**WEB SERVICE DE LA APLICACIÓN**

**ARCHIVO SERVICE.CS**

using System;

using System.Collections.Generic;

using System.Linq;

using System.Runtime.Serialization;

using System.ServiceModel;

using System.ServiceModel.Web;

using System.Text;

using System.Data;

using System.Data.SqlClient;

using System.Configuration;

public class Service : IService

{

DataSet ds = new DataSet();

SqlDataAdapter da;

string conexion = ConfigurationManager.ConnectionStrings["conexionSQL"].ToString();

public DataSet Insertar\_usuario(string Cod\_usuario, string nombre, string apellido, int edad, string sexo, string fechanac, string pais, string depto, string telefono, string email, string pasword, string fecharegistro)

{

da = new SqlDataAdapter("Insertar\_usuario", conexion);

da.SelectCommand.CommandType = CommandType.StoredProcedure;

da.SelectCommand.Parameters.AddWithValue("@Cod\_usuario", Cod\_usuario);

da.SelectCommand.Parameters.AddWithValue("@nombre", nombre);

da.SelectCommand.Parameters.AddWithValue("@Apellido", apellido);

da.SelectCommand.Parameters.AddWithValue("@Edad", edad);

da.SelectCommand.Parameters.AddWithValue("@Sexo", sexo);

da.SelectCommand.Parameters.AddWithValue("@Fecha\_Nac", fechanac);

da.SelectCommand.Parameters.AddWithValue("@Pais", pais);

da.SelectCommand.Parameters.AddWithValue("@Depto", depto);

da.SelectCommand.Parameters.AddWithValue("@Telefono", telefono);

da.SelectCommand.Parameters.AddWithValue("@Email", email);

da.SelectCommand.Parameters.AddWithValue("@Pasword", pasword);

da.SelectCommand.Parameters.AddWithValue("@Fecharegistro", fecharegistro);

da.Fill(ds, "Usuario Insertado");

return ds;

}

public DataSet Modificar\_Usuario(string Cod\_usuario, string nombre, string apellido, int edad, string sexo, string fechanac, string pais, string depto, string telefono, string email, string pasword)

{

da = new SqlDataAdapter("Modificar\_Usuario", conexion);

da.SelectCommand.CommandType = CommandType.StoredProcedure;

da.SelectCommand.Parameters.AddWithValue("@Cod\_usuario", Cod\_usuario);

da.SelectCommand.Parameters.AddWithValue("@nombre", nombre);

da.SelectCommand.Parameters.AddWithValue("@Apellido", apellido);

da.SelectCommand.Parameters.AddWithValue("@Edad", edad);

da.SelectCommand.Parameters.AddWithValue("@Sexo", sexo);

da.SelectCommand.Parameters.AddWithValue("@Fecha\_Nac", fechanac);

da.SelectCommand.Parameters.AddWithValue("@Pais", pais);

da.SelectCommand.Parameters.AddWithValue("@Depto", depto);

da.SelectCommand.Parameters.AddWithValue("@Telefono", telefono);

da.SelectCommand.Parameters.AddWithValue("@Email", email);

da.SelectCommand.Parameters.AddWithValue("@Pasword", pasword);

da.Fill(ds, "Usuario modificado");

return ds;

}

public DataSet Eliminar\_Usuario(string Cod\_usuario)

{

da = new SqlDataAdapter("Eliminar\_Usuario", conexion);

da.SelectCommand.CommandType = CommandType.StoredProcedure;

da.SelectCommand.Parameters.AddWithValue("@Cod\_usuario", Cod\_usuario);

da.Fill(ds, "Usuario Eliminado");

return ds;

}

public DataSet Consultar\_Usuario()

{

da = new SqlDataAdapter("Consultar\_Usuario", conexion);

da.SelectCommand.CommandType = CommandType.StoredProcedure;

da.Fill(ds, "Data de usuario");

return ds;

}

public DataSet Insertar\_Tema(string nombre\_tema)

{

da = new SqlDataAdapter("Insertar\_Tema", conexion);

da.SelectCommand.CommandType = CommandType.StoredProcedure;

da.SelectCommand.Parameters.AddWithValue("@nombre\_tema", nombre\_tema);

da.Fill(ds, "Tema Insertado");

return ds;

}

public DataSet Modificar\_Tema(int ID\_tema, string nombre\_tema)

{

da = new SqlDataAdapter("Modificar\_Tema", conexion);

da.SelectCommand.CommandType = CommandType.StoredProcedure;

da.SelectCommand.Parameters.AddWithValue("@ID\_tema",ID\_tema );

da.SelectCommand.Parameters.AddWithValue("@nombre\_tema", nombre\_tema);

da.Fill(ds, "Tema Modificado");

return ds;

}

public DataSet Eliminar\_Tema(int ID\_tema)

{

da = new SqlDataAdapter("Eliminar\_Tema", conexion);

da.SelectCommand.CommandType = CommandType.StoredProcedure;

da.SelectCommand.Parameters.AddWithValue("@ID\_tema", ID\_tema);

da.Fill(ds, "Tema Eliminado");

return ds;

}

public DataSet Consultar\_tema()

{

da = new SqlDataAdapter("Consultar\_", conexion);

da.SelectCommand.CommandType = CommandType.StoredProcedure;

da.Fill(ds, "Data ded Temas");

return ds;

