

# Laboratorul 8

## Introducere în ASP.NET

*În cadrul acestui laborator vom învăța să creem aplicații ASP.NET. În prima parte a laboratorului vom învăța cum să facem partea de interfață a aplicațiilor, iar în a doua parte vom învăța cum vom scrie cod sursă pentru controale și forme.*

### 1. Crearea unei noi aplicații ASP.NET

Pentru a crea o nouă aplicație ASP.NET vom parcurge următorii pași:

File → New → Project → Visual C# → ASP.NET Web Application (.NET Framework) → Empty (Figura 1).

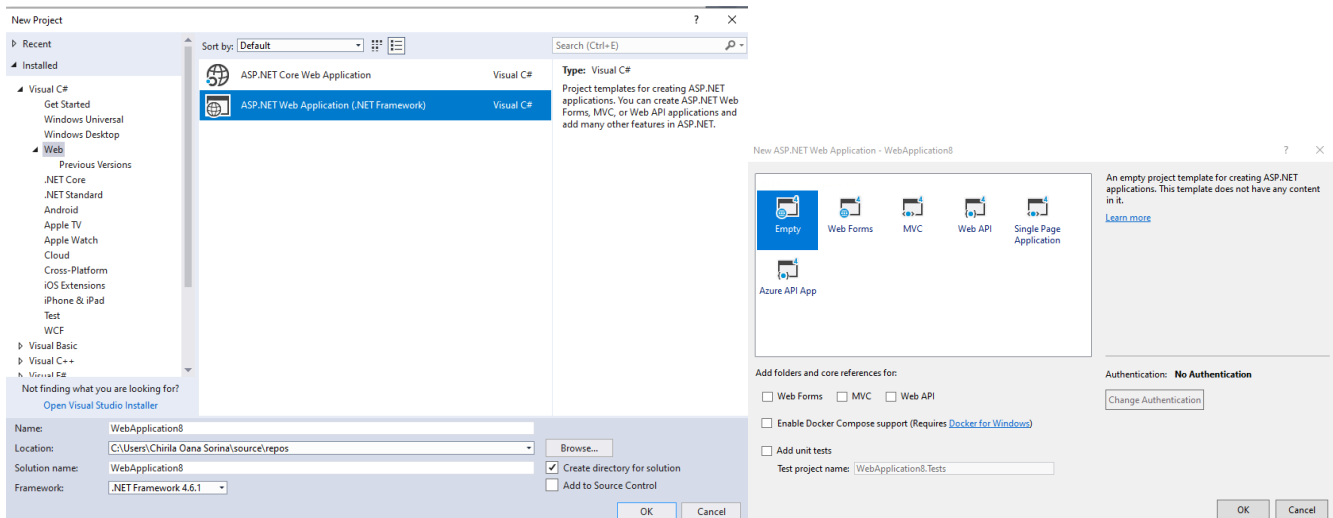


Figura 1. Aplicație ASP.NET

Se adaugă o nouă componentă proiectului nou creat (Add → New Item), Web Form (Figura 2)

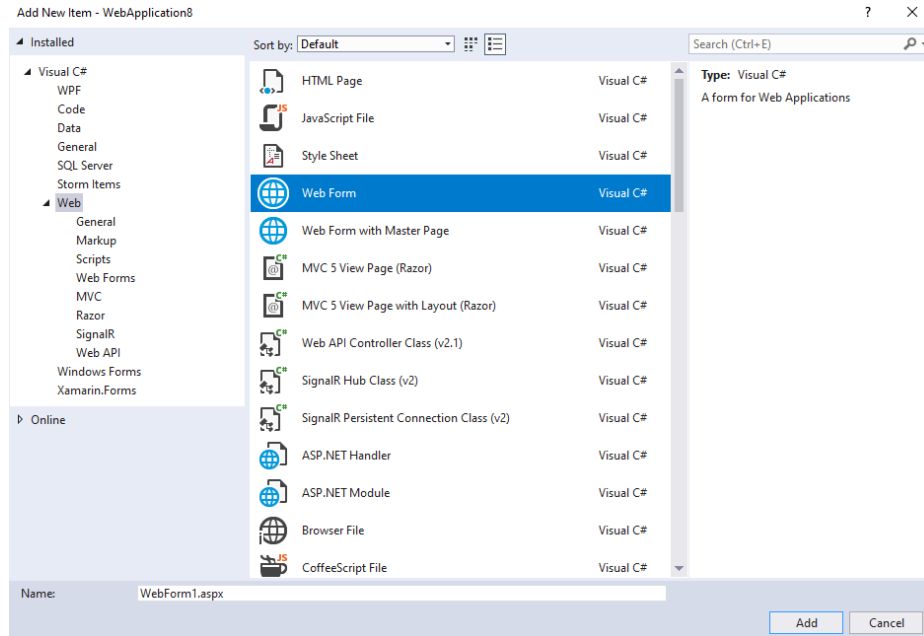


Figura 2. Adăugarea unei noi forme

Forma care se adaugă va avea 2 posibilități de vizualizare (Design sau Source). Dacă o vizualizăm în mod Design vom putea adăuga componente în cadrul ei din toolbox, iar dacă o vizualizăm în mod Design putem scrie cod HTML pentru a crea interfața.

De asemenea, pentru crearea interfeței vom putea adăuga biblioteca bootstrap și putem folosi componente din această bibliotecă pentru crearea interfeței. Tot pentru a crea mai ușor o interfață vom putea folosi template-uri bootstrap și nu numai.

Bootstrap este cel mai popular framework HTML, CSS și JavaScript pentru dezvoltarea de web front-end receptivă și mobilă. Conține șabloane de design bazate pe CSS și JavaScript pentru tipografie, formulare, butoane, navigare și alte componente de interfață. (Figura 3)

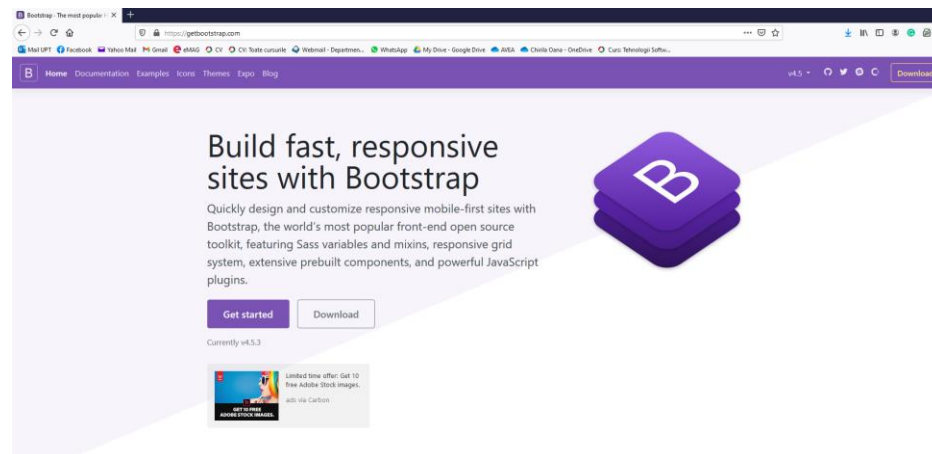


Figura 3. Bootstrap

## 2. Adăugarea elementelor de bootstrap pe interfață

După ce adăugăm o nouă formă sursa formei respective va arăta în felul următor:



```

WebForm1.aspx
1  <%@ Page Language="C#" AutoEventWireup="true" CodeBehind="WebForm1.aspx.cs" Inherits="WebApplication8.WebForm1" %>
2
3  <!DOCTYPE html>
4
5  <html xmlns="http://www.w3.org/1999/xhtml">
6  <head runat="server">
7      <title></title>
8  </head>
9  <body>
10     <form id="form1" runat="server">
11         <div>
12         </div>
13     </form>
14 </body>
15 </html>
  
```

Figura 4. Sursa .aspx a formei web

Pentru a putea adăuga conținut și grafică interfeței putem folosi diverse coduri html din biblioteca bootstrap prezentată mai sus.

De exemplu pentru a adăuga un meniu de navigare putem insera următorul cod în cadrul sursei de mai sus:

1. Înlocuim codul dintre tag-urile <head> cu următorul cod sursă:

```

<html lang="en">
<head>
  <title>Bootstrap Example</title>
  <meta charset="utf-8">
  <meta name="viewport" content="width=device-width, initial-scale=1">
  <link rel="stylesheet" href="https://maxcdn.bootstrapcdn.com/bootstrap/4.5.2/css/bootstrap.min.css">
  <script src="https://ajax.googleapis.com/ajax/libs/jquery/3.5.1/jquery.min.js"></script>
  <script src="https://cdnjs.cloudflare.com/ajax/libs/popper.js/1.16.0/umd/popper.min.js"></script>
  <script src="https://maxcdn.bootstrapcdn.com/bootstrap/4.5.2/js/bootstrap.min.js"></script>
</head>
  
```

2. Adăugăm la tag-ul <body> următoarele linii de cod:

```

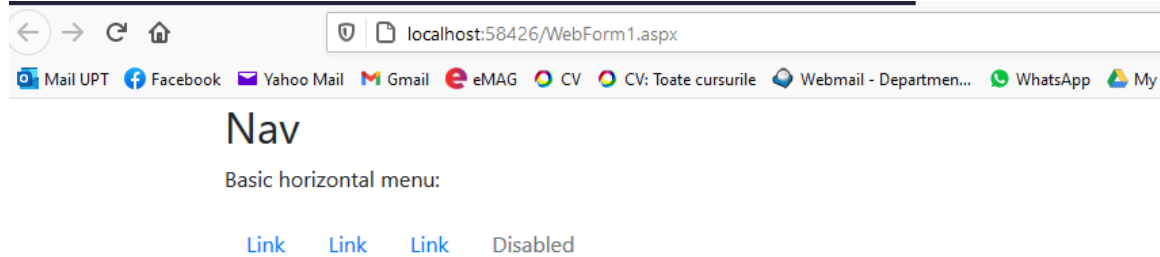
<body>
  <form id="form1" runat="server">
    <div>

<div class="container">
  <h2>Nav</h2>
  <p>Basic horizontal menu:</p>
  <ul class="nav">
    <li class="nav-item">
      <a class="nav-link" href="#">Link</a>
    </li>
  
```

```
<li class="nav-item">
  <a class="nav-link" href="#">Link</a>
</li>
<li class="nav-item">
  <a class="nav-link" href="#">Link</a>
</li>
<li class="nav-item">
  <a class="nav-link disabled" href="#">Disabled</a>
</li>
</ul>
</div>

</div>
</form>
</body>
```

Rezultatul rulării acestui cod va fi următorul:



3. Dacă în locul meniului dorim să punem o bară de navigare (navbar) în locul codului de la punctul 2 vom insera următorul cod:

```
<body>
  <form id="form1" runat="server">
    <div>

<nav class="navbar navbar-expand-sm bg-light">
  <ul class="navbar-nav">
    <li class="nav-item">
      <a class="nav-link" href="#">Link 1</a>
    </li>
    <li class="nav-item">
      <a class="nav-link" href="#">Link 2</a>
    </li>
    <li class="nav-item">
      <a class="nav-link" href="#">Link 3</a>
    </li>
  </ul>
</nav>
<br>

<div class="container-fluid">
  <h3>Exemplu NavBar</h3>
  <p>O bară de navigare este un antet de navigare plasat în partea de sus a paginii.</p>
</div>

</div>
</form>
</body>
```

Rezultatul în acest caz va fi următorul:

[Link 1](#) [Link 2](#) [Link 3](#)

## Exemplu NavBar

O bară de navigare este un antet de navigare plasat în partea de sus a paginii.

În acest fel putem să adăugăm conținutul vizual al paginii noastre. Pentru mai multe exemple de folosire a elementelor bootstrap accesați următoarele tutoriale:

<https://www.w3schools.com/bootstrap4/>

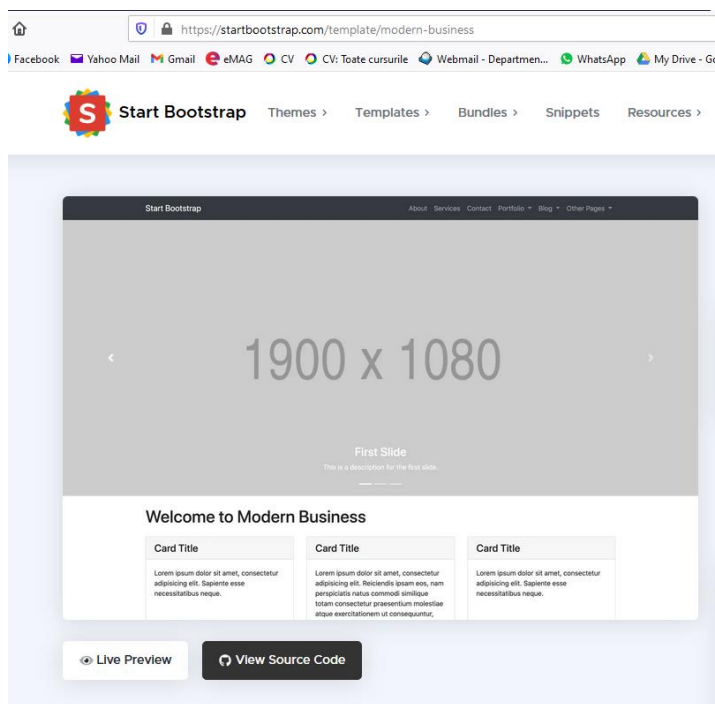
<https://getbootstrap.com/docs/4.0/examples/>

<https://startbootstrap.com/snippets>

<https://www.tutorialrepublic.com/twitter-bootstrap-examples.php>

De asemenea putem folosi pentru o anumită pagină un template în următorul fel:

Vom alege de pe <https://startbootstrap.com/> un template care se potrivește pentru aplicația noastră.

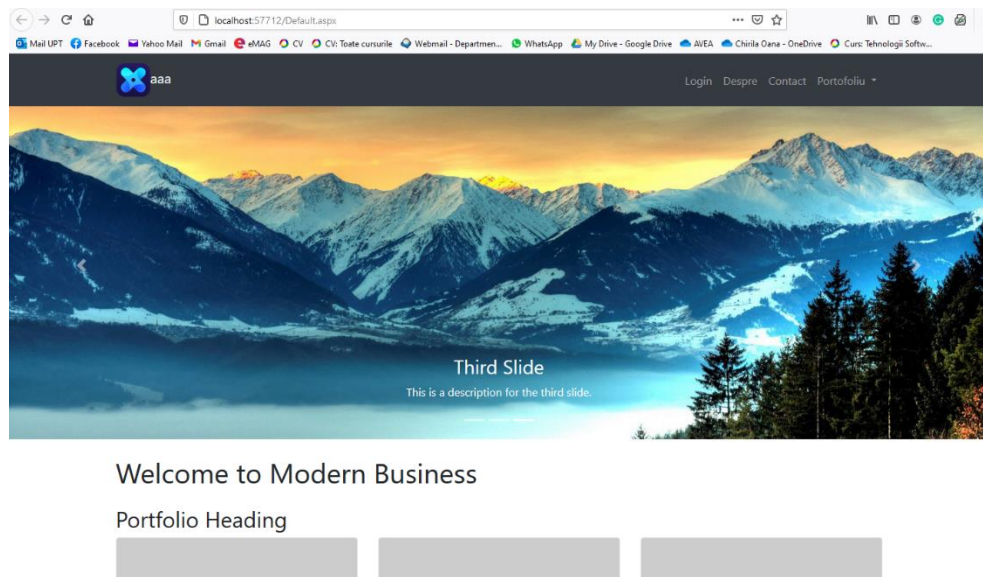


După ce am ales un template vom apăsa butonul View Source Code. De pe pagina care se deschide vom copia codul sursă pentru pagina principală, sau pentru toate paginile de care avem nevoie. De asemenea vom copia toate folderele cu fișiere necesare paginii noastre:

css	Bootstrap 4.5.3 and dependencies update
js	Update contact_me.js to prevent additional dialogs
mail	dev environment
vendor	Bootstrap 4.5.3 and dependencies update
.gitignore	dev environment
.travis.yml	update dependencies and copyright dates
404.html	update dependencies and build
LICENSE	update dependencies and build
README.md	Bootstrap 4.5.3 and dependencies update
about.html	fix HTML validation errors
blog-home-1.html	fix HTML validation errors
blog-home-2.html	fix HTML validation errors
blog-post.html	fix HTML validation errors
contact.html	fix HTML validation errors
faq.html	fix HTML validation errors
full-width.html	fix HTML validation errors
gulpfile.js	update dependencies
index.html	fix HTML validation errors
package-lock.json	Bootstrap 4.5.3 and dependencies update
package.json	Bootstrap 4.5.3 and dependencies update

Folderele respective le vom copia in cadrul proiectului nostru, iar in sursa ferestrei vom lua continutul fișierului index.html.

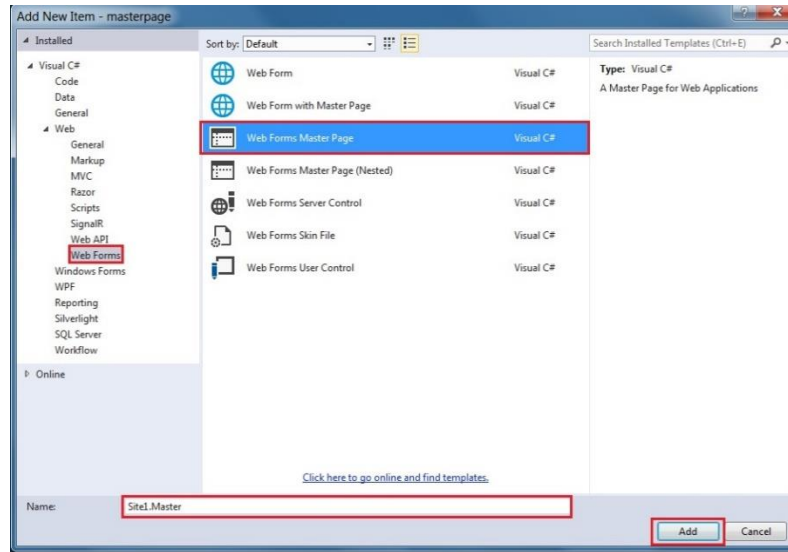
După editarea template-ului respectiv s-a obținut următorul rezultat:



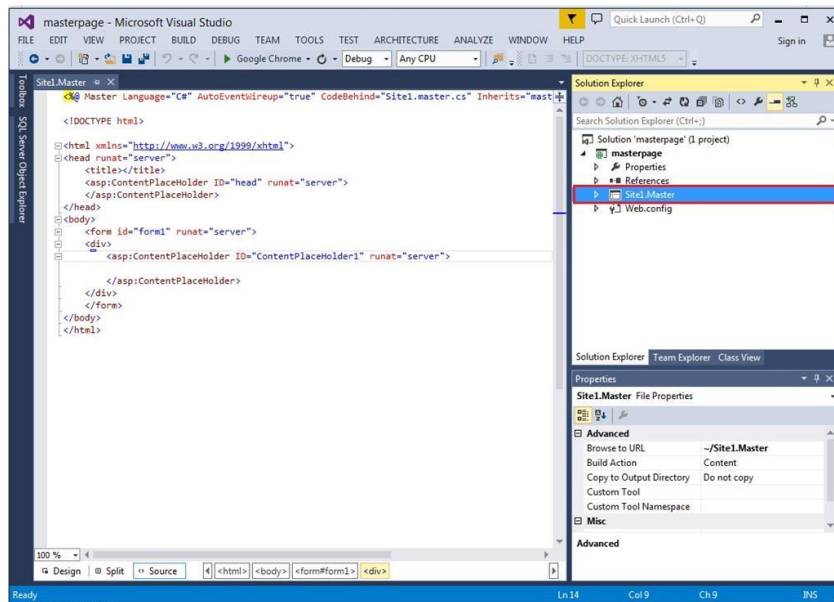
#### 4. Adăugarea unei forme Master Page

O pagină principală (Master Page) oferă aspectul și funcționalitatea altor pagini. Pentru a crea un master page parcurgem următorii pași:

1. Se adaugă o nouă componentă la proiect (Add → New Item)



2. După ce faceți click pe butonul Add, pagina principală „site1.master” se adaugă la proiectul nostru. Faceți clic pe site1.master în Solution Explorer (prezentat în imagine):



3. Proiectați pagina principală, utilizând HTML. Și aici se pot folosi componentele bootstrap.
4. După crearea paginii principale, în momentul adăugării unei noi forme în cadrul proiectului, vom selecta Web Form with Master Page și vom alege pagina principală care dorim să o moștenească.

## 5. Exemplu aplicație ASP.NET simplă

Vom crea un nou proiect ASP.Net, vom adăuga un Master Page, în care vom adăuga următorul cod sursă:

```
<%@ Master Language="C#" AutoEventWireup="true" CodeBehind="Site1.master.cs"
Inherits="Exemplu_MasterPage.Site1" %>

<!DOCTYPE html>

<html lang="en">
<head runat="server">
    <title>Aplicație ASP.NET</title>
    <link href="css/my.css" rel="stylesheet" />
    <link rel="stylesheet" type="text/css" href="my.css">
    <asp:ContentPlaceHolder ID="head" runat="server">
    </asp:ContentPlaceHolder>
</head>

<body>
<header id="header">
<h1>Aplicație ASP.NET</h1>
</header>
<nav id="nav">
    <ul>
        <li><a href="WebForm1.aspx">Acasa</a></li>
        <li><a href="#">Despre</a></li>
        <li><a href="#">Contact</a></li>
    </ul>
</nav>

    <div id="con">
        <asp:ContentPlaceHolder ID="ContentPlaceHolder1" runat="server">
        </asp:ContentPlaceHolder>
    </div>

<footer id="footer">
    Chirila Oana Sorina - Medii si tehnologii de programare - ASP.NET
</footer>
</body>
</html>
```

Fișierul my.css folosit îl vom adăuga în cadrul aplicației noastre, în folderul css. Acest fișier are următoarea structură:

```
#header{
    color: #247BA0;
    text-align: center;
    font-size: 20px;
}
#nav{
    background-color:#940200;
    padding: 5px;
}
ul{
    list-style-type: none;
}
li a {
    color: #F1FAEE;
    font-size: 30px;
    column-width: 5%;
}
```



```

    li
    {
        display: inline;
        padding-left: 2px;
        column-width: 20px;
    }
    a{
text-decoration: none;
margin-left:20px
    }
    li a:hover{
        background-color: #F3FFBD;
        color: #FF1654;
        padding:1%;
    }
    #footer{
        background-color: #82020d;
        text-align:center;
        padding-bottom: 5%;
        font-size: 20px;
    }
    #con{
        border:double;
        border-color:burlywood;
    }
}

```

După crearea paginii principale vom adăuga o nouă pagină de tip Web Form with Master Page la care vom selecta Master Page-ul creat anterior. În cadrul acestei forme, în formatul Design vom adăuga din meniul Table un tabel care ne va ajuta să plasăm controalele pe formă într-o manieră mai ordonată. Vom adăuga controale pentru a crea o fereastră de autentificare.

În același mod vom adăuga încă o formă în care se va adăuga doar un control textBox care va primi o variabilă din forma precedentă.

Codul sursă pentru forma de login este:

```

using System;
using System.Collections.Generic;
using System.IO;
using System.Linq;
using System.Web;
using System.Web.UI;
using System.Web.UI.WebControls;

namespace Exemplu_MasterPage
{
    public partial class WebForm1 : System.Web.UI.Page
    {
        protected void Page_Load(object sender, EventArgs e)
        {
            string[] utilizatori = File.ReadAllLines(Server.MapPath("~/Fisiere/") + "utilizatori.txt");
            foreach (var line in utilizatori)
            {
                string[] inregistrare = line.Split(',');
                DropDownList1.Items.Add(inregistrare[0]);
            }
        }

        protected void Button1_Click(object sender, EventArgs e)
        {
            string[] utilizatori = File.ReadAllLines(Server.MapPath("~/Fisiere/") + "utilizatori.txt");

```

```
foreach (var line in utilizatori)
{
    string[] inregistrare = line.Split(',');

    if ((DropDownList1.Text).Equals(inregistrare[0]))
    {
        if (TextBoxPWD.Text.Trim().Equals(inregistrare[1].Trim()))
        {
            Response.Redirect("Despre.aspx?utilizator="+DropDownList1.Text.Trim());
        }
        else
        {
            string script = "alert(\"Parola incorecta!\");";
            ScriptManager.RegisterStartupScript(this, GetType(), "ServerControlScript", script, true);
        }
    }
}
}
```

Codul sursă pentru a doua formă este:

```
using System;
using System.Collections.Generic;
using System.Linq;
using System.Web;
using System.Web.UI;
using System.Web.UI.WebControls;

namespace Exemplu_MasterPage
{
    public partial class Despre : System.Web.UI.Page
    {
        protected void Page_Load(object sender, EventArgs e)
        {
            String nume = Request.QueryString["utilizator"];
            TextBox1.Text = nume;
        }
    }
}
```

Rezultatul la rulare este următorul:

## Aplicație ASP.NET

Acasa Despre Contact

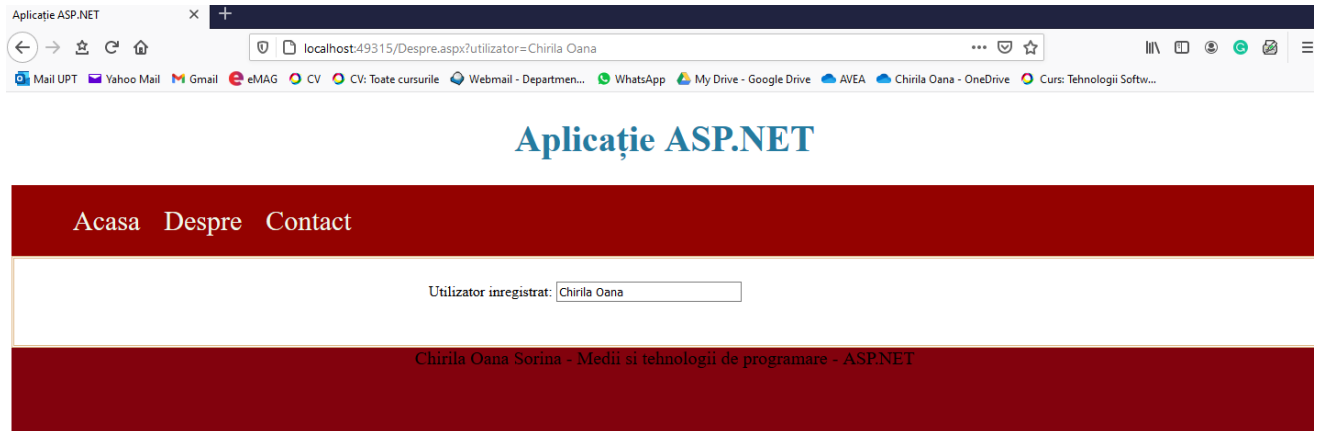
**Autentificare**

Utilizator

Parola

Autentificare

Chirila Oana Sorina - Medii si tehnologii de programare - ASP.NET



## Probleme Propuse

Realizați o aplicație la alegere cu un design la alegere, cu minim 3 pagini în care să se afișeze diferite informații. Aplicația să conțină minim o citire/scriere din fișier și o transmitere de parametrii de la o formă la alta. (Exemplu: O formă de Login, se transmite numele utilizatorului, în a doua formă utilizatorul completează un formular de contact care se salvează în fișier, forma a 3-a conține informații despre aplicație.)