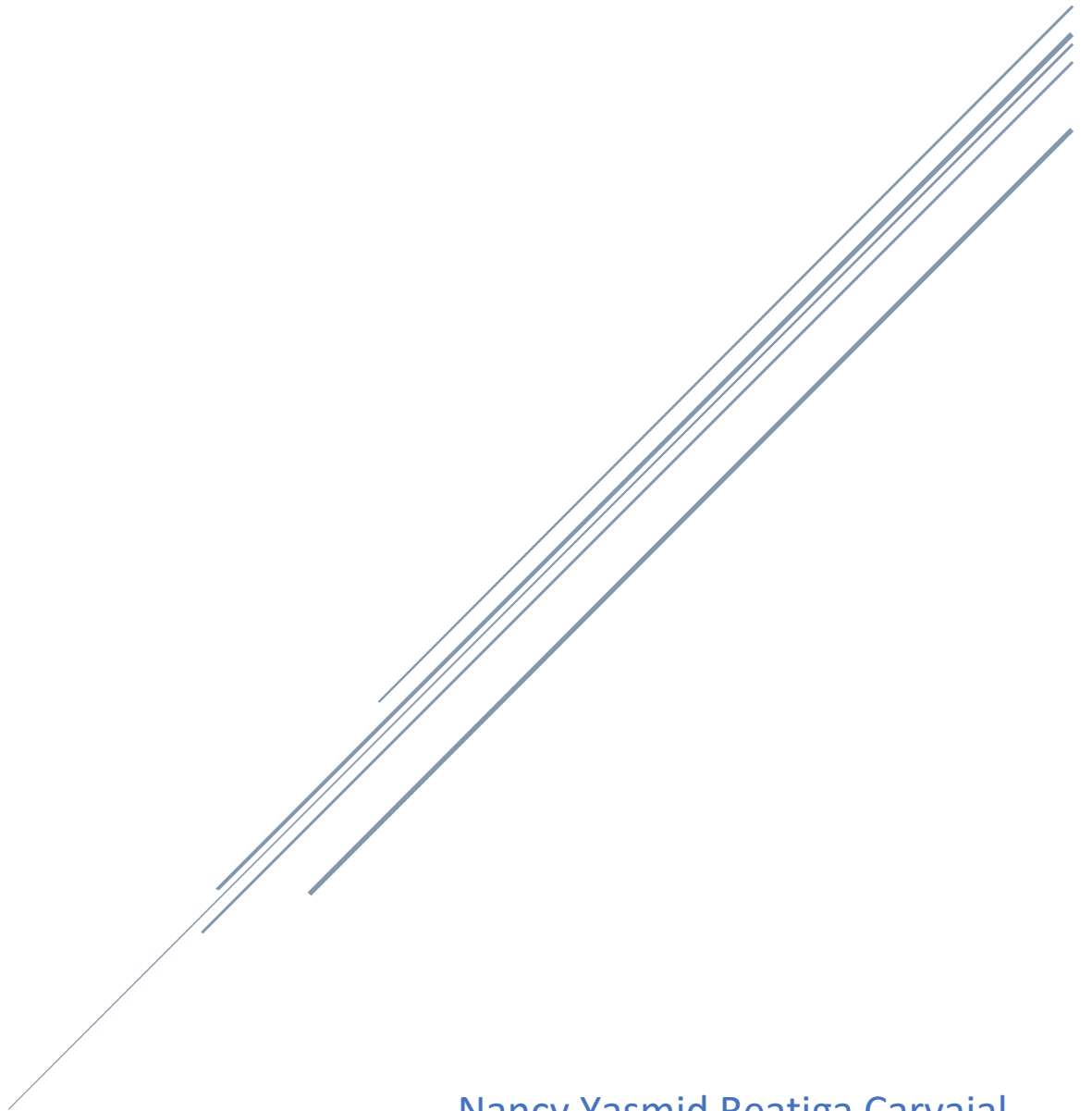
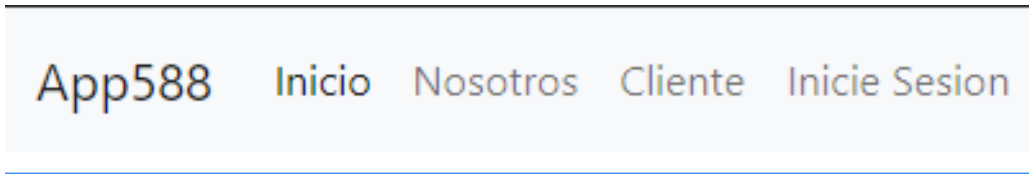


MOCKUPS



Nancy Yasmid Reatiga Carvajal
Análisis Desarrollo Sistemas de Información
Ficha: 2184588
Bogotá D.C

`<header>` es a la cabecera de la aplicación la cual me permite navegar en las diferentes vistas `</header>`



Para crear los ítems de navegación que se muestran debe ir dentro de un:

```
<div class="collapse navbar-collapse"
id="navbarNav">
<ul class="navbar-nav">
```

```
<li class="nav-item">
<a class="nav-link active" aria-current="page"
asp-controller="Home" asp-
action="Home">Inicio</a>
</li>
```

VISTA DE INICIO

Para el título de la vista de inicio se debe colocar

```
<h2 class="text-center m-2">My App Asp Net Core</h2>
```



Para agrupar el contenido debe estar dentro de un div que es el contenedor principal: `<div class="col-12 bg-primary d-flex flex-column align-items-center flex-md-row flex-lg-row justify-content-md-center justify-content-lg-center align-items-md-start align-items-lg-start">`

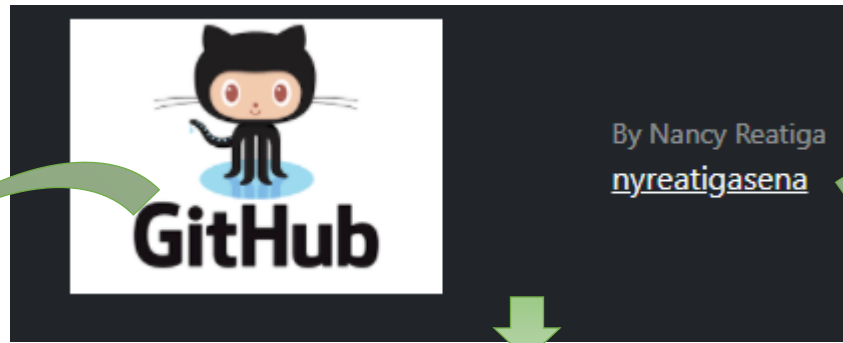
Para colocar la imagen debe de estar con un **div class** para moldear el contenedor y **img class** para traer la imagen de la carpeta.

```
<div class="col-12 col-md-6 col-lg-3 bg-white p-5
border rounded m-2 d-flex align-items-center justify-
content-start">
</div>
```

Para colocar el texto debe de estar con un **div class** para moldear el contenedor y **p** para colocar un párrafo.

```
<div class="col-12 col-md-6 col-lg-8 bg-white p-3
border rounded m-2">
<p>El texto que se quiera implementar por
párrafo</p>
<p>El texto que se quiera implementar por
párrafo</p>
```

<footer> Es el pie de página de la aplicación la cual solo permite visualizar la información **</footer>**



Para agrupar el contenido debe estar dentro de un div que es el contenedor principal:
`<footer class="text-white-50 bg-dark p-3 d-flex justify-content-center">`
`<div class="d-flex align-items-center">`

Para colocar la imagen debe de estar con un **div class** para moldear el contenedor y **img class** para traer la imagen de la carpeta.

```
<div></div>
<div class="d-flex flex-column">
```

Para colocar el texto debe de están con un **small** y en un **a href** que es el enlace para un correo. Todo va dentro de un div.

```
<div class="d-flex flex-column">
<small>By Nancy Reatiga</small>
<a href="#" class="linklight">nyreatigasena </a>
</div>
```

VISTA DE NOSOTROS



Para el titulo se coloca `<h2 class="text-center m-2">`, para los textos se coloca la `<p>` en el footer estaría el `<div></div>` para el texto sería `<small>texto</small>` ``, importante no olvidar los contenedores de cada parámetro

VISTA DE CLIENTES

Esta vista nos muestra el título, el footer, adicionalmente muestra un listado de clientes con sus datos básicos que se relacionan un base de datos, también tiene una paginación

My App Asp Net Core

Codigo	Nombre	Correo
1010	Julián Pedroza	jpedroza@misena.edu.co
1111	sebastian linares	sebas@gmail.com
1212	lorena rojas	lore@yahoo.es
2000	Lyzeth Muñoz	lyz@yahoo.com
2002	Cristian Castaño orjuela	cris@gmai.com
2003	Ricardo Montes	richar@hotmail.com

1 2 3 4 >

Para crear la paginación se debe enlazar la base de datos por consola a punta de comandos o por paquetes nuget

```
public IActionResult Index (int? page)
var bd = new netsenaContext ();
//Lista y pagina clientes
var pagenumber = page?? 1;
int pagesize = 6;
var lisclientes = bd. Clientes.ToPagedList(pagenumber,
pagesize)
return View(lisclientes);
```

Para visualizar la información presente se debe realizar la relación de la base de datos con el entity framework:

```
public partial class netsenaContext: DbContext
public netsenaContext ()
public netsenaContext (DbContextOptions<netsenaContext>
options)
base(options)
public virtual DbSet<Clientes> Clientes {get; set;}
```

```
protected override void OnModelCreating (ModelBuilder
modelBuilder)
modelBuilder.Entity<Clientes>(entity =>
entity.HasKey(e => e. Codigo)
. HasName("PRIMARY");
entity. ToTable("clientes");
entity. Property (e => e. Codigo). HasColumnType ("int (4)");
entity. Property (e => e. Correo)
. IsRequired ());
```

La imagen representa la base de datos que está relacionada con la app, muestra de tal manera las columnas con la información requerida.

				Codigo	Nombre	Correo	Rol
<input type="checkbox"/>	Editar	Copiar	Borrar	1010	Julían Pedroza	jpedroza@misena.edu.co	sinRol
<input type="checkbox"/>	Editar	Copiar	Borrar	1111	sebastian linaires	sebas@gmail.com	sinRol
<input type="checkbox"/>	Editar	Copiar	Borrar	1212	lorena rojas	lore@yahoo.es	sinRol
<input type="checkbox"/>	Editar	Copiar	Borrar	2000	Lyzeth Muñoz	lyz@yahoo.com	sinRol
<input type="checkbox"/>	Editar	Copiar	Borrar	2002	Cristian Castaño orjuela	cris@gmail.com	sinRol
<input type="checkbox"/>	Editar	Copiar	Borrar	2003	Ricardo Montes	richar@hotmail.com	sinRol
<input type="checkbox"/>	Editar	Copiar	Borrar	2005	Dora Salgado	dorita@yahoo.com	sinRol
<input type="checkbox"/>	Editar	Copiar	Borrar	2006	Carlo Torres	car@yahoo.com	sinRol
<input type="checkbox"/>	Editar	Copiar	Borrar	2007	Sabina Carvajal	carba@hotmail.co	sinRol
<input type="checkbox"/>	Editar	Copiar	Borrar	2009	Luis Gonzales	lgz@gmail.com	sinRol
<input type="checkbox"/>	Editar	Copiar	Borrar	2010	Gustavo Restrepo	gus@yahoo.co	sinRol
<input type="checkbox"/>	Editar	Copiar	Borrar	2011	Jesus Reatiga	rea@gmail.com	sinRol
<input type="checkbox"/>	Editar	Copiar	Borrar	2012	Alejandra Martinez	alem@hotmail.co	sinRol
<input type="checkbox"/>	Editar	Copiar	Borrar	2034	Carlos Tabera	tabera@gmail.com	sinRol
<input type="checkbox"/>	Editar	Copiar	Borrar	2131	laura gomez	lau@hotmail.om	sinRol
<input type="checkbox"/>	Editar	Copiar	Borrar	2134	Alejandro Cardona	ale@gmail.com	sinRol
<input type="checkbox"/>	Editar	Copiar	Borrar	3000	Pepita Gomez	pepit@hotmail.com	sinRol
<input type="checkbox"/>	Editar	Copiar	Borrar	3001	Jesus Ropero	fers@misena.edu	sinRol
<input type="checkbox"/>	Editar	Copiar	Borrar	3002	Aurelio Reatiga	aur@misena.edu	sinRol

LOGUIN

Esta vista nos permite loquearnos con usuario y contraseña para acceder a diferentes acciones dentro de la base de datos

INICIAR SESION

El usuario o contraseña no son validos

Usuario

Contraseña

Completar este campo

Iniciar Sesion

Para crear el botón que nos permite iniciar sesión

```
<button type="submit" class="btn btn-primary">
  Iniciar Sesion </button>
```

Para poder validar si la información es real y cumple con los parámetros establecidos, al ingresar los datos correctos me envía a la vista privada para










```
var returnUrl = ViewData["ReturnUrl"] as string;
var error = TempData["Error"] as string;
returnUrl = ViewData["ReturnUrl"] as string;
error = TempData["Error"] as string;
```

Para crear el formulario se debe tener la relación de la tabla de accesos con los roles creados dentro de la base de datos y así poder loquearse










Form

```
action="/Login?ReturnUrl=@WebUtility.UrlEncode(returnUrl)"
method="post">
<div class="mb-3">
<label class="form-label">Usuario</label>
<input type="text" name="username" class="form-control"
required></div>
<div class="mb-3">
<label class="form-label">Contraseña</label>
<input type="password" name="password" class="form-control"
```

Se crea una tabla en la base de datos con la asignación de los roles, también con la descripción de las actividades que pueden realizar

+ Opciones					Id	Rol	Descripción
<input type="checkbox"/>	 Editar	 Copiar	 Borrar		1	Vendedor	Encargados del detalle y etc
<input type="checkbox"/>	 Editar	 Copiar	 Borrar		2	Coordinador	Encargado del detalle y editar, etc.
<input type="checkbox"/>	 Editar	 Copiar	 Borrar		3	Administrador	Encargado de crear, borrar, editar y observar los ...

También se crea una tabla con la información de los usuarios que tienen acceso a ingresar a la plataforma para: modificar, crear, eliminar los clientes que se encuentran.

+ Opciones					ID	Nombre	Usuario	Contraseña	Rol
<input type="checkbox"/>	 Editar	 Copiar	 Borrar		1820	Jerundio Gonzales	pipeqk	0000	Vendedor
<input type="checkbox"/>	 Editar	 Copiar	 Borrar		3121	Claudia Lopez	tabata	1234	Administrador
<input type="checkbox"/>	 Editar	 Copiar	 Borrar		5432	Agustin Hangarita	nino	5678	Coordinador

PRIVADO

Cuando se loguean el **header** nos muestra dos funciones nuevas las cuales son: **Privado** que nos da una bienvenida con el nombre de la persona que ingreso y las funciones que tiene permitidas y la otra es **Cerrar Sesión** que nos permite finalizar actividades solo con oprimir

App588
Inicio
Nosotros
Cliente
Privado
Cerrar Sesión

Información Privada

Bienvenido Claudia Lopez

Lorem ipsum dolor sit amet consectetur adipisicing elit. Necessitatibus odit, repudiandae quidem tempore unde reprehenderit ipsum numquam dolorem iste officia esse dolores assumenda ipsa tenetur. Cum commodi blanditiis doloremque voluptas!

Lorem ipsum dolor sit amet consectetur adipisicing elit. Necessitatibus odit, repudiandae quidem tempore unde reprehenderit ipsum numquam dolorem iste officia esse dolores assumenda ipsa tenetur. Cum commodi blanditiis doloremque voluptas!



By Nancy Reatiga
[nyreatigasena](#)

Para que aparezca el nombre del usuario que se logueo se debe tener `<h2 class="text-center m-2">Bienvenido @User.Identity. Name</h2>`

DENEGADO

En esta vista si la persona no es el administrador nos va aparecer que no tienen permisos para realizar algunas actividades de igual manera nos muestra la información del usuario como el nombre, apellido y rol

Informacion Privada

Usuario Agustin Hangarita No tiene permiso para acceder

mostrar los datos del usuario logueado

1. username: nino
2. http://schemas.xmlsoap.org/ws/2005/05/identity/claims/nameidentifier: nino
3. http://schemas.xmlsoap.org/ws/2005/05/identity/claims/name: Agustin Hangarita
4. http://schemas.microsoft.com/ws/2008/06/identity/claims/role: Coordinador

Para que aparezca el nombre del usuario que se logueo se debe tener `<h2 class="text-center m-2">Usuario @User.Identity. Name No tiene permiso para acceder</h2>`

Para que aparezca el nombre del usuario que se logueo se debe tener

`<div class="col-12 col-md-6 col-lg-6 bg-white p-3 border rounded m-2">`

mostrar los datos del usuario logueado

``

`@foreach (var c in User.Claims)`

` @c. Type: @c. Value`

``

`</div>`

Esta es la vista del usuario que tiene acceso a todas las funciones de sistema

My App Asp Net Core

Agregar Cliente

Codigo	Nombre	Correo	
1010	Julián Pedroza	jpdroza@misena.edu.co	Detalle Editar Borrar
1111	sebastian linares	sebas@gmail.com	Detalle Editar Borrar
1212	lorena rojas	lore@yahoo.es	Detalle Editar Borrar
2000	Lyzeth Muñoz	lyz@yahoo.com	Detalle Editar Borrar
2002	Cristian Castaño orjuela	cris@gmai.com	Detalle Editar Borrar
2003	Ricardo Montes	richar@hotmail.com	Detalle Editar Borrar

1 2 3 4 >

Para que aparezca el botón de agregar clientes debe tener `<li class="nav-item">
Agregar Cliente
`

Para que aparezca las diferentes acciones se debe tener en cuenta lo siguiente

`@if (User.IsInRole("Administrador")) ||`

`User.IsInRole("Vendedor"))`

`@Html.ActionLink("Detalle", "Detalle", new {id = @mycliente. Codigo})`

`@if (User.IsInRole("Administrador")) ||`

`User.IsInRole("Coordinador"))`

`@Html.ActionLink("Editar", "Editar", new {id = @mycliente. Codigo})`

`@if (User.IsInRole("Administrador"))`

`@Html.ActionLink("Borrar", "Borrar", new {id = @mycliente.`

DETALLE

Esta es la vista que nos muestra el detalle del cliente también nos muestra un retorno a la vista anterior, que se llama ir a lista, también nos muestra dos acciones más para realizar cambios

Detalle De Un Cliente		
Codigo	Nombre	Correo
2000	Lyzeth Muñoz	lyz@yahoo.com

Ir a Lista

EditarBorrar

Para que aparezca el botón de **ir a lista** y las diferentes acciones se debe tener en cuenta lo siguiente

```
<ul class="nav nav-pills nav-fill">
<li class="nav-item">
<a class="nav-link active" aria-current="page" asp-
action="Index">Ir a Lista</a>
@if (User.IsInRole("Administrador") ||
User.IsInRole("Coordinador"))
@Html.ActionLink("Editar", "Editar", new {id =
@Model.Codigo})
@if (User.IsInRole("Administrador"))
@Html.ActionLink("Borrar", "Borrar", new {id =
@Model.Codigo})
</li>
</ul>Codigo}}
```

Para que aparezca la información del debe tener en cuenta lo siguiente

```
<table class="table">
<thead>
<tr>
<th scope="col">Codigo</th>
<th scope="col">Nombre</th>
<th scope="col">Correo</th>
</tr>
</thead>
<tbody>
<tr>
<th scope="row">@Model.Codigo</th>
<td>@Model.Nombre</td>
<td>@Model.Correo</td>
</tr>
</tbody>
</table>Codigo}}
```

Para realizar la actividad se debe validar en la base de datos también la validación del rol, para mostrar la información del cliente escogido

```
[Authorize (Roles = "Administrador, Vendedor")]
[HttpGet]
public IActionResult Detalle (int id)
var bd = new netsenaContext ();
var mycliente = bd. Clientes.Find(id);
return View(mycliente);
```


EDITAR

Esta es la vista que nos muestra los campos que se pueden editar, teniendo en cuenta que el único campo no editable es el código, también nos muestra retorno a la vista anterior

Editar Datos De Un Cliente

[Ir a Lista](#)

Codigo: 2000

Nombre de Usuario
Lyzeth Muñoz

Correo
lyz@yahoo.com

[Guardar Cambios](#)

Para que aparezca el botón de **ir a lista** se debe tener en cuenta lo siguiente

```
<ul class="nav nav-pills nav-fill">
<li class="nav-item">
<a class="nav-link active" aria-current="page" asp-action="Index">Ir a
Lista</a>
```

Para que aparezca el botón de **Guardar Cambios** se debe tener en cuenta lo siguiente

```
<div class="col-12">
<button class="btn btn-primary"
type="submit">Guardar
Cambios</button>
```

Para que aparezca el código sin poder editar ya que es la llave primaria se requiere

```
<form asp-action="Editar" class="row g-3">
<div class="col-md-4">
<label class="form-label">Codigo:
@Model.Codigo</label>
<input asp-for="Codigo" type="hidden"
class="form-control"></div>
```

Para que aparezca los campos editables con la información del cliente se debe tener en cuenta lo siguiente

```
<div class="col-md-4">
<label class="form-label">Nombre de Usuario</label>
<input asp-for="Nombre" type="text" class="form-control"
required placeholder="Nombre Apellido">
</div>
<div class="col-md-4">
<label class="form-label">Correo</label>
<input asp-for="Correo" type="email" class="form-control"
required placeholder="usuario@hotmail.com">
</div>
</form>
```

Para realizar la actividad se debe validar en la base de datos también la validación del rol, para mostrar la información del cliente escogido

```
[Authorize (Roles = "Administrador, Coordinador")]
[HttpGet]
public IActionResult Editar (int id)
var bd = new netsenaContext ();
var mycliente = bd. Clientes.Find(id);
return View(mycliente);
[Authorize (Roles = "Administrador, Coordinador")]
[HttpPost]
public IActionResult Editar (Clientes mycliente)
var bd = new netsenaContext ();
var objcliente = bd. Clientes.Find(mycliente. Codigo);
objcliente. Nombre = mycliente. Nombre;
objcliente. Correo = mycliente. Correo;
bd. SaveChanges ();
return RedirectToAction("Index");
```

BORRAR

Esta es la vista que nos muestra los campos que se van a borrar, donde nos aparece un mensaje de alerta y un botón para finalizar la acción.

[Ir a Lista](#)

Borrar Cliente

Realmente esta seguro de borrar los datos

Codigo	Nombre	Correo
3003	Ginsa Calderon Lopes	lopes@gmail.com

Borrar Datos

Para que aparezca el botón de **Borrar Datos** se debe tener en cuenta lo siguiente

```
<form asp-action="Borrar">
<input asp-for="Codigo"
type="hidden" class="form-
control">
<button class="btn btn-
primary"
type="submit">Borrar
Datos</button>
</form></div>
```

Para que aparezca el mensaje de alerta se debe tener en cuenta lo siguiente

```
<div class="p-3 mb-2 bg-danger
text-White fs-3">Realmente
está seguro de borrar los
datos</div>
```

Para que aparezca la información que se va a eliminar se debe tener en cuenta lo siguiente

```
<table class="table">
<thead>
<tr>
<th scope="col">Codigo</th>
<th scope="col">Nombre</th>
<th scope="col">Correo</th>
</tr>
</thead>
<tbody>
<tr>
<th
scope="row">@Model.Codigo</th>
<td>@Model.Nombre</td>
<td>@Model.Correo</td>
</tr>
</tbody>
</table></div>
```

Para realizar la actividad se debe validar en la base de datos también la validación del rol, para mostrar la información del cliente escogido

```
[Authorize]
[HttpGet]
public IActionResult Borrar (int id)
var bd = new netsenaContext ();
var mycliente = bd. Clientes.Find(id);
return View(mycliente);
[Authorize (Roles = "Administrador")]
[HttpPost, ActionName("Borrar")]
public IActionResult ConfirmarBorrar (int id)
var bd = new netsenaContext ();
var mycliente = bd. Clientes.Find(id);
bd. Remove(mycliente);
bd. SaveChanges ();
return RedirectToAction("Index");
```

CREAR

Esta es la vista que nos muestra los campos que se requieren para crear un nuevo cliente en la base de datos

Ir a Lista

Codigo

0000

Escoja el Rol

Nombre de Usuario

Nombre Apellido

Correo

usuario@hotmail.com

Enviar Datos

Para que aparezca el botón de **Enviar Datos** se debe tener en cuenta lo siguiente

```
<div class="col-12">
<button class="btn btn-
primary" type="submit">Enviar
Datos</button>
</div>
```

Para realizar la actividad se debe validar en la base de datos también la validación del rol, para mostrar la información del cliente escogido

```
[Authorize]
public IActionResult Crear ()
var bd = new netsenaContext ();
//Cargar Roles de la base de datos
var listaRoles = bd. Roles.ToList();
ViewBag.Roles = listaRoles;
return View ();
[Authorize (Roles = "Administrador")]
[HttpPost]
public IActionResult Crear (Clientes mycliente)
var bd = new netsenaContext ();
var Existe = bd. Clientes.Find(mycliente. Codigo);
if (Existe == null)
bd. Clientes.Add(mycliente);
bd. SaveChanges ();
return RedirectToAction("Index");
else
ViewData["msj"] = $"El Codigo {mycliente. Codigo}
ya Existe";
return View ();
```

Para que aparezca los campos y las características de la información a llenar se debe tener en cuenta lo siguiente

```
<form asp-action="Crear" class="row g-3 ">
<div class="col-md-12">
<div>
<label class="form-label">Codigo</label>
<input asp-for="Codigo" type="text" class="form-control"
required placeholder="0000">
<span class="bg-warning" text. dark> @ViewData["msj"]
</span>
</div>
<div class="col-md-6">
<select asp-for="Rol" class="form-select" aria-
label="Default select example">
<option selected>Escoja el Rol</option>
@foreach (var rol in milista)
<option value="@rol. Rol">@rol. Rol</option>
</select>
</div>
</div>
<div class="col-md-12">
<div class="col-md-6">
<label class="form-label">Nombre de Usuario</label>
<input asp-for="Nombre" type="text" class="form-
control" required placeholder="Nombre Apellido">
</div>
<div class="col-md-6">
<label class="form-label">Correo</label>
<input asp-for="Correo" type="email" class="form-
control" required placeholder="usuario@hotmail.com">
<div class="invalid-feedback">
please choose a username.
</div>
```