}

public DataSet Insertar\_curso(string nombre\_curso, string estado, string ID\_tema)

{

da = new SqlDataAdapter("Insertar\_curso", conexion);

da.SelectCommand.CommandType = CommandType.StoredProcedure;

da.SelectCommand.Parameters.AddWithValue("@nombre\_curso", nombre\_curso);

da.SelectCommand.Parameters.AddWithValue("@Estado", estado);

da.SelectCommand.Parameters.AddWithValue("@ID\_tema", ID\_tema);

da.Fill(ds, "Curso Insertado");

return ds;

}

public DataSet Modificar\_curso(int id\_curso, string nombre\_curso, string estado, string ID\_tema)

{

da = new SqlDataAdapter("Modificar\_curso", conexion);

da.SelectCommand.CommandType = CommandType.StoredProcedure;

da.SelectCommand.Parameters.AddWithValue("@id\_curso", id\_curso);

da.SelectCommand.Parameters.AddWithValue("@nombre\_curso", nombre\_curso);

da.SelectCommand.Parameters.AddWithValue("@Estado", estado);

da.SelectCommand.Parameters.AddWithValue("@ID\_tema", ID\_tema);

da.Fill(ds, "Curso Modificado");

return ds;

}

public DataSet Eliminar\_curso(int id\_curso)

{

da = new SqlDataAdapter("Eliminar\_curso", conexion);

da.SelectCommand.CommandType = CommandType.StoredProcedure;

da.SelectCommand.Parameters.AddWithValue("@id\_curso", id\_curso);

da.Fill(ds, "Curso Eliminado");

return ds;

}

public DataSet Consultar\_curso()

{

da = new SqlDataAdapter("Consultar\_curso", conexion);

da.SelectCommand.CommandType = CommandType.StoredProcedure;

da.Fill(ds, "Data de Cursos");

return ds;

}

public DataSet Insertar\_Categoria(string nombre\_cat)

{

da = new SqlDataAdapter("Insertar\_Categoria", conexion);

da.SelectCommand.CommandType = CommandType.StoredProcedure;

da.SelectCommand.Parameters.AddWithValue("@nombre\_cat", nombre\_cat);

da.Fill(ds, "Categoria creada");

return ds;

}

public DataSet Modificar\_Categoria(int ID\_Categoria, string nombre\_cat)

{

da = new SqlDataAdapter("Modificar\_Categoria", conexion);

da.SelectCommand.CommandType = CommandType.StoredProcedure;

da.SelectCommand.Parameters.AddWithValue("@ID\_Categoria", ID\_Categoria);

da.SelectCommand.Parameters.AddWithValue("@nombre\_cat", nombre\_cat);

da.Fill(ds, "Categoria modificada");

return ds;

}

public DataSet Eliminar\_Categoria(int ID\_Categoria, string nombre\_cat)

{

da = new SqlDataAdapter("Eliminar\_Categoria", conexion);

da.SelectCommand.CommandType = CommandType.StoredProcedure;

da.SelectCommand.Parameters.AddWithValue("@ID\_Categoria", ID\_Categoria);

da.Fill(ds, "Categoria Eliminada");

return ds;

}

public DataSet Consultar\_Categoria()

{

da = new SqlDataAdapter("Consultar\_Categoria", conexion);

da.SelectCommand.CommandType = CommandType.StoredProcedure;

return ds;

}

}

**ARCHIVO ISERVICE.CS**

using System;

using System.Collections.Generic;

using System.Linq;

using System.Runtime.Serialization;

using System.ServiceModel;

using System.ServiceModel.Web;

using System.Text;

using System.Data;

// NOTA: puede usar el comando "Rename" del menú "Refactorizar" para cambiar el nombre de interfaz "IService1" en el código y en el archivo de configuración a la vez.

[ServiceContract]

public interface IService

{

[OperationContract]

DataSet Insertar\_usuario(string Cod\_usuario, string nombre, string apellido, int edad, string sexo, string fechanac, string pais, string depto, string telefono, string email, string pasword, string fecharegistro);

[OperationContract]

DataSet Modificar\_Usuario(string Cod\_usuario, string nombre, string apellido,

int edad, string sexo, string fechanac, string pais, string depto,

string telefono, string email, string pasword);

[OperationContract]

DataSet Eliminar\_Usuario(string Cod\_usuario);

[OperationContract]

DataSet Consultar\_Usuario();

[OperationContract]

DataSet Insertar\_Tema(string nombre\_tema);

[OperationContract]

DataSet Modificar\_Tema(int ID\_tema, string nombre\_tema);

[OperationContract]

DataSet Eliminar\_Tema(int ID\_tema);

[OperationContract]

DataSet Consultar\_tema();

[OperationContract]

DataSet Insertar\_curso(string nombre\_curso, string estado, string ID\_tema);

[OperationContract]

DataSet Modificar\_curso(int id\_curso, string nombre\_curso, string estado,

string ID\_tema);

[OperationContract]

DataSet Eliminar\_curso(int id\_curso);

[OperationContract]

DataSet Consultar\_curso();

[OperationContract]

DataSet Insertar\_Categoria(string nombre\_cat);

[OperationContract]

DataSet Modