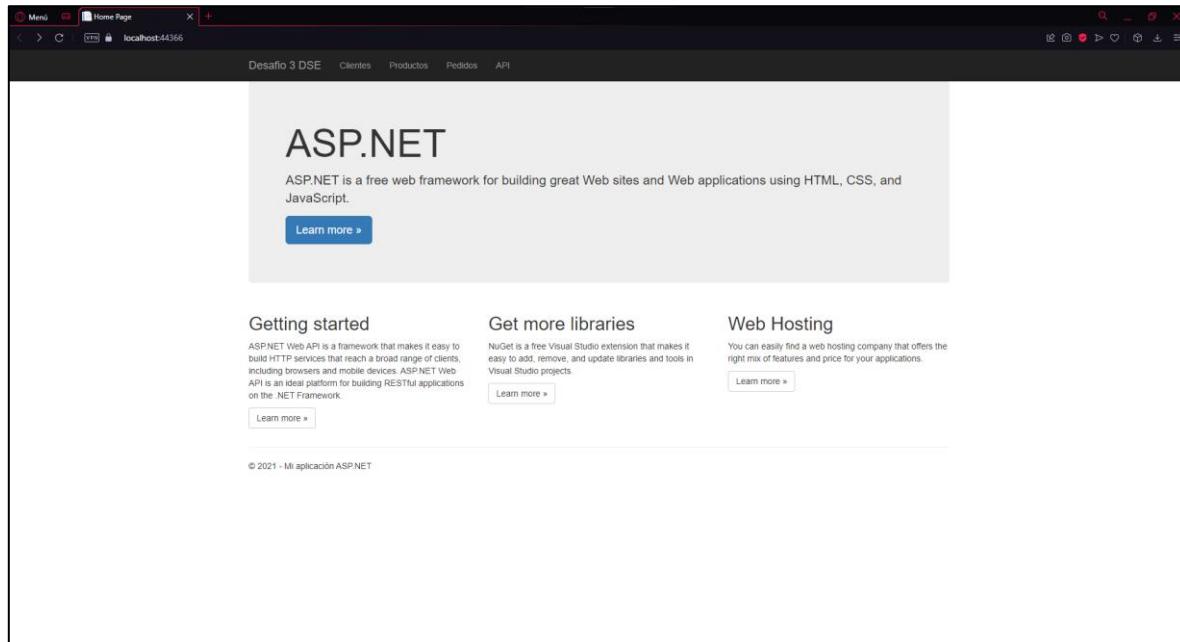
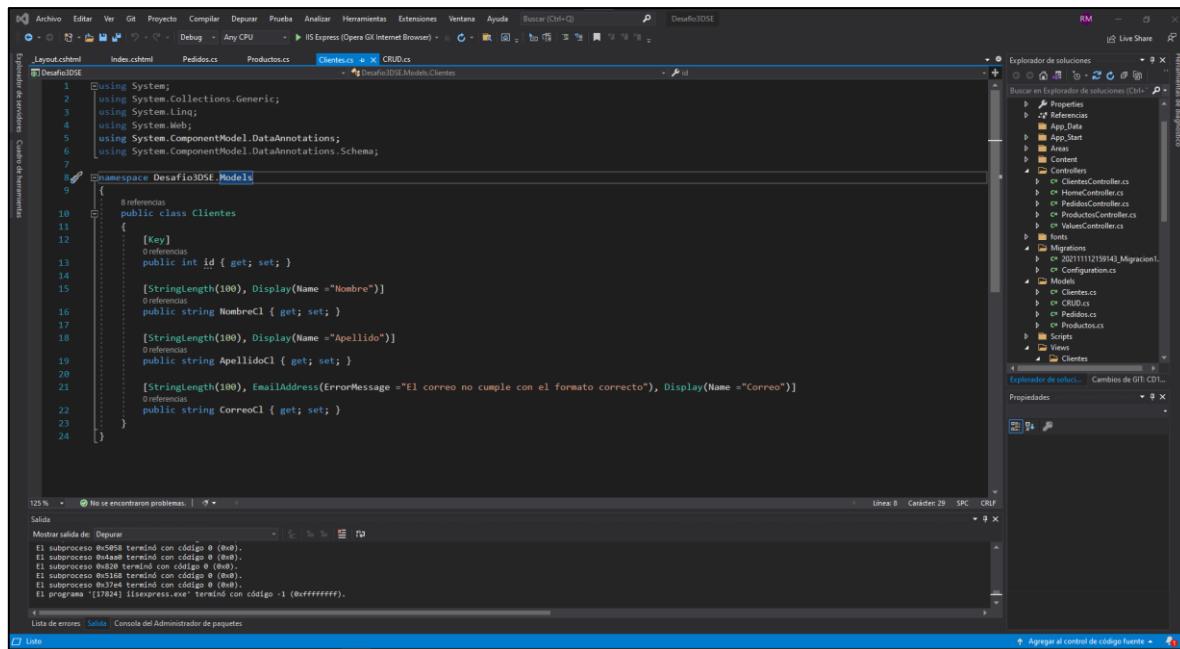


Tercer Desafío Práctico

DESARROLLO DE SOFTWARE EMPRESARIAL 04L

Roberto Fernando Funes Morán – FM180247
12 de noviembre de 2021

Desarrollo del ejercicio:**1. Index****2. Tabla Clientes****a. Clientes.cs**

b. Clientes/Index.cshtml

The screenshot shows the Microsoft Visual Studio interface. The code editor displays the `Index.cshtml` file, which contains HTML and C# code for displaying a list of clients. The Solution Explorer on the right shows the project structure with files like `ClientesController.cs`, `ProductosController.cs`, `CRUD.cs`, and `Index.cshtml`. The status bar at the bottom indicates "No se encontraron problemas" (No problems found).

```

@model IEnumerable<DesafioDSE.Models.Clientes>
@{
    ViewBag.Title = "Index";
}


## Clients



Añadir cliente



| Nombre     | Apellido | Correo                 |
|------------|----------|------------------------|
| John Doe   | J. Doe   | john.doe@example.com   |
| Jane Doe   | J. Doe   | jane.doe@example.com   |
| John Smith | J. Smith | john.smith@example.com |
| Jane Smith | J. Smith | jane.smith@example.com |

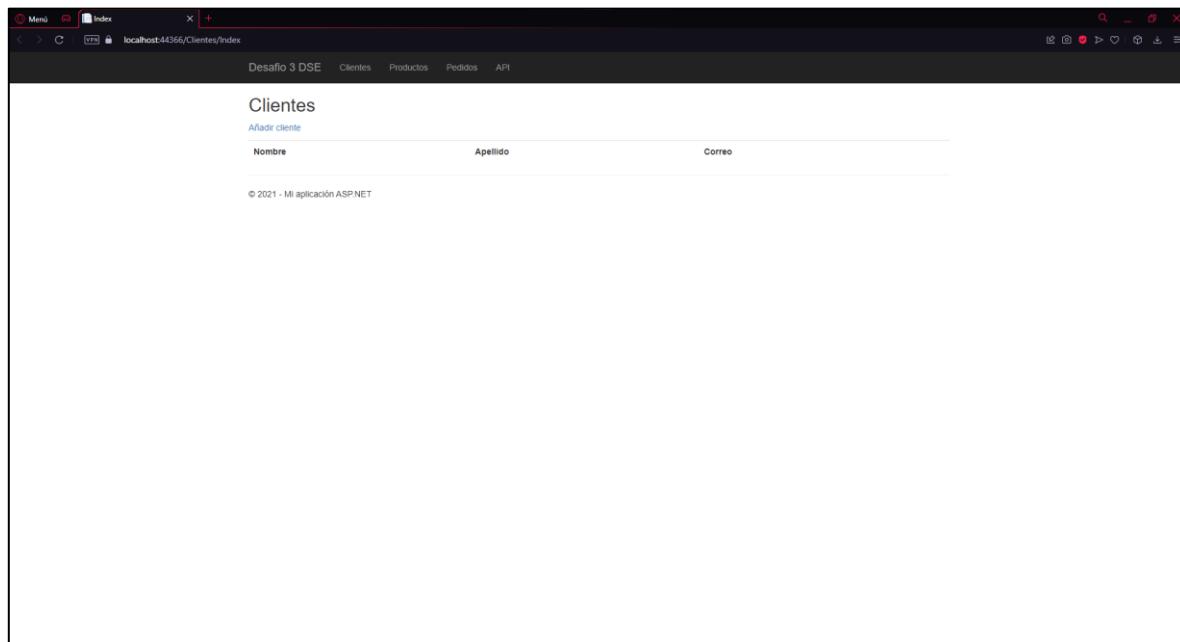


Actions:



- Edit
- Details
- Delete

```



c. Clientes/Create.cshtml

```

Index.cshtml
1 <div><a href="#">Home</a></div>
2 <div><h2>Añadir Clientes</h2>
3 
4 <div>
5   <h3>Datos del cliente</h3>
6   <form>
7     <div>
8       <div><label>Nombre:</label><input type="text" name="NombreCl"/></div>
9       <div><label>Apellido:</label><input type="text" name="ApellidoCl"/></div>
10      <div><label>Correo:</label><input type="text" name="CorreoCl"/></div>
11      <div><input type="submit" value="Añadir" class="btn btn-default"/></div>
12    </div>
13  </form>
14 </div>
15 
16 <div><a href="#">Home</a></div>
17 
```

```

Index.cshtml
1 <div><a href="#">Home</a></div>
2 <div><h2>Añadir Clientes</h2>
3 
4 <div>
5   <h3>Datos del cliente</h3>
6   <form>
7     <div>
8       <div><label>Nombre:</label><input type="text" name="NombreCl"/></div>
9       <div><label>Apellido:</label><input type="text" name="ApellidoCl"/></div>
10      <div><label>Correo:</label><input type="text" name="CorreoCl"/></div>
11      <div><input type="submit" value="Añadir" class="btn btn-default"/></div>
12    </div>
13  </form>
14 </div>
15 
16 <div><a href="#">Home</a></div>
17 
```

Añadir clientes

Ciudades

Nombre

Apellido

Correo

[Regresar](#)

© 2021 - Mi aplicación ASP.NET

Clientes

Añadir cliente

Nombre	Apellido	Correo	
Fernando	Funes	fernando99funes@gmail.com	Editar Detalles Borrar

© 2021 - Mi aplicación ASP.NET

d. Clientes/Edit.cshtml

```

@model DesafioTOS.E.Models.Clientes
@{
    ViewBag.Title = "Edit";
}


## Edit



@using (Html.BeginForm())
    {
        @Html.AntiForgeryToken()
        <div class="form-horizontal">
            <div>
                <div>
                    <div class="form-group">
                        <label for="Nombre1" class="control-label col-md-2">
                            @Html.LabelFor(model => model.Nombre1, htmlAttributes: new { @class = "control-label col-md-2" })
                        </label>
                        <div class="col-md-10">
                            @Html.EditorFor(model => model.Nombre1, new { htmlAttributes = new { @class = "form-control" } })
                            @Html.ValidationMessageFor(model => model.Nombre1, "", new { @class = "text-danger" })
                        </div>
                    </div>
                    <div class="form-group">
                        <label for="Apellido1" class="control-label col-md-2">
                            @Html.LabelFor(model => model.Apellido1, htmlAttributes: new { @class = "control-label col-md-2" })
                        </label>
                        <div class="col-md-10">
                            @Html.EditorFor(model => model.Apellido1, new { htmlAttributes = new { @class = "form-control" } })
                            @Html.ValidationMessageFor(model => model.Apellido1, "", new { @class = "text-danger" })
                        </div>
                    </div>
                    <div class="form-group">
                        <label for="Correo1" class="control-label col-md-2">
                            @Html.LabelFor(model => model.Correcto1, htmlAttributes: new { @class = "control-label col-md-2" })
                        </label>
                        <div class="col-md-10">
                            @Html.EditorFor(model => model.Correcto1, new { htmlAttributes = new { @class = "form-control" } })
                            @Html.ValidationMessageFor(model => model.Correcto1, "", new { @class = "text-danger" })
                        </div>
                    </div>
                </div>
                <div class="form-group">
                    <div class="col-md-offset-2 col-md-10">
                        <input type="submit" value="Guardar" class="btn btn-default" />
                    </div>
                </div>
            </div>
        </div>
    }


