Tipo : Guía de Enunciado
Capítulo : Aplicando técnicas en una aplicación ASP.NET MVC
Duración : 180 minutos

I. OBJETIVO

Implementa un Modal Popup para la carga de vistas parciales.

II. REQUISITOS

Los siguientes elementos de software son necesarios para la realización del laboratorio:

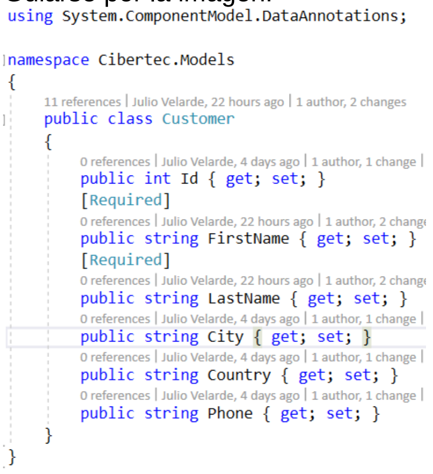
- Windows 10 (como mínimo Windows 8)
- Visual Studio 2017 (como mínimo Visual Studio 2015)

III. EJECUCIÓN DEL LABORATORIO

1. Abrir la solución del módulo anterior.

2. En el proyecto “**Cibertec.Models**” agregamos la referencia al namespace “**System.ComponentModel.DataAnnotations**”

3. En el mismo proyecto agregamos este dataanotation en la clase Customer: “**[Required]**” Guiarse por la imagen:



```
using System.ComponentModel.DataAnnotations;

namespace Cibertec.Models
{
    11 references | Julio Velarde, 22 hours ago | 1 author, 2 changes
    public class Customer
    {
        0 references | Julio Velarde, 4 days ago | 1 author, 1 change |
        public int Id { get; set; }
        [Required]
        0 references | Julio Velarde, 22 hours ago | 1 author, 2 change
        public string FirstName { get; set; }
        [Required]
        0 references | Julio Velarde, 22 hours ago | 1 author, 2 change
        public string LastName { get; set; }
        0 references | Julio Velarde, 4 days ago | 1 author, 1 change |
        public string City { get; set; }
        0 references | Julio Velarde, 4 days ago | 1 author, 1 change |
        public string Country { get; set; }
        0 references | Julio Velarde, 4 days ago | 1 author, 1 change |
        public string Phone { get; set; }
    }
}
```

4. Instala el siguiente paquete al proyecto “**Cibertec.Mvc**”:
   - ➢ Install-Package Microsoft.jQuery.Unobtrusive.Ajax

5. Procedemos a modificar el controlador Customer, esto con la finalidad de responder con vistar parciales, con el siguiente código:

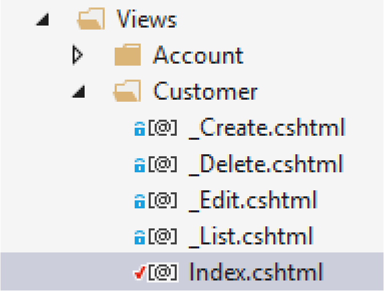
```
using Cibertec.Models;
using Cibertec.UnitOfWork;
using System.Web.Mvc;
using log4net;
using Cibertec.Mvc.ActionFilters;

namespace Cibertec.Mvc.Controllers
{
    [ErrorActionFilter]
    public class CustomerController : BaseController
    {
        public CustomerController(ILog log, IUnitOfWork unit) : base(log,
unit)
        {
        }
        public ActionResult Error()
        {
            throw new System.Exception("Test error to validate Action
Filter");
        }

        public ActionResult Index()
        {
            _log.Info("Execution of Customer Controller OK");
            return View(_unit.Customers.GetList());
        }

        public PartialViewResult Create()
        {
            return PartialView("_Create", new Customer());
        }

        [HttpPost]
        public ActionResult Create(Customer customer)
        {
            if (!ModelState.IsValid) return PartialView("_Create",
customer);
            _unit.Customers.Insert(customer);
            return RedirectToAction("Index");
        }

        public ActionResult Edit(int id)
        {
            return PartialView("_Edit", _unit.Customers.GetById(id));
        }

        [HttpPost]
        public ActionResult Edit(Customer customer)
        {
            if (!ModelState.IsValid) return PartialView("_Edit",
customer);
            _unit.Customers.Update(customer);
            return RedirectToAction("Index");
        }

        public ActionResult Delete(int id)
        {
            return PartialView("_Delete", _unit.Customers.GetById(id));
        }

        [HttpPost]
        public ActionResult Delete(Customer customer)
        {
            if (_unit.Customers.Delete(customer)) return
RedirectToAction("Index");
            return PartialView("_Delete", customer);
        }
    }
}
```

6. Iniciamos a refactorizar las vistas de la carpeta **Customer**:

(1) Iniciamos con la vista “**Index**”:

```
@{
    ViewBag.Title = "Index";
}

<h2>Index</h2>

<p>
    <a href="#" onclick="cibertec.getModal('@Url.Action("Create")');
return false;" data-toggle="modal" data-target="#modal-container"
class="btn btn-success btn-sm">
        <span class="glyphicon glyphicon-plus"></span> New
    </a>
</p>
@Html.Partial("_List")

@Modal.GetModal("Customer")
```

(2) Antes de continuar con el refactoring de las vistas Create, Edit y Delete las renombramos de la siguiente manera: **_Create, _Edit y _Delete**:



```
Views
    Account
    Customer
        _Create.cshtml
        _Delete.cshtml
        _Edit.cshtml
        _List.cshtml
        Index.cshtml
```

(3) Editamos la vista “**_Create**” y la dejamos con el siguiente código:

```
@model Cibertec.Models.Customer
<h2>Create</h2>
@using (Ajax.BeginForm("Create", new { Controller = "Customer" },
new AjaxOptions
{
   HttpMethod = "POST",
   InsertionMode = InsertionMode.Replace,
   UpdateTargetId = "modal-body",
   OnSuccess = "customer.success();"
},
htmlAttributes: new { id = "createForm" }))
{
    @Html.AntiForgeryToken()
    <div class="form-horizontal">
        @Html.ValidationSummary(true, "", new { @class = "text-danger"
})

        <div class="form-group">
            @Html.LabelFor(model => model.FirstName, htmlAttributes:
new { @class = "control-label col-md-2" })
            <div class="col-md-10">
                @Html.EditorFor(model => model.FirstName, new {
htmlAttributes = new { @class = "form-control" } })
                @Html.ValidationMessageFor(model => model.FirstName,
"", new { @class = "text-danger" })
            </div>
        </div>

        <div class="form-group">
```

```
            @Html.LabelFor(model => model.LastName, htmlAttributes: new
{ @class = "control-label col-md-2" })
            <div class="col-md-10">
                @Html.EditorFor(model => model.LastName, new {
htmlAttributes = new { @class = "form-control" } })
                @Html.ValidationMessageFor(model => model.LastName, "",
new { @class = "text-danger" })
            </div>
        </div>

        <div class="form-group">
            @Html.LabelFor(model => model.City, htmlAttributes: new {
@class = "control-label col-md-2" })
            <div class="col-md-10">
                @Html.EditorFor(model => model.City, new {
htmlAttributes = new { @class = "form-control" } })
                @Html.ValidationMessageFor(model => model.City, "", new
{ @class = "text-danger" })
            </div>
        </div>

        <div class="form-group">
            @Html.LabelFor(model => model.Country, htmlAttributes: new
{ @class = "control-label col-md-2" })
            <div class="col-md-10">
                @Html.EditorFor(model => model.Country, new {
htmlAttributes = new { @class = "form-control" } })
                @Html.ValidationMessageFor(model => model.Country, "",
new { @class = "text-danger" })
            </div>
        </div>

        <div class="form-group">
            @Html.LabelFor(model => model.Phone, htmlAttributes: new {
@class = "control-label col-md-2" })
            <div class="col-md-10">
                @Html.EditorFor(model => model.Phone, new {
htmlAttributes = new { @class = "form-control" } })
                @Html.ValidationMessageFor(model => model.Phone, "",
new { @class = "text-danger" })
            </div>
        </div>

        <div class="form-group">
            <div class="col-md-offset-2 col-md-10">
                <input type="submit" value="Create" class="btn btn-
default" />
            </div>
        </div>
    </div>
}

<script type="text/javascript">
    $(function () {
        $.validator.unobtrusive.parse("#createForm");
    });
</script>
```

(4) Editamos la vista "**_Edit**" y la dejamos con el siguiente código:

```
@model Cibertec.Models.Customer
<h2>Edit</h2>
@using (Ajax.BeginForm("Edit", new { Controller = "Customer" },
                                new AjaxOptions
                                {
                                    HttpMethod = "POST",
                                    InsertionMode =
InsertionMode.Replace,
                                    UpdateTargetId = "modal-body",
                                    OnSuccess = "customer.success();"
                                }, htmlAttributes: new { id =
"editForm" }))
{
    @Html.AntiForgeryToken()

    <div class="form-horizontal">
        <hr />
        @Html.ValidationSummary(true, "", new { @class = "text-danger"
})
        @Html.HiddenFor(model => model.Id)

        <div class="form-group">
            @Html.LabelFor(model => model.FirstName, htmlAttributes:
new { @class = "control-label col-md-2" })
            <div class="col-md-10">
                @Html.EditorFor(model => model.FirstName, new {
htmlAttributes = new { @class = "form-control" } })
                @Html.ValidationMessageFor(model => model.FirstName,
"", new { @class = "text-danger" })
            </div>
        </div>

        <div class="form-group">
            @Html.LabelFor(model => model.LastName, htmlAttributes: new
{ @class = "control-label col-md-2" })
            <div class="col-md-10">
                @Html.EditorFor(model => model.LastName, new {
htmlAttributes = new { @class = "form-control" } })
                @Html.ValidationMessageFor(model => model.LastName, "",
new { @class = "text-danger" })
            </div>
        </div>

        <div class="form-group">
            @Html.LabelFor(model => model.City, htmlAttributes: new {
@class = "control-label col-md-2" })
            <div class="col-md-10">
                @Html.EditorFor(model => model.City, new {
htmlAttributes = new { @class = "form-control" } })
                @Html.ValidationMessageFor(model => model.City, "", new
{ @class = "text-danger" })
            </div>
        </div>

        <div class="form-group">
            @Html.LabelFor(model => model.Country, htmlAttributes: new
{ @class = "control-label col-md-2" })
            <div class="col-md-10">
                @Html.EditorFor(model => model.Country, new {
htmlAttributes = new { @class = "form-control" } })
```

```
                @Html.ValidationMessageFor(model => model.Country, "", 
new { @class = "text-danger" })
            </div>
        </div>

        <div class="form-group">
            @Html.LabelFor(model => model.Phone, htmlAttributes: new { 
@class = "control-label col-md-2" })
            <div class="col-md-10">
                @Html.EditorFor(model => model.Phone, new { 
htmlAttributes = new { @class = "form-control" } })
                @Html.ValidationMessageFor(model => model.Phone, "", 
new { @class = "text-danger" })
            </div>
        </div>

        <div class="form-group">
            <div class="col-md-offset-2 col-md-10">
                <input type="submit" value="Save" class="btn btn-
default" />
            </div>
        </div>
    </div>
}
<script type="text/javascript">
    $(function () {
        $.validator.unobtrusive.parse("#editForm");
    });
</script>
```

(5) Editamos la vista "**_Delete**" y la dejamos con el siguiente código:

```
@model Cibertec.Models.Customer

<h2>Delete</h2>
<h3>Are you sure you want to delete this?</h3>
<div>
    <hr />
    <dl class="dl-horizontal">
        <dt>
            @Html.DisplayNameFor(model => model.FirstName)
        </dt>

        <dd>
            @Html.DisplayFor(model => model.FirstName)
        </dd>

        <dt>
            @Html.DisplayNameFor(model => model.LastName)
        </dt>

        <dd>
            @Html.DisplayFor(model => model.LastName)
        </dd>

        <dt>
            @Html.DisplayNameFor(model => model.City)
        </dt>

        <dd>
            @Html.DisplayFor(model => model.City)
        </dd>
        <dt>
            @Html.DisplayNameFor(model => model.Country)
        </dt>

        <dd>
            @Html.DisplayFor(model => model.Country)
        </dd>

        <dt>
            @Html.DisplayNameFor(model => model.Phone)
        </dt>

        <dd>
            @Html.DisplayFor(model => model.Phone)
        </dd>

    </dl>

    @using (Ajax.BeginForm("Delete", new { Controller = "Customer" },
                                            new AjaxOptions
                                            {
                                                HttpMethod = "POST",
                                                InsertionMode =
InsertionMode.Replace,
                                                UpdateTargetId =
"modal-body",
                                                OnSuccess =
"customer.success();"
                                            }, htmlAttributes: new { id
= "deleteForm" }))
    {

        @Html.AntiForgeryToken()

        <div class="form-actions no-color">
            <input type="submit" value="Delete" class="btn btn-default"
/>
        </div>
    }
</div>

<script type="text/javascript">
    $(function () {
        $.validator.unobtrusive.parse("#deleteForm");
    });
</script>
```

(6) Creamos la vista parcial “**_List**” con el siguiente código:

```
@model IEnumerable<Cibertec.Models.Customer>

<div class="table-responsive">
    <table class="table">
        <tr>
            <th>
                @Html.DisplayNameFor(model => model.FirstName)
            </th>
            <th>
                @Html.DisplayNameFor(model => model.LastName)
            </th>
            <th>
                @Html.DisplayNameFor(model => model.City)
            </th>
            <th>
                @Html.DisplayNameFor(model => model.Country)
            </th>
            <th>
                @Html.DisplayNameFor(model => model.Phone)
            </th>
            <th></th>
        </tr>

        @foreach (var item in Model)
        {
            <tr>
                <td>
                    @Html.DisplayFor(modelItem => item.FirstName)
                </td>
                <td>
                    @Html.DisplayFor(modelItem => item.LastName)
                </td>
                <td>
                    @Html.DisplayFor(modelItem => item.City)
                </td>
                <td>
                    @Html.DisplayFor(modelItem => item.Country)
                </td>
                <td>
                    @Html.DisplayFor(modelItem => item.Phone)
                </td>
                <td>
                    <a href="#"
onclick="cibertec.getModal('@Url.Action("Edit",new { id=item.Id})');
return false;" data-toggle="modal" data-target="#modal-container"
class="btn btn-warning btn-sm">
                        <span class="glyphicon glyphicon-edit"></span>
Edit
                    </a>
                    <a href="#"
onclick="cibertec.getModal('@Url.Action("Delete",new { id=item.Id})');
return false;" data-toggle="modal" data-target="#modal-container"
class="btn btn-danger btn-sm">
                        <span class="glyphicon glyphicon-edit"></span>
Delete
                    </a>
                </td>
            </tr>
        }
    </table>
</div>
```

(7) Editamos la vista “**_Layout**” que se encuentra en la ruta “**Views\Shared**” con el siguiente código:

```
<!DOCTYPE html>
<html>
<head>
    <meta charset="utf-8" />
    <meta name="viewport" content="width=device-width, initial-
scale=1.0">
    <title>@ViewBag.Title - My ASP.NET Application</title>
    @Styles.Render("~/Content/css")
    @Scripts.Render("~/bundles/jquery")
    @Scripts.Render("~/bundles/jqueryval")
    <script src="~/Scripts/jquery.unobtrusive-ajax.js"></script>
    @Scripts.Render("~/bundles/modernizr")
</head>
<body>
    <div class="navbar navbar-inverse navbar-fixed-top">
        <div class="container">
            <div class="navbar-header">
                <button type="button" class="navbar-toggle" data-
toggle="collapse" data-target=".navbar-collapse">
                    <span class="icon-bar"></span>
                    <span class="icon-bar"></span>
                    <span class="icon-bar"></span>
                </button>
                @Html.ActionLink("Cibertec Web", "Index", "Home", new {
area = "" }, new { @class = "navbar-brand" })
            </div>
            <div class="navbar-collapse collapse">
                <ul class="nav navbar-nav">
                    <li>@Html.ActionLink("Home", "Index", "Home")</li>
                    <li>@Html.ActionLink("Customer", "Index",
"Customer")</li>
                </ul>
                @Html.Partial("_LoginPartial")
            </div>
        </div>
    </div>
    <div class="container body-content">
        @RenderBody()
        <hr />
        <footer>
            <p>&copy; @DateTime.Now.Year - Cibertec Web Application</p>
        </footer>
    </div>

    @Scripts.Render("~/bundles/bootstrap")
    @RenderSection("scripts", required: false)
</body>
</html>
```

7. En la raíz del proyecto “**Cibertec.Mvc**” creamos la carpeta “**App**” en la que ubicaremos nuestros ficheros Javascript con el código necesario para usar las vistas parciales en un modal popup.

8. Crear en la carpeta App un archivo nuevo con el nombre “**cibertec.js**” y agrega el siguiente código:

```
(function (cibertec) {

    cibertec.getModal = getModalContent;
    cibertec.closeModal = closeModal;

    return cibertec;

    function getModalContent(url) {
        $.get(url, function (data) {
            $('.modal-body').html(data);
        })
    }

    function closeModal() {
        $("button[data-dismiss='modal']").click();
        $('.modal-body').html("");
    }

})(window.cibertec = window.cibertec || {});
```

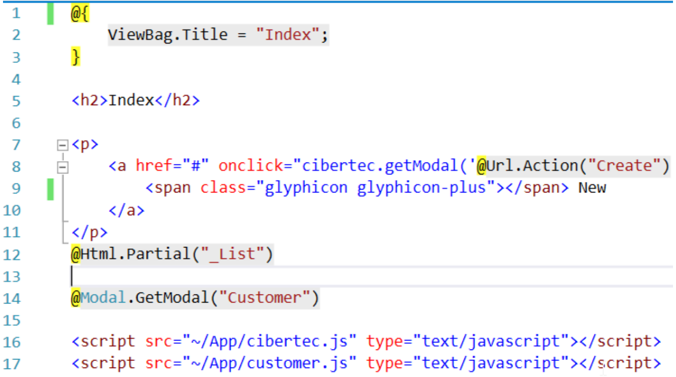
9. Crea un archivo nuevo javascript con el nombre “**customer.js**” y agrega el siguiente código:

```
(function (customer) {
    customer.success = successReload;

    return customer;

    function successReload() {
        cibertec.closeModal();
    }

})(window.customer = window.customer || {});
```

10. En la vista “**Index**” de la carpeta “**Views\Customer**” agrega las llamadas a los 2 ficheros javascript creados:

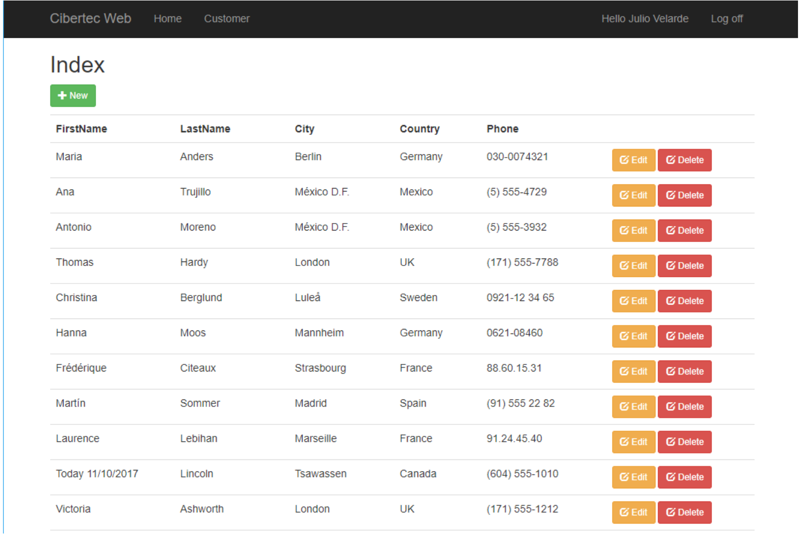
```
<script src="~/App/cibertec.js" type="text/javascript"></script>
<script src="~/App/customer.js" type="text/javascript"></script>
```

Guiarse de la imagen:



```
@{
    ViewBag.Title = "Index";
}

<h2>Index</h2>

<p>
    <a href="#" onclick="cibertec.getModal('@Url.Action("Create")
        <span class="glyphicon glyphicon-plus"></span> New
    </a>
</p>
@Html.Partial("_List")

@Modal.GetModal("Customer")

<script src="~/App/cibertec.js" type="text/javascript"></script>
<script src="~/App/customer.js" type="text/javascript"></script>
```

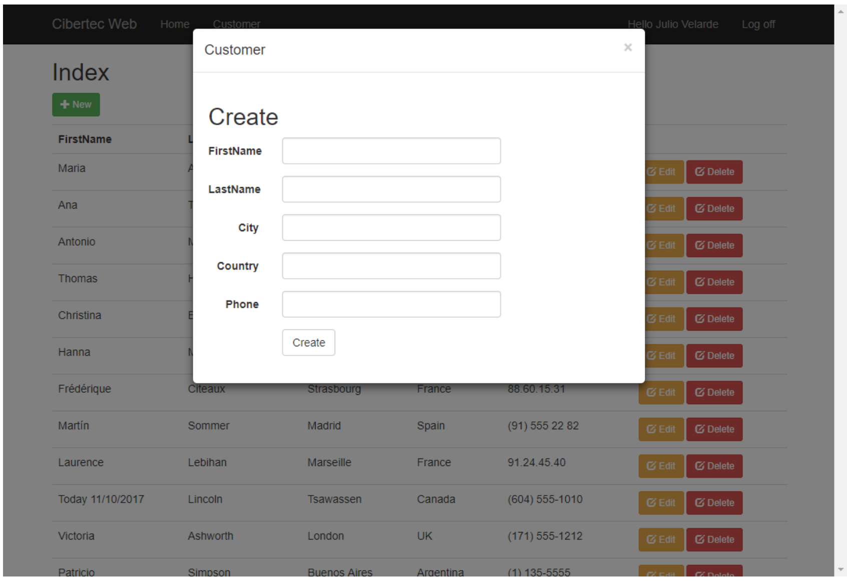
11.El resultado será como en las siguientes imágenes:



Index

FirstName	LastName	City	Country	Phone	
Maria	Anders	Berlin	Germany	030-0074321	Edit Delete
Ana	Trujillo	México D.F.	Mexico	(5) 555-4729	Edit Delete
Antonio	Moreno	México D.F.	Mexico	(5) 555-3932	Edit Delete
Thomas	Hardy	London	UK	(171) 555-7788	Edit Delete
Christina	Berglund	Luleå	Sweden	0921-12 34 65	Edit Delete
Hanna	Moos	Mannheim	Germany	0621-08460	Edit Delete
Frédérique	Citeaux	Strasbourg	France	88.60.15.31	Edit Delete
Martín	Sommer	Madrid	Spain	(91) 555 22 82	Edit Delete
Laurence	Lebihan	Marseille	France	91.24.45.40	Edit Delete
Today 11/10/2017	Lincoln	Tsawassen	Canada	(604) 555-1010	Edit Delete
Victoria	Ashworth	London	UK	(171) 555-1212	Edit Delete

Al hacer clic en “**New**”



Create

FirstName

LastName

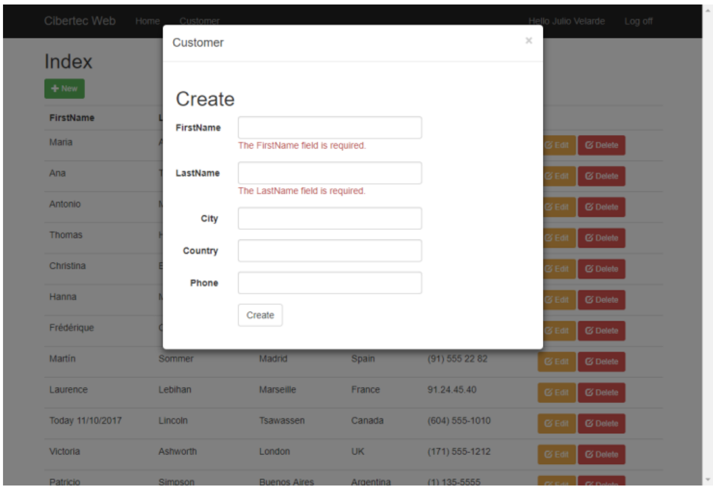
City

Country

Phone

FirstName					
Frédérique	Citeaux	Strasbourg	France	88.60.15.31	Edit Delete
Martín	Sommer	Madrid	Spain	(91) 555 22 82	Edit Delete
Laurence	Lebihan	Marseille	France	91.24.45.40	Edit Delete
Today 11/10/2017	Lincoln	Tsawassen	Canada	(604) 555-1010	Edit Delete
Victoria	Ashworth	London	UK	(171) 555-1212	Edit Delete
Patricio	Simpson	Buenos Aires	Argentina	(1) 135-5555	

Validación de errores:



Create

FirstName

The FirstName field is required.

LastName

The LastName field is required.

City

Country

Phone

FirstName					
Martín	Sommer	Madrid	Spain	(91) 555 22 82	Edit Delete
Laurence	Lebihan	Marseille	France	91.24.45.40	Edit Delete
Today 11/10/2017	Lincoln	Tsawassen	Canada	(604) 555-1010	Edit Delete
Victoria	Ashworth	London	UK	(171) 555-1212	Edit Delete
Patricio	Simpson	Buenos Aires	Argentina	(1) 135-5555	

12.Hacer lo mismo con las demás opciones.

IV. EVALUACIÓN

1. ¿Por qué para vistas parciales se deben de usar “_” el guion bajo?

   Bajo el lenguaje Razor, el guion bajo permite evitar mostrar Web Pages cuando son solicitadas directamente como parte de un request.