

## Creando Sitio Web ASP.Net – MySql (Fase-1 Registro de Usuario)

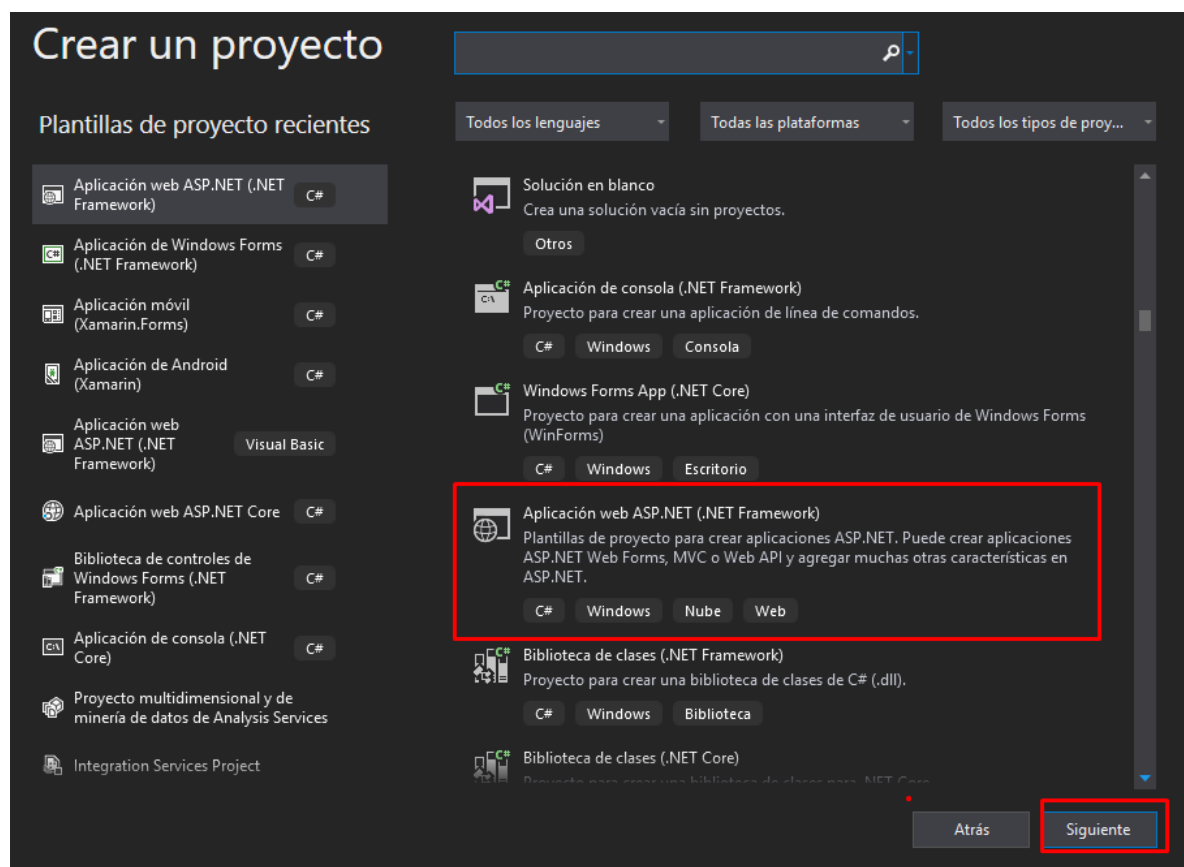
La presente guía tiene la finalidad ayudar a comprender al estudiantes como se realiza un proyecto web CRUD con la tecnología Asp.net y el gestor de base de datos MySQL. Cabe aclarar y es necesario tomar nota que posteriormente no profundizaremos en algunos temas por razones de tiempo, pero trataremos de explicarlos y despejar las dudas. Ya que es necesario que el estudiante descubra otros temas con la finalidad de motivarlos a buscar; proporcionaremos los nombres:

- Estilos Css
- Bootstrap
- Notificaciones al usuario con jQuery Sweet Alert
- Navbar
- Carrusel

**Nota aclaratoria:** es mucho más fácil aplicar los ejemplos durante los ejercicios para poder comprender como funcionan las diferentes tecnologías al momento de programar.

Comencemos.

Como primero paso debemos crear un proyecto Asp.Net como se muestra en la imagen:



Como vemos tenemos un proyecto vacío. Agregaremos una serie de nuevos formularios aspx y les llamaremos:

- Login.aspx
- Inicio.aspx
- Registro.aspx

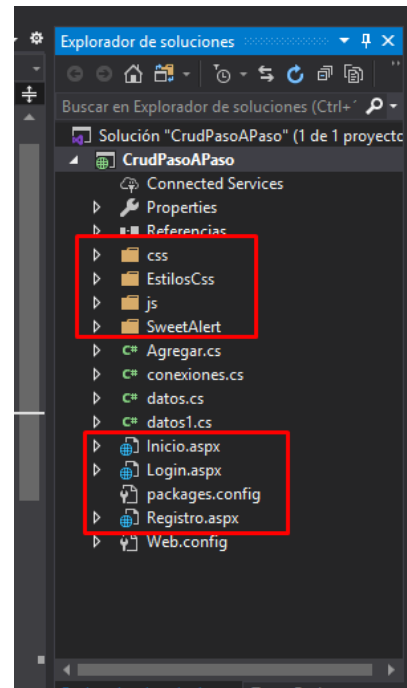
Posteriormente exportaremos una serie de carpetas que nos servirán para poder hacer funcionar diferentes acciones.

La carpeta css tiene todos los estilos manejados por bootstrap que utilizaremos.

EstilosCss contiene todos los estilos de los formularios web.

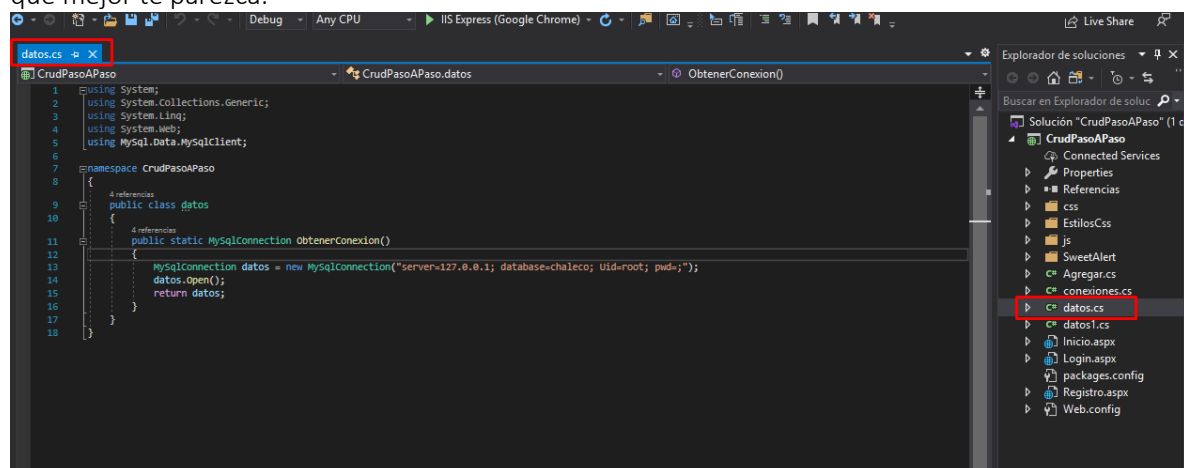
Js contiene los script que se utilizarán.

SweetAlert contiene todas las animaciones de notificaciones que recibirá el usuario.

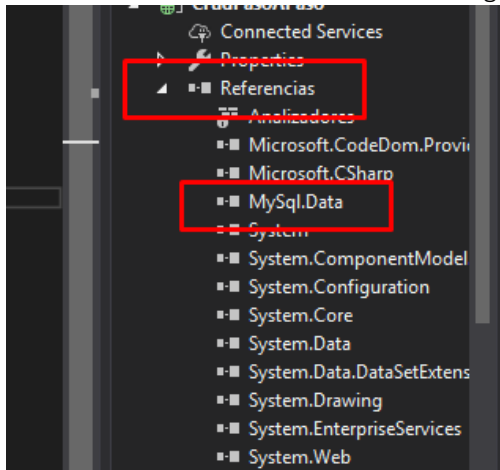


Posterior a esto crearemos las clases necesarias para poder hacer las diferentes peticiones; explicaremos cada clase.

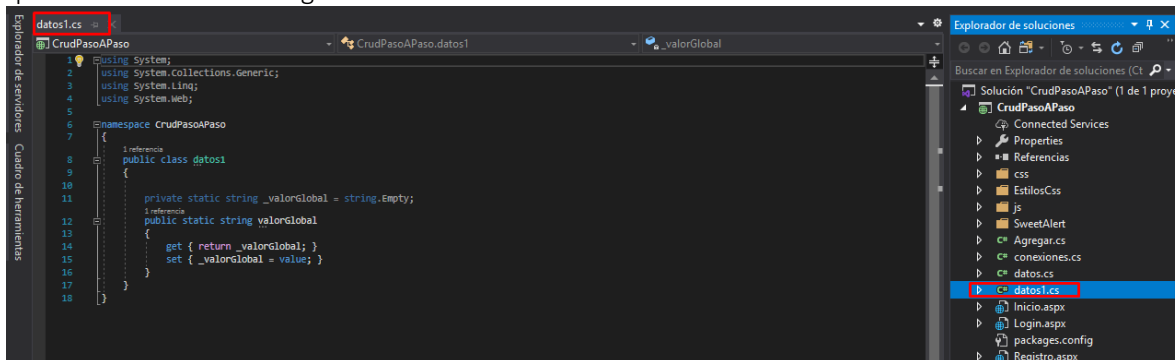
La clase datos.cs contendrá la conexión a la base de datos; puedes cambiar el nombre de la clase que mejor te parezca.



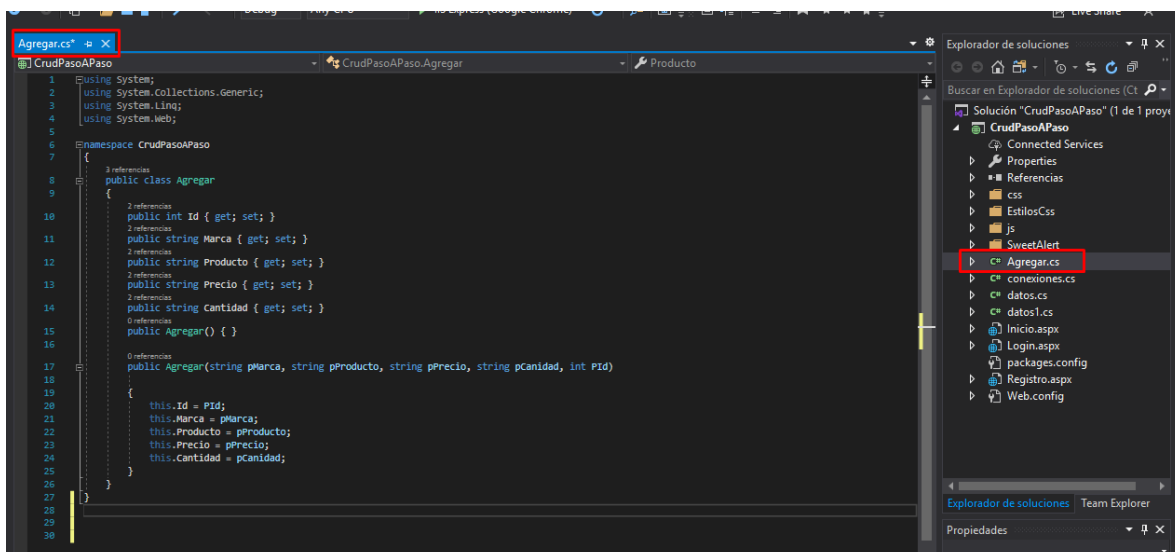
Recuerda que utilizaremos el conector Mysql. Este el mismo conector que utilizamos en el diseño de CRUD con WindowsForm. Debes de agregarlos a referencias.



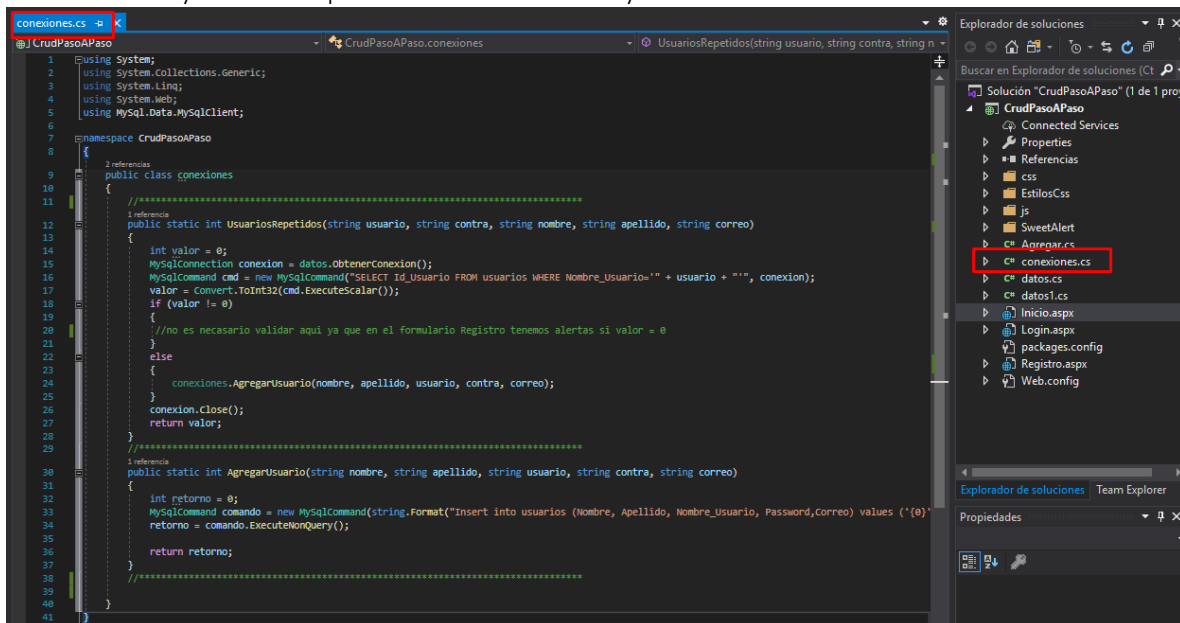
Crearemos la clase datos1.cs que se encargará de capturar los parámetros de la variable usuario que utilizaremos en el Login.



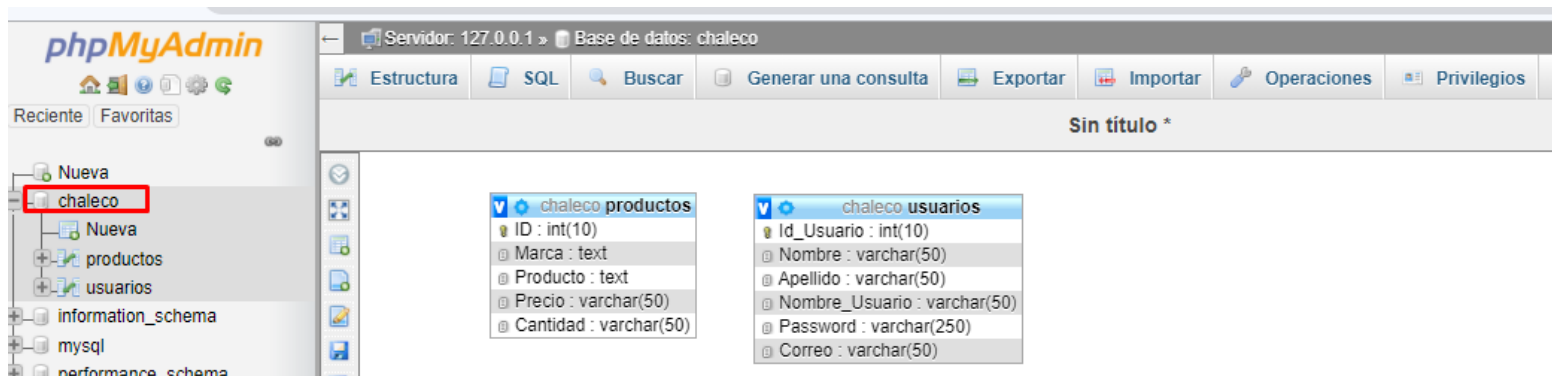
Crearemos la clase Agregar.cs que leerá los parámetros y luego asignara dicho valores a las variables creadas dentro de este:



Crearemos la clase conexiones.cs que contendrá los métodos que utilizaremos para Agregar un nuevo usuario y el método para saber si ese usuario ya existe en nuestra base de datos.



La BD que utilizaremos en este ejemplo. Solo debes de crear las dos tablas como se muestra en la imagen.



Ahora crearemos nuestros Front-End de cada formulario Web y su respectivo Back-End. Comenzaremos con nuestro Login.aspx (Front-End). En cuadro rojo llamamos o hacemos referencias a todas las librerías que se le indiquen; están dentro del encabezado.

- 1º línea roja: agregamos un botón para poder hacer el registro del usuario.
- 2º línea roja: uso del textbox y lo mismo se hace en los demás objetos.

Si presta atención todos los objetos llaman una clase que se encuentra en /EstilosCss/Login.css

```
1 <!DOCTYPE html>
2 <html xmlns="http://www.w3.org/1999/xhtml">
3 <head runat="server">
4 <meta http-equiv="Content-Type" content="text/html; charset=utf-8" />
5 <link href="css/bootstrap.min.css" rel="stylesheet" />
6 <link href="EstilosCss/Login.css" rel="stylesheet" />
7 <script src="SweetAlert/sweetalert2.all.min.js"></script>
8 <script src="SweetAlert/sweetalert2.js"></script>
9 </head>
10 <title>Login</title>
11 </head>
12 <body>
13 <form id="form1" runat="server">
14 <nav class="navbar navbar-expand-sm bg-dark navbar-dark fixed-top">
15 <!-- Brand -->
16 <br />
17 <a class="navbar-brand" href="#">Drugstore</a>
18 <asp:Button ID="btnregistro" class="btn btn-outline-success my-2 my-sm-0" type="submit" runat="server" Text="Registro" OnClick="Button1_Click" />
19 <br />
20 </nav>
21 <div class="my-content">
22 <div class="container">
23 <div class="row">
24 <div class="col-sm-12">
25 <div>
26 </div>
27 </div>
28 <div class="row">
29 <div class="col-sm-6 col-sm-offset-3 myform-cont">
30 <div class="myform-top">
31 <div class="myform-top-left">
32 <h3>Ingresa a Drugstore</h3>
33 <p>Digite tu usuario y contraseña</p>
34 </div>
35 <div class="myform-top-right">
36 <!--cl class="fas fa-key"></i-->
37 </div>
38 </div>
39 <div class="myform-bottom">
40 <!--form role="form" action="" method="post" class=""-->
41 <div class="form-group">
42 <asp:TextBox ID="TxtUsuario" runat="server" placeholder="Usuario..." type="text" class="form-control" onpaste="return false" minlength="4" MaxLength="15" />
43 </div>
44 <div class="form-group">
45 <asp:TextBox ID="TxtContra" runat="server" placeholder="Contraseña..." type="password" class="form-control" onpaste="return false" minlength="2" MaxLength="15" />
46 </div>
47 <div>
48 <asp:Button runat="server" Text="Entrar" CssClass="mybtn" OnClick="Unnamed1_Click"></asp:Button>
49 </div>
50 <div>
51 <asp:HyperLink ID="olvidaste" runat="server" CssClass="olvidaste" Text="¿Has olvidado tu contraseña?" NavigateUrl="RecuperarCuenta.aspx" />
52 </div>
53 <asp:Literal ID="alerta" runat="server" Text=""></asp:Literal>
54 </div>
55 </div>
56 </div>
57 </div>
58 </div>
59 <script src="https://cdn.jsdelivr.net/npm/sweetalert2@8"></script>
60 <script src="js/bootstrap.min.js"></script>
61 </form>
62 </body>
63 </html>
64
```

Ahora trabajaremos nuestro Back-End de Login.aspx.cs

```
Login.aspx.cs
CrudPasoAPaso
CrudPasoAPaso.Login
Button1_Click(object sender, EventArgs e)

1 using System;
2 using System.Collections.Generic;
3 using System.Linq;
4 using System.Web;
5 using System.Web.UI;
6 using System.Web.UI.WebControls;
7 using MySql.Data.MySqlClient;
8 using System.Data;
9 using System.Security.Cryptography;
10 using System.Text;
11 using System.IO;
12 using System.Data.SqlClient;
13
14
15 namespace CrudPasoAPaso
16 {
17     1 referencia
18     public partial class Login : System.Web.UI.Page
19     {
20         0 referencias
21         protected void Unnamed1_Click(object sender, EventArgs e)
22         {
23             if (TxtContra.Text != "" && TxtUsuario.Text != "") //Validando espacio vacio
24             {
25                 string contra, usuario;
26                 contra = EncryptString(TxtContra.Text, initVector);
27                 usuario = TxtUsuario.Text;
28                 datos1.valorGlobal = usuario; //llamamos la clase datos1
29
30                 MySqlConnection conexion = new MySqlConnection("Server=127.0.0.1; database= chaleco; Uid=root; pwd=;");
31                 var cmd = "SELECT Id_Usuario from usuarios WHERE Nombre_Usuario='" + usuario + "' AND Password='" + contra + "'";
32                 MySqlCommand comando = new MySqlCommand(cmd, conexion);
33                 conexion.Open();
34                 int retorno = Convert.ToInt32(comando.ExecuteScalar());
35                 if (retorno != 0)
36                 {
37                     Session["usermane"] = TxtUsuario; //Creamos una session
38                     Response.Redirect("Inicio.aspx"); //Nos redirecciona a nuestro formulario
39                 }
40                 else
41                 {
42                     //Mensaje de alerta que se mostrará; como que fuese un MessageBox solo que con animacion javascript de SweetAlert
43                     alerta.Text = "<script>Swal.fire('Error de Credenciales', 'Su usuario o contraseña no son correctos', 'error') </script>";
44                     TxtContra.Text = ""; //Limpia los text box
45                     TxtUsuario.Text = "";
46                 }
47             }
48             else
49             {
50                 alerta.Text = "<script>Swal.fire('ADVERTENCIA', 'No deje espacios en blanco', 'error') </script>";
51             }
52         }
53
54         private const string initVector = "pemgail9uzpgr188"; //llave del encryptado
55         private const int keysize = 256; // Esta constante se usa para determinar el tamaño de clave del algoritmo de cifrado
56         //Encryptar
57         1 referencia
58         public static string EncryptString(string plainText, string passPhrase)
59         {
60             byte[] initVectorBytes = Encoding.UTF8.GetBytes(initVector);
61             byte[] plainTextBytes = Encoding.UTF8.GetBytes(plainText);
62             PasswordDeriveBytes password = new PasswordDeriveBytes(passPhrase, null);
63             byte[] keyBytes = password.GetBytes(keysize / 8);
64             RijndaelManaged symmetricKey = new RijndaelManaged();
65             symmetricKey.Mode = CipherMode.CBC;
66             ICryptoTransform encryptor = symmetricKey.CreateEncryptor(keyBytes, initVectorBytes);
67             MemoryStream memoryStream = new MemoryStream();
68             CryptoStream cryptoStream = new CryptoStream(memoryStream, encryptor, CryptoStreamMode.Write);
69             cryptoStream.Write(plainTextBytes, 0, plainTextBytes.Length);
70             cryptoStream.FlushFinalBlock();
71             byte[] cipherTextBytes = memoryStream.ToArray();
72             memoryStream.Close();
73             cryptoStream.Close();
74             return Convert.ToBase64String(cipherTextBytes);
75         }
76
77         0 referencias
78         protected void Button1_Click(object sender, EventArgs e)
79         {
80             Response.Redirect("Registro.aspx"); //nos redirecciona al formulario Web de registro
81         }
82     }
83 }
```