```

```

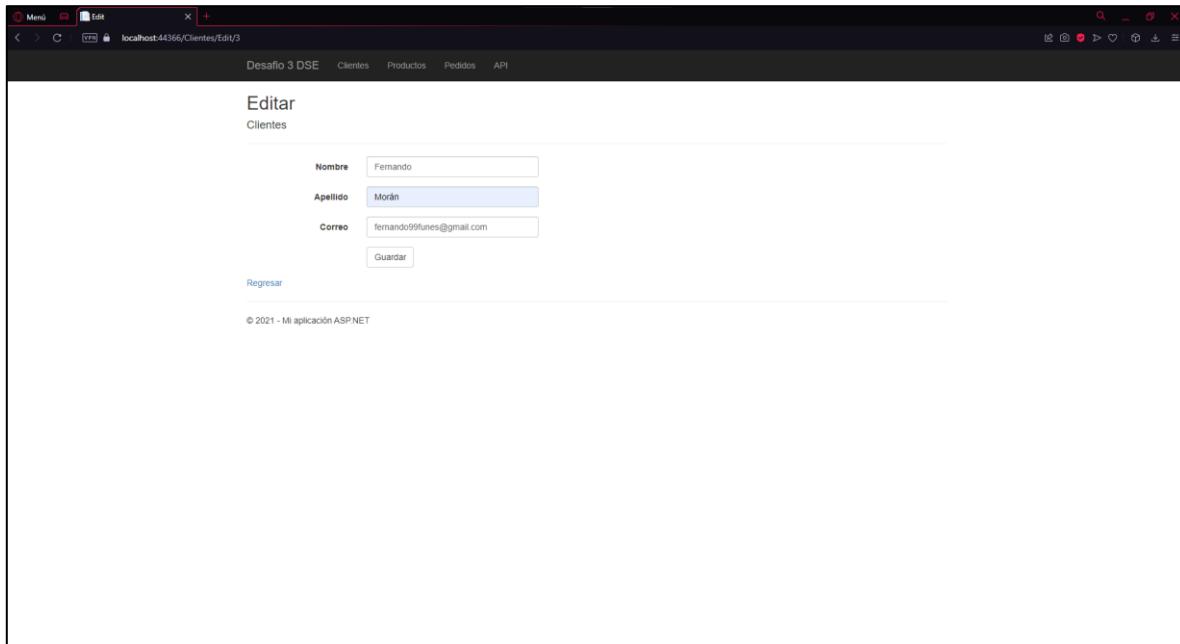
@model DesafioTOS.E.Models.Clientes
@{
    ViewBag.Title = "Edit";
}


## Edit



@using (Html.BeginForm())
    {
        @Html.AntiForgeryToken()
        <div class="form-horizontal">
            <div>
                <div>
                    <div class="form-group">
                        <label for="Nombre1" class="control-label col-md-2">
                            @Html.LabelFor(model => model.Nombre1, htmlAttributes: new { @class = "control-label col-md-2" })
                        </label>
                        <div class="col-md-10">
                            @Html.EditorFor(model => model.Nombre1, new { htmlAttributes = new { @class = "form-control" } })
                            @Html.ValidationMessageFor(model => model.Nombre1, "", new { @class = "text-danger" })
                        </div>
                    </div>
                    <div class="form-group">
                        <label for="Apellido1" class="control-label col-md-2">
                            @Html.LabelFor(model => model.Apellido1, htmlAttributes: new { @class = "control-label col-md-2" })
                        </label>
                        <div class="col-md-10">
                            @Html.EditorFor(model => model.Apellido1, new { htmlAttributes = new { @class = "form-control" } })
                            @Html.ValidationMessageFor(model => model.Apellido1, "", new { @class = "text-danger" })
                        </div>
                    </div>
                    <div class="form-group">
                        <label for="Correo1" class="control-label col-md-2">
                            @Html.LabelFor(model => model.Correcto1, htmlAttributes: new { @class = "control-label col-md-2" })
                        </label>
                        <div class="col-md-10">
                            @Html.EditorFor(model => model.Correcto1, new { htmlAttributes = new { @class = "form-control" } })
                            @Html.ValidationMessageFor(model => model.Correcto1, "", new { @class = "text-danger" })
                        </div>
                    </div>
                </div>
                <div class="form-group">
                    <div class="col-md-offset-2 col-md-10">
                        <input type="submit" value="Guardar" class="btn btn-default" />
                    </div>
                </div>
            </div>
        </div>
    }


```



e. Clientes/Details.cshtml

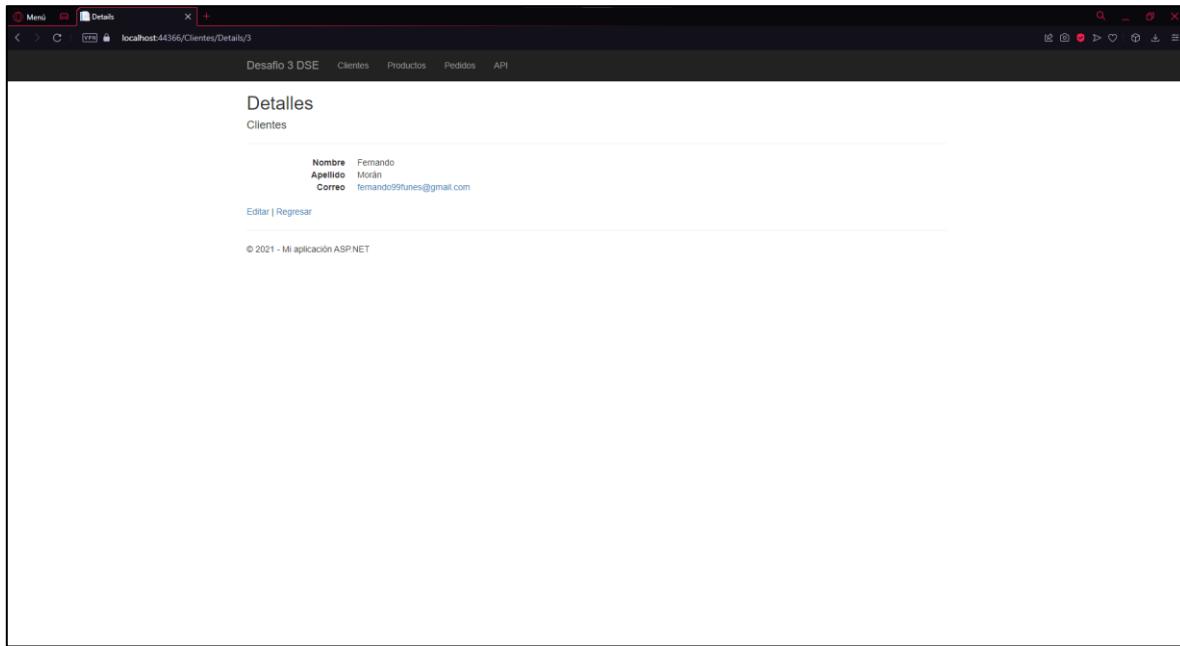
Screenshot of the Visual Studio IDE showing the code editor for the Details.cshtml file. The code is an ASP.NET MVC view for displaying client details.

```

<%@ Page Title="<%= ViewData["Title"] %>" %>
<h2>Detalles</h2>
<div>
    <h3>Datos del Cliente</h3>
    <table class="table">
        <tr>
            <td><label for="NombreC1">Nombre</label></td>
            <td><input type="text" value="<%= Model.NombreC1 %>" /></td>
        </tr>
        <tr>
            <td><label for="ApellidoC1">Apellido</label></td>
            <td><input type="text" value="<%= Model.ApellidoC1 %>" /></td>
        </tr>
        <tr>
            <td><label for="CorreoC1">Correo</label></td>
            <td><input type="text" value="<%= Model.CorreoC1 %>" /></td>
        </tr>
    </table>
    <p><a href="#">Editar</a> | <a href="#">Regresar</a></p>
</div>

```

The code includes a table to display client details (NombreC1, ApellidoC1, CorreoC1) and links for editing and returning.



f. Clientes/Delete.cshtml

```

@model DesafioDSE.Models.Cliente

@{
    ViewBag.Title = "Delete";
}



## Borrar



Are you sure you want to delete this?



|                  |                   |
|------------------|-------------------|
| <b>Nombre:</b>   | @Model.NombreCl   |
| <b>Apellido:</b> | @Model.ApellidoCl |
| <b>Correo:</b>   | @Model.CorreoCl   |



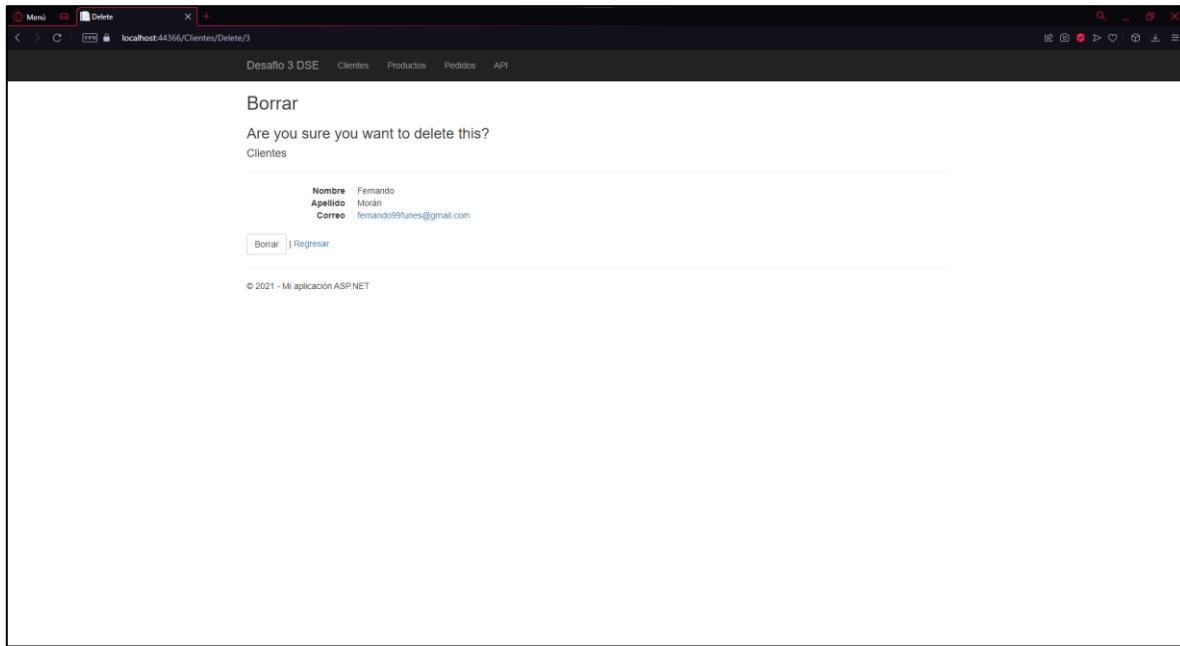
@using (Html.BeginForm())
    {
        @Html.AntiForgeryToken()

Regresar


    }


```



3. Tabla Productos

a. Products.cs

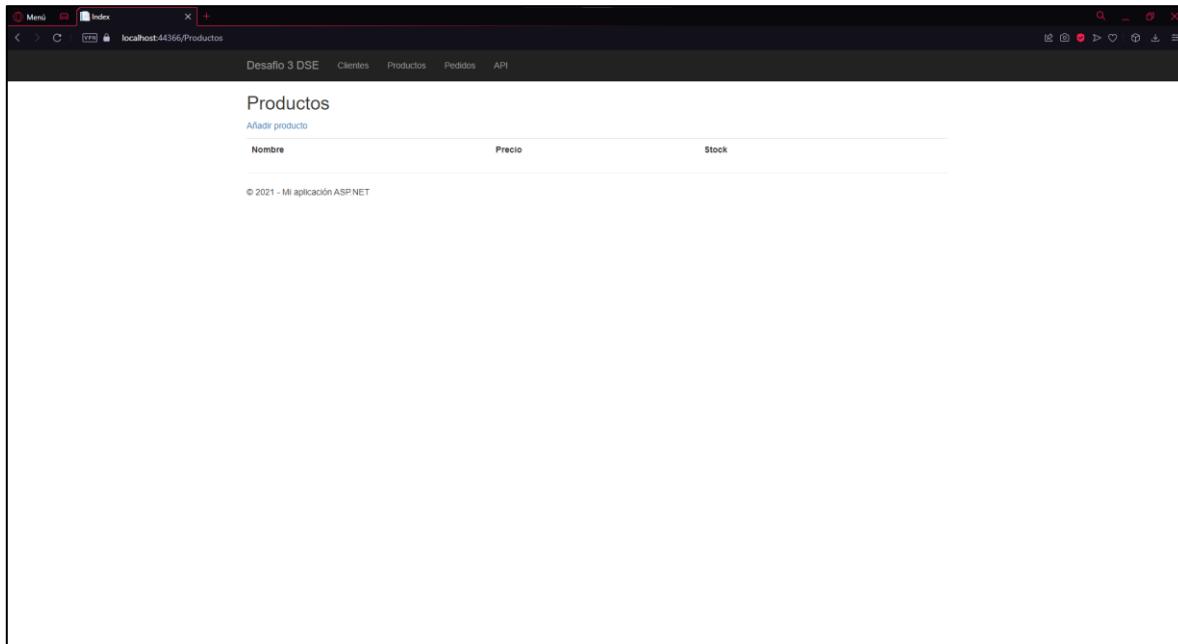
```

16     <tr>
17         <td><input type="checkbox" name="model.NombreProd" /></td>
18         <td><input type="text" name="model.ApellidoProd" /></td>
19         <td><input type="text" name="model.CorrreoProd" /></td>
20     </tr>
21 
22     <tr>
23         <td><input type="button" value="Actualizar" /></td>
24         <td><input type="button" value="Nuevo" /></td>
25         <td><input type="button" value="Borrar" /></td>
26     </tr>
27 
```

