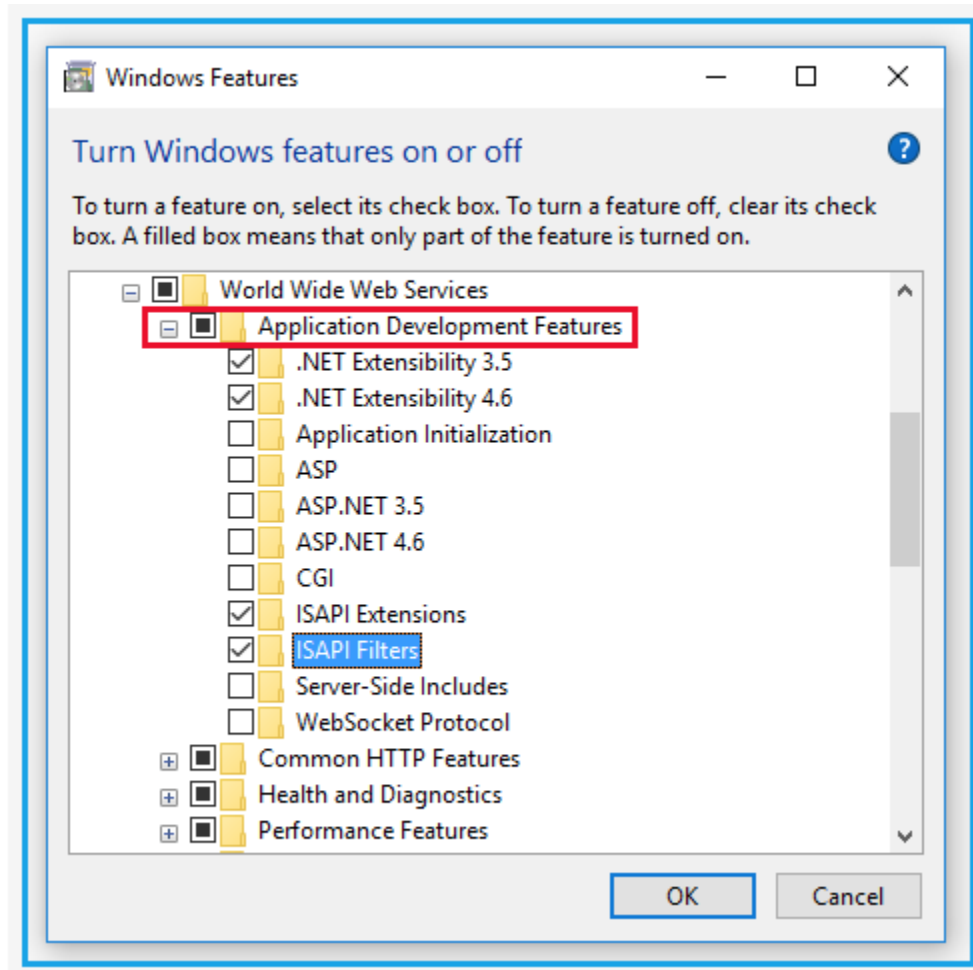
În cazul în care în momentul rularii, browserul v-a returna eroarea 404 Not Found aveți mai jos pașii pe care trebuie să îi efectuați înspire remedierea situației. Accesați următoarea cale:  
**Control Panel -> Programs and Features -> Turn Windows features on or off**



**Navigați până la următoarele opțiuni:**

**Internet Information Services -> World Wide Web Services -> Application Development Features**





Asigurati-va ca aveti optiunile de mai sus bifate/selectate. Selectati mai apoi optiunea OK din fereastra.

### Exercitii

1. Sa se creeze un serviciu Web care sa contina urmatoarele metode :
  - a) Conversie F to C si invers;
  - b) Afisarea orei si datei exacte;
  - c) Afisarea unei liste cu 5 elemente;
  - d) Conversia Intre doua valute (ex: lei - euro).
2. Sa se creeze doi clienti unul de tip Console Application si unul de tip Windows Form Application care sa utilizeze metodele serviciului.

OBS! Interfata grafica pentru aplicatia de tip Win Form App ar putea sa arate precum cea de mai jos:



The screenshot shows a Windows Form titled "Form1". The form contains the following elements:

- A list box labeled "listBox\_Lista" on the left side.
- A button labeled "Add List" below the list box.
- Two input fields for temperature: "Temp C" and "Temp F".
- An input field labeled "Rezultat" below the temperature fields.
- Two buttons for temperature conversion: "FtoC" and "CtoF".
- A label "Data" on the left side, below the "Add List" button.
- An input field labeled "Euro to Ron :" on the right side.
- A second, empty input field below the "Euro to Ron :" label.

**Tema:** Creati un serviciu Web care ofera acces la o baza de date care stocheaza informatii. Apoi, creati un client de tip Windows Form App care utilizeaza serviciul. Serviciul va oferi metode de adaugarea si modificare a informatiilor. Tema aplicatiei este la alegere.