Ok listo; con nuestro Login ya estamos completo Puede ejecutarlo para que observes. Ahora trabajaremos nuestro formulario web Registro.aspx para poder crear nuestros usuarios.

Este formulario Web Registro.aspx (Front End) Le agregaremos los textbox que utilizaremos para poder registrar nuestros usuarios.

```
<%@ Page Language="C#" AutoEventWireup="true" CodeBehind="Registro.aspx.cs" Inherits="CrudPasoAPaso.Registro" %>
<!DOCTYPE html>

<html xmlns="http://www.w3.org/1999/xhtml">
<head runat="server">
  <meta http-equiv="Content-Type" content="text/html; charset=utf-8" />
  <link href="css/bootstrap.min.css" rel="stylesheet" />
  <link href="EstilosCss/Login.css" rel="stylesheet" />
  <script src="SweetAlert/sweetalert2.all.min.js"></script>
  <script src="SweetAlert/sweetalert2.js"></script>
</head>
<body>
  <form id="form1" runat="server">
    <nav class="navbar navbar-expand-sm bg-dark navbar-dark fixed-top">
      <!-- Brand -->
      <a class="navbar-brand" href="#">Drugstore</a>
      <asp:Button ID="btnlogin" class="btn btn-outline-success my-2 my-sm-0" type="submit" runat="server" Text="Login" OnClick="btnlogin_Click" />
    </nav>
    <div class="my-content">
      <div class="container">
        <div class="row">
          <div class="col-sm-12">
            </div>
          </div>
          <div class="row">
            <div class="col-sm-6 col-sm-offset-3 myform-cont">
              <h1>Registro </h1>
              <div class="myform-top">
                <div class="myform-top-left">
                  <h3>Drugstore</h3>
                </div>
                <div class="myform-bottom">
                  <div class="form-group">
                    <asp:TextBox ID="txtfirst" runat="server" placeholder="Nombres..." type="text" class="form-control" onpaste="return false" minlength="5" MaxLength="30"
onkeypress="return validar(event)"></asp:TextBox>
                  </div>
                  <div class="form-group">
                    <asp:TextBox ID="txtapellido" runat="server" placeholder="Apellidos..." type="text" class="form-control" onpaste="return false" minlength="5" MaxLength="30"
onkeypress="return validar(event)"></asp:TextBox>
                  </div>
                  <div class="form-group">
                    <asp:TextBox ID="txtusuario" runat="server" placeholder="Usuario..." type="text" class="form-control" onpaste="return false" minlength="5" MaxLength="15" />
                  </div>
                  <div class="form-group">
                    <asp:TextBox ID="txtclave" runat="server" placeholder="Contraseña..." type="password" class="form-control" onpaste="return false" minlength="2" MaxLength="15" />
                  </div>
                  <div class="form-group">
                    <asp:TextBox ID="txtconfirm" runat="server" placeholder="Repita Contraseña..." type="password" class="form-control" onpaste="return false" minlength="2" MaxLength="15"
/>
                  </div>
                  <div class="form-group">
                    <asp:TextBox ID="txtgmail" runat="server" placeholder="Gmail..." type="email" class="form-control" onpaste="return false" minlength="5" MaxLength="30" />
                  </div>
                  <asp:Button runat="server" Text="Registrarse" CssClass="mybtn" OnClick="Unnamed1_Click"></asp:Button>
                  <asp:Literal ID="alerta" runat="server" Text=""></asp:Literal>
                </div>
              </div>
            </div>
          </div>
        </div>
        <script src="https://cdn.jsdelivr.net/npm/sweetalert2@8"></script>
        <script src="js/bootstrap.min.js"></script>
      </form>
    </body>
  </html>
```

Este formulario Web Registro.aspx.cs (Back End) tendrá validaciones, con el objetivo de evitar inconsistencias e inyecciones SQL.

```
using System;
using System.Collections.Generic;
using System.Linq;
using System.Web;
using System.Web.UI;
using System.Web.UI.WebControls;
using System.Security.Cryptography;
using System.Text;
using System.IO;

namespace CrudPasoAPaso
{
    public partial class Registro : System.Web.UI.Page
    {
        protected void Unnamed1_Click(object sender, EventArgs e)
        {
            if (txtusuario.Text.Trim() != "" && txtclave.Text.Trim() != "" && txtconfirm.Text.Trim() != "" && txtfirst.Text.Trim() != "" && txtapellido.Text.Trim() != "" && txtgmail.Text.Trim() != "")
            {
                if (txtclave.Text == txtconfirm.Text)
                {
                    string nombre;
                    string apellido;
                    string encriptada;
                    string usuario;
                    string correo;

                    nombre = txtfirst.Text;
                    apellido = txtapellido.Text;
                    usuario = txtusuario.Text;
                    encriptada = EncryptString(txtclave.Text, initVector);
                    correo = txtgmail.Text;
                    if (conexiones.UsuariosRepetidos(usuario, encriptada, nombre, apellido, correo) == 0)
                    {
                        alerta.Text = "<script>Swal.fire('Registrado con éxito', '¡Gracias por preferirnos!', 'success'); </script>";

                        txtfirst.Text = "";
                        txtapellido.Text = "";
                        txtusuario.Text = "";
                        txtclave.Text = "";
                        txtconfirm.Text = "";
                        txtgmail.Text = "";
                    }
                    else
                    {
                        alerta.Text = "<script>Swal.fire('Este usuario ya existe', 'Elige nuevo nombre de usuario', 'error'); </script>";
                    }
                }
                else
                {
                    alerta.Text = "<script>Swal.fire('Contraseña incorrecta', 'Repita su contraseña.', 'error'); </script>";
                }
            }
            else
            {
                alerta.Text = "<script>Swal.fire('ADVERTENCIA', 'No deje espacios en blanco', 'error') </script>";
            }
        }

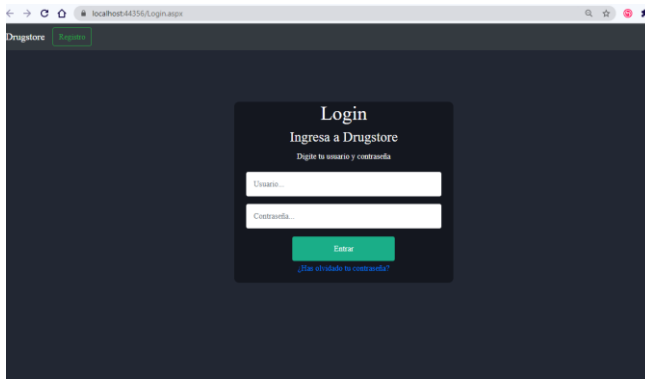
        private const string initVector = "pemgaill9uzpgzl88";
        // This constant is used to determine the keysize of the encryption algorithm
        private const int keysize = 256;
        // Encrypt
        public static string EncryptString(string plainText, string passPhrase)
        {
            byte[] initVectorBytes = Encoding.UTF8.GetBytes(initVector);
            byte[] plainTextBytes = Encoding.UTF8.GetBytes(plainText);
            PasswordDeriveBytes password = new PasswordDeriveBytes(passPhrase, null);
            byte[] keyBytes = password.GetBytes(keysize / 8);
            RijndaelManaged symmetricKey = new RijndaelManaged();
            symmetricKey.Mode = CipherMode.CBC;
            ICryptoTransform encryptor = symmetricKey.CreateEncryptor(keyBytes, initVectorBytes);
            MemoryStream memoryStream = new MemoryStream();
            CryptoStream cryptoStream = new CryptoStream(memoryStream, encryptor, CryptoStreamMode.Write);
            cryptoStream.Write(plainTextBytes, 0, plainTextBytes.Length);
            cryptoStream.FlushFinalBlock();
            byte[] cipherTextBytes = memoryStream.ToArray();
            memoryStream.Close();
            cryptoStream.Close();
            return Convert.ToBase64String(cipherTextBytes);
        }

        protected void btnlogin_Click(object sender, EventArgs e)
        {
            Response.Redirect("~/Login.aspx");
        }
    }
}
```

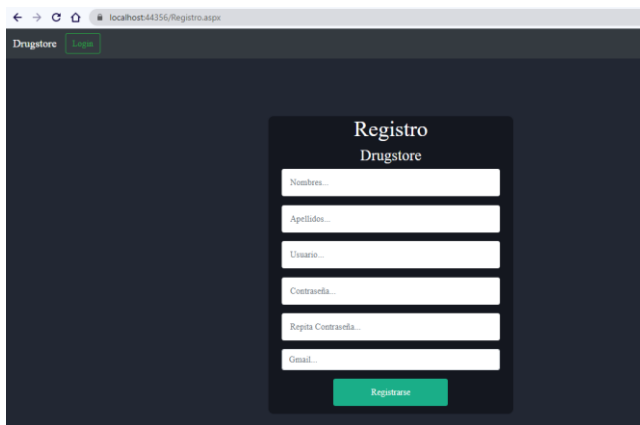


Hasta este punto debe de funcionar el registro de usuarios y el Login; además si recuerdas al inicio creamos los tres formularios web que utilizaremos y uno de ellos es el del CRUD que se llamará inicio.

Vistas de los Tres Formularios:

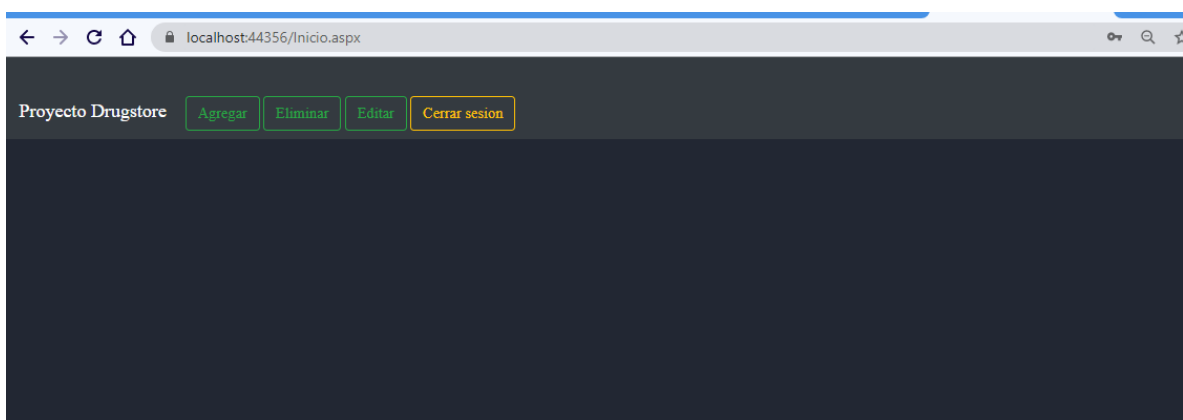


The screenshot shows a web browser window with the address bar displaying 'localhost:44356/Login.aspx'. The page has a dark background. At the top left, there is a 'Drugstore' label and a green 'Registra' button. The main content is a white box titled 'Login' with the subtitle 'Ingresa a Drugstore' and the instruction 'Digite tu usuario y contraseña'. It contains two input fields: 'Usuario...' and 'Contraseña...'. Below these is a green 'Entrar' button and a link that says '¿Has olvidado tu contraseña?'.



The screenshot shows a web browser window with the address bar displaying 'localhost:44356/Registro.aspx'. The page has a dark background. At the top left, there is a 'Drugstore' label and a green 'Login' button. The main content is a white box titled 'Registro' with the subtitle 'Drugstore'. It contains six input fields: 'Nombre...', 'Apellidos...', 'Usuario...', 'Contraseña...', 'Repita Contraseña...', and 'Email...'. Below these is a green 'Regístrate' button.

En este formulario aun no le hemos ingresado programación; pero se vera de esta manera o mejor aún:



The screenshot shows a web browser window with the address bar displaying 'localhost:44356/Inicio.aspx'. The page has a dark background. At the top left, there is a 'Proyecto Drugstore' label. To its right are four buttons: 'Agregar' (green), 'Eliminar' (green), 'Editar' (green), and 'Cerrar sesion' (yellow). The rest of the page is empty.

## Creando Sitio Web ASP.Net – MySql (Fase-2 CREATE)