b. Products/Index.cshtml

```

1     <table border="1">
2         <thead>
3             <tr>
4                 <th>NombreProd</th>
5                 <th>ApellidoProd</th>
6                 <th>CantidadProd</th>
7             </tr>
8         </thead>
9         <tbody>
10            @foreach (var item in Model) {
11                <tr>
12                    <td>@item.NombreProd</td>
13                    <td>@item.ApellidoProd</td>
14                    <td>@item.CantidadProd</td>
15                </tr>
16            }
17        </tbody>
18    </table>
19 
```



c. Products/Create.cshtml

```

<div class="row">
    <div class="col-md-4">
        <form method="post">
            <div class="form-group">
                <label for="NombreProd">Nombre del producto
                <input type="text" id="NombreProd" name="NombreProd"/>
                <small>Este campo es obligatorio.
            </div>
            <div class="form-group">
                <label for="PrecioProd">Precio
                <input type="text" id="PrecioProd" name="PrecioProd"/>
                <small>Este campo es obligatorio.
            </div>
            <div class="form-group">
                <label for="CantidadProd">Cantidad
                <input type="text" id="CantidadProd" name="CantidadProd"/>
                <small>Este campo es obligatorio.
            </div>
            <div class="form-group">
                <input type="submit" value="Crear"/>
            </div>
        </form>
    </div>

```

```

<div class="form-group">
    <label for="NombreProd">NombreProd</label>
    <input type="text" id="NombreProd" name="NombreProd" value="Gas" class="form-control" />
    <small>Este campo es obligatorio</small>
</div>
<div class="form-group">
    <label for="PrecioProd">PrecioProd</label>
    <input type="text" id="PrecioProd" name="PrecioProd" value="20.00" class="form-control" />
    <small>Este campo es obligatorio</small>
</div>
<div class="form-group">
    <label for="StockProd">StockProd</label>
    <input type="text" id="StockProd" name="StockProd" value="5" class="form-control" />
    <small>Este campo es obligatorio</small>
</div>
<div class="form-group">
    <input type="submit" value="Añadir" class="btn btn-default" />
    <a href="#" class="btn btn-link" style="float: right;">Regresar
</div>

```

Lista de errores:

- El elemento 'meta' no se puede anidar en el elemento 'html'.

Añadir Productos

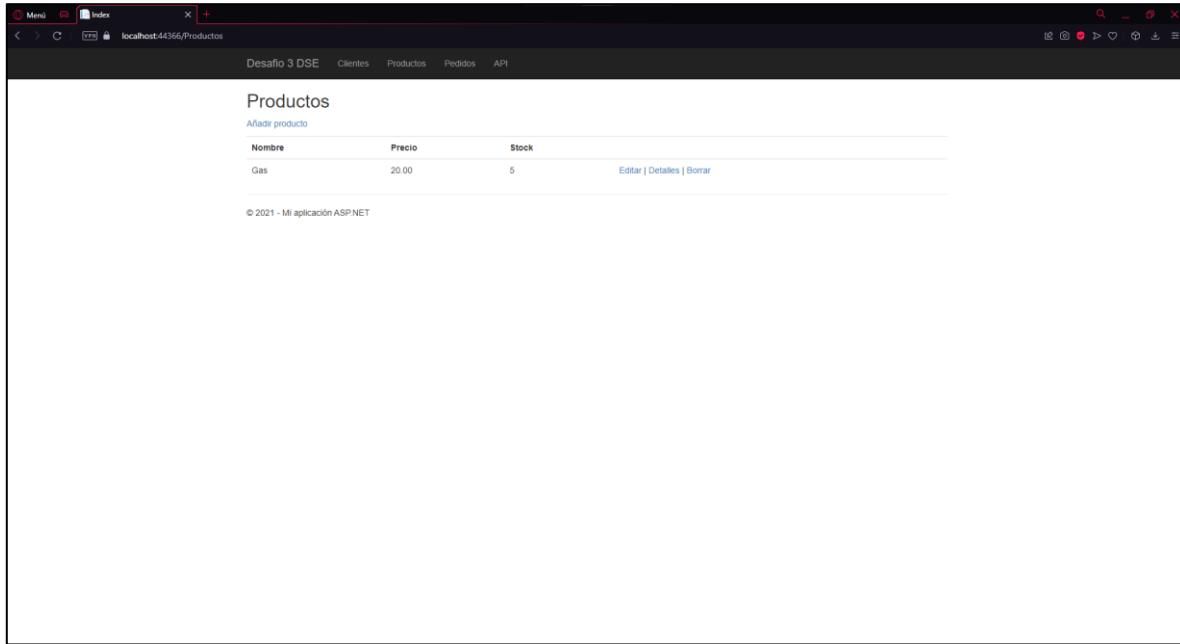
Productos

Nombre	Gas
Precio	20.00
Stock	5

Añadir

Regresar

© 2021 - Mi aplicación ASP.NET



d. Products/Edit.cshtml

```

@model DesafioDSE.Models.Productos

@{
    ViewBag.Title = "Edit";
}



## Edit



@Html.LabelFor(model => model.NombreProd, htmlAttributes: new { @class = "control-label col-md-2" })
    @Html.EditorFor(model => model.NombreProd, new { htmlAttributes = new { @class = "form-control" } })
    @Html.ValidationMessageFor(model => model.NombreProd, "", new { @class = "text-danger" })



@Html.LabelFor(model => model.PrecioProd, htmlAttributes: new { @class = "control-label col-md-2" })
    @Html.EditorFor(model => model.PrecioProd, new { htmlAttributes = new { @class = "form-control" } })
    @Html.ValidationMessageFor(model => model.PrecioProd, "", new { @class = "text-danger" })



@Html.LabelFor(model => model.CantidadProd, htmlAttributes: new { @class = "control-label col-md-2" })
    @Html.EditorFor(model => model.CantidadProd, new { htmlAttributes = new { @class = "form-control" } })
    @Html.ValidationMessageFor(model => model.CantidadProd, "", new { @class = "text-danger" })


```

```

@model DesafioDSE.Models.Productos

@{
    ViewBag.Title = "Edit";
}



@Html.LabelFor(model => model.NombreProd, htmlAttributes: new { @class = "control-label col-md-2" })
    @Html.EditorFor(model => model.NombreProd, new { htmlAttributes = new { @class = "form-control" } })
    @Html.ValidationMessageFor(model => model.NombreProd, "", new { @class = "text-danger" })



@Html.LabelFor(model => model.PrecioProd, htmlAttributes: new { @class = "control-label col-md-2" })
    @Html.EditorFor(model => model.PrecioProd, new { htmlAttributes = new { @class = "form-control" } })
    @Html.ValidationMessageFor(model => model.PrecioProd, "", new { @class = "text-danger" })



@Html.LabelFor(model => model.CantidadProd, htmlAttributes: new { @class = "control-label col-md-2" })
    @Html.EditorFor(model => model.CantidadProd, new { htmlAttributes = new { @class = "form-control" } })
    @Html.ValidationMessageFor(model => model.CantidadProd, "", new { @class = "text-danger" })



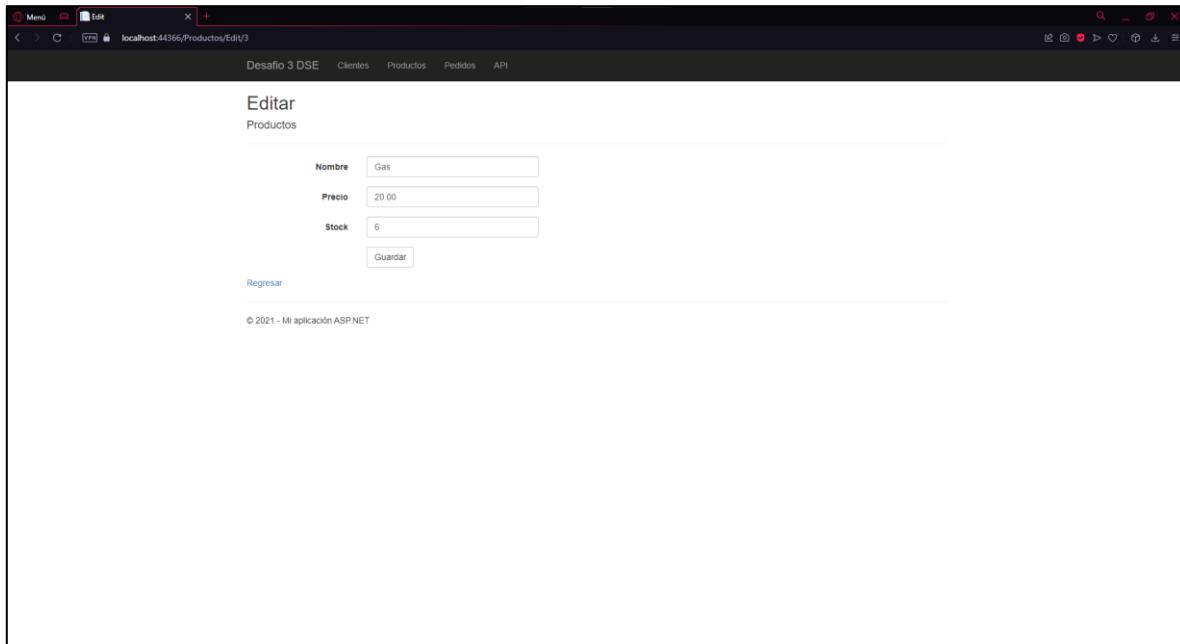
Regresar
Index



Script

```
Scripts.Render("~/bundles/queryval")
```


```

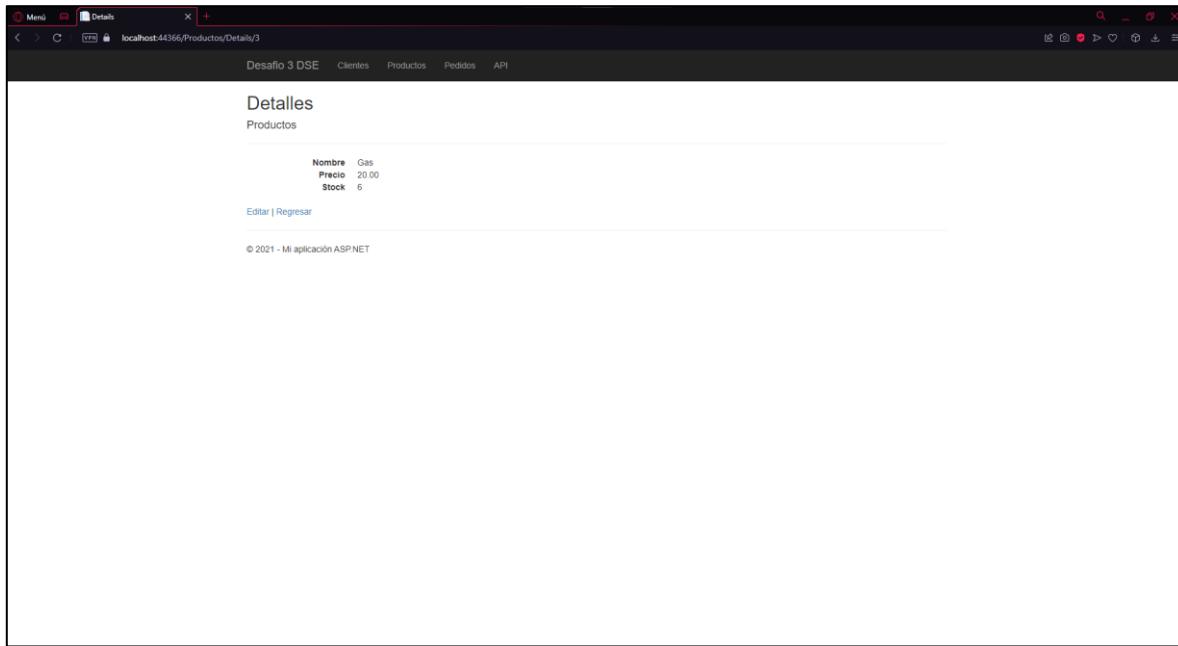


e. Products/Details.cshtml

```

1 <model>DesafioDSE.Models.Producto</model>
2 
3 <h1>Vista de detalle</h1>
4 
5 <h2>Detalles</h2>
6 
7 <div>
8   <h3>Productos</h3>
9   <div class="dl-horizontal">
10     <dt><model>.NombreProd</model></dt>
11     <dd><model>.PrecioProd</model></dd>
12     <dt><model>.CantidadProd</model></dt>
13     <dd><model>.CategoríaProd</model></dd>
14   </div>
15   <p><a href="#">Editar</a> | <a href="#">Regresar</a> | <a href="#">Index</a></p>
16 </div>
17 
```

The screenshot shows the 'Explorador de soluciones' (Solution Explorer) on the right, displaying the project structure for 'DesafíoDSE'. The 'Views' folder contains 'Clientes', 'Home', 'Pedidos', and 'Productos' subfolders, each with its own 'Create.cshtml', 'Delete.cshtml', 'Edit.cshtml', and 'Index.cshtml' files. The 'Controllers' folder also contains 'ClientesController.cs', 'HomeController.cs', 'PedidosController.cs', and 'ProductosController.cs'. The 'Properties' and 'Migrations' folders are also visible. The 'Edad.cshtml' file is shown open in the code editor, containing the provided HTML and C# code.



f. Products/Delete.cshtml

The screenshot shows the Microsoft Visual Studio IDE interface. The main window displays the code for the `Delete.cshtml` view, which is part of the `DesafioIDSE` project. The code is written in C# Razor syntax and includes logic to handle product deletion. The `Layout.cshtml` file is referenced at the top. The `Model` is `DesafioIDSE.Models.Productos`. The code includes sections for displaying product details and confirming deletion. The `ModelState` is checked for errors, and a `ValidateAntiForgeryToken` token is included in the form.

```
Layout.cshtml (CRUD.cs)
1 <model> DesafioIDSE.Models.Productos</model>
2
3 <h1>Viendo: <model>.NombreProd</model></h1>
4
5 <h2>Delete</h2>
6
7 <p>Are you sure you want to delete this?</p>
8
9 <form>
10    <input type="hidden" name="id" value="<model>.IdProd" />
11    <div class="diagonal">
12        <table border="1">
13            <tr>
14                <td><model>.NombreProd</td>
15                <td><model>.PrecioProd</td>
16                <td><model>.CantidadProd</td>
17            </tr>
18        </table>
19    </div>
20    <div>
21        <input type="submit" value="Delete" />
22    </div>
23 </form>
24
25 <script>(function () {
26     'use strict';
27     angular
28         .module('app.core')
29         .factory('dataservice', dataservice);
30
31     dataservice.$inject = ['$http'];
32     /* @ngInject */
33     function dataservice($http) {
34         var serviceBase = 'api/';
35
36         var service = {
37             getTodos: getTodos,
38             getOne: getOne,
39             addOne: addOne,
40             deleteOne: deleteOne
41         };
42
43         return service;
44     }
45
46     /* @ngInject */
47     function getTodos() {
48         return $q.all();
49     }
50
51     /* @ngInject */
52     function getOne(id) {
53         var def = $q.defer();
54         $http.get(serviceBase + 'products/' + id)
55             .then(def.resolve, def.reject);
56         return def.promise;
57     }
58
59     /* @ngInject */
60     function addOne(producto) {
61         var def = $q.defer();
62         $http.post(serviceBase + 'products', producto)
63             .then(def.resolve, def.reject);
64         return def.promise;
65     }
66
67     /* @ngInject */
68     function deleteOne(id) {
69         var def = $q.defer();
70         $http.delete(serviceBase + 'products/' + id)
71             .then(def.resolve, def.reject);
72         return def.promise;
73     }
74
75     /* @ngInject */
76     function getTodosPromise() {
77         return $q.all();
78     }
79
80     /* @ngInject */
81     function getOnePromise(id) {
82         var def = $q.defer();
83         $http.get(serviceBase + 'products/' + id)
84             .then(def.resolve, def.reject);
85         return def.promise;
86     }
87
88     /* @ngInject */
89     function addOnePromise(producto) {
90         var def = $q.defer();
91         $http.post(serviceBase + 'products', producto)
92             .then(def.resolve, def.reject);
93         return def.promise;
94     }
95
96     /* @ngInject */
97     function deleteOnePromise(id) {
98         var def = $q.defer();
99         $http.delete(serviceBase + 'products/' + id)
100            .then(def.resolve, def.reject);
101        return def.promise;
102    }
103
104    return service;
105 }
106 )();
```

The screenshot shows the Microsoft Visual Studio interface with the following components:

- Code Editor:** Displays the `Delete.cshtml` file content, which is a partial view for the `CRUD.cs` controller. It contains HTML and C# code for displaying product details and a confirmation message before deletion.
- Solution Explorer:** Shows the project structure for "DesafíoDSE". It includes controllers like `ClientesController.cs`, `HomeController.cs`, `PedidosController.cs`, and `ProductosController.cs`, and views such as `Index.cshtml` and `Details.cshtml`.
- Error List:** Shows one warning: "El elemento meta no se puede anidar en el elemento 'html'." (The meta element cannot be nested in the html element).
- Browser Preview:** A separate window shows the "Delete" page for products. The URL is `localhost:44366/Productos/Delete/3`. The page displays a product's details (Nombre: Gas, Precio: 20.00, Stock: 6) and two buttons: "Borrar" (Delete) and "Regresar" (Back).

4. Tabla Clientes

a. Pedidos.cs

The screenshot shows the Microsoft Visual Studio interface. The code editor displays the `Pedidos.cs` file, which contains the definition of a `Pedidos` class with properties like `Id`, `NombreC1`, `CantidadPedida`, and `PrecioProd`. The Solution Explorer on the right shows the project structure for `DesafioIDSE`, including controllers like `ClientesController.cs` and `PedidosController.cs`, and views like `Index.cshtml` and `Create.cshtml`.

```

using System;
using System.Collections.Generic;
using System.Linq;
using System.Web;
using System.ComponentModel.DataAnnotations;
using System.ComponentModel.DataAnnotations.Schema;

namespace DesafioIDSE.Models
{
    public class Pedidos
    {
        [Key]
        public int Id { get; set; }

        public string NombreC1 { get; set; }

        [Required, Range(0, int.MaxValue, ErrorMessage = "Por favor ingrese una cantidad valida")]
        public int CantidadPedida { get; set; }

        public string PrecioProd { get; set; }

        public virtual Clientes Clientes { get; set; }

        public virtual Productos Productos { get; set; }
    }
}

```

1. Pedidos/Index.cshtml

The screenshot shows the Microsoft Visual Studio interface. The code editor displays the `Index.cshtml` file, which contains an `ActionLink` for creating new items and a `Table` for displaying existing items. The table has columns for `NombreC1`, `NombreProd`, `CantidadPedida`, `PrecioProd`, and `Actions`. The Solution Explorer on the right shows the project structure for `DesafioIDSE`, including controllers like `ClientesController.cs` and `PedidosController.cs`, and views like `Index.cshtml` and `Create.cshtml`.

```

@model IEnumerable<DesafioIDSE.Models.Pedidos>
@{
    ViewBag.Title = "Index";
}



## Pedidos



Create New |
    Index



|  | NombreC1 | NombreProd | CantidadPedida | PrecioProd | Actions |
|--|----------|------------|----------------|------------|---------|
|  |          |            |                |            |         |


```

```

@model IEnumerable<DesafioDSE.Models.Pedido>


|                            | NombreCl | NombreProd | CantidadPedida | PrecioProd |
|----------------------------|----------|------------|----------------|------------|
| <a href="#">Create New</a> |          |            |                |            |
| <a href="#">Edit</a>       | NombreCl | NombreProd | CantidadPedida | PrecioProd |
| <a href="#">Details</a>    |          |            |                |            |
| <a href="#">Delete</a>     |          |            |                |            |


```

NombreCl	NombreProd	CantidadPedida	PrecioProd
----------	------------	----------------	------------

2. Pedidos/Create.cshtml

Gestión de Archivos y Directories

SIO
Práctica 13

```
1 <%@ Page Title="Create" %>
2 
3 <h2>Create</h2>
4 
5 
6 
7 
8 <using> (Html.BeginForm())
9 {
10     <div> AntiforgeryToken()
11 
12     <div> Form-horizontal
13         <div> ModelNameProd, htmlAttributes: new { @class = "control-label col-md-2" }
14             <label> NombreProd
15             <input type="text" />
16             <div> ValidationSummary(model => model.NombreProd, new { @class = "text-danger" })
17             <div> EditorFor(model => model.NombreProd, new { htmlAttributes = new { @class = "form-control" } })
18             <div> ValidationMessageFor(model => model.NombreProd, "", new { @class = "text-danger" })
19         </div>
20 
21         <div> ModelNameProd, htmlAttributes: new { @class = "control-label col-md-2" }
22             <label> CantidadPedida
23             <input type="text" />
24             <div> EditorFor(model => model.CantidadPedida, new { htmlAttributes = new { @class = "form-control" } })
25             <div> ValidationMessageFor(model => model.CantidadPedida, "", new { @class = "text-danger" })
26         </div>
27 
28         <div> ModelNameProd, htmlAttributes: new { @class = "control-label col-md-2" }
29             <label> PrecioProd, htmlAttributes: new { @class = "control-label col-md-2" }
30             <div> ModelNameProd, htmlAttributes: new { @class = "control-label col-md-10" }
31                 <input type="text" />
32                 <div> EditorFor(model => model.PrecioProd, new { htmlAttributes = new { @class = "form-control" } })
33                 <div> ValidationMessageFor(model => model.PrecioProd, "", new { @class = "text-danger" })
34             </div>
35 
36         <div> ModelNameProd, htmlAttributes: new { @class = "control-label col-md-2" }
37             <div> ModelNameProd, htmlAttributes: new { @class = "control-label col-md-10" }
38                 <input type="text" />
39                 <div> EditorFor(model => model.ModelNameProd, new { htmlAttributes = new { @class = "form-control" } })
40                 <div> ValidationMessageFor(model => model.ModelNameProd, "", new { @class = "text-danger" })
41             </div>
42 
43         <div> ModelNameProd, htmlAttributes: new { @class = "control-label col-md-2" }
44             <div> ModelNameProd, htmlAttributes: new { @class = "control-label col-md-10" }
45                 <input type="text" />
46                 <div> EditorFor(model => model.ModelNameProd, new { htmlAttributes = new { @class = "form-control" } })
47                 <div> ValidationMessageFor(model => model.ModelNameProd, "", new { @class = "text-danger" })
48             </div>
49 
50         <div> ModelNameProd, htmlAttributes: new { @class = "control-label col-md-2" }
51             <div> ModelNameProd, htmlAttributes: new { @class = "control-label col-md-10" }
52                 <input type="text" />
53                 <div> EditorFor(model => model.ModelNameProd, new { htmlAttributes = new { @class = "form-control" } })
54                 <div> ValidationMessageFor(model => model.ModelNameProd, "", new { @class = "text-danger" })
55             </div>
56         </div>
57 
58         <div> ActionLink("Back to List", "Index")
59     </div>
60 
61     <div> Scripts
62         <script> Render("~/bundles/jqueryval") </script>
63     </div>
64 }
65 
```

Explorador de soluciones

- DesafíoSDE
 - Connected Services
 - Properties
 - Referencias
 - App_Data
 - App_Start
 - Areas
 - Content
 - Controllers
 - ClientesController.cs
 - HomeController.cs
 - PedidosController.cs
 - ProductosController.cs
 - ValuesController.cs
 - Models
 - Clients.cs
 - CRUD.cs
 - Pedidos.cs
 - Productos.cs
 - Scripts
 - Views
 - Clientes
 - Home
 - Pedidos
 - Create.cshtml
 - Delete.cshtml
 - Details.cshtml
 - Edit.cshtml
 - Index.cshtml
 - Layout.cshtml
 - ViewStart.cshtml
 - Productos
 - Pedidas
 - Migraciones
 - 2011112159143_Migracion1.cs
 - Configuration.cs
 - Modelos
 - Models
 - Clients.cs
 - CRUD.cs
 - Pedidos.cs
 - Productos.cs
 - Scripts
 - Views
 - Pedidas
 - Create.cshtml
 - Delete.cshtml
 - Details.cshtml
 - Edit.cshtml
 - Index.cshtml
 - Layout.cshtml
 - ViewStart.cshtml
 - Productos
 - Pedidas
 - Shared
 - Layout.cshtml
 - Error.cshtml
 - ViewStart.cshtml
 - Web.config
 - Global.asax
 - Web.config
 - WebReferences.config
 - DesafíoSDE.Tests

Explorador de soluciones Cambios de GIT: CD16156-FM100...

The screenshot shows a web application window titled "Desafío 3 DSE". The URL in the address bar is "localhost:44366/Pedidos/Create". The page header includes links for "Menu", "Create", "C", "localhost:44366/Pedidos/Create", "Clientes", "Productos", "Pedidos", and "API". Below the header, the main content area has a title "Create" under "Pedidos". There are four input fields: "NombreCl" with value "Fernando", "NombreProd" with value "Gas", "CantidadPedida" with value "2", and "PrecioProd" with value "20.00". At the bottom of the form is a "Create" button. A link "Back to List" is located just above the footer. The footer contains the text "© 2021 - Mi aplicación ASP.NET".

NombreCl: Fernando

NombreProd: Gas

CantidadPedida: 2

PrecioProd: 20.00

Create

Back to List

© 2021 - Mi aplicación ASP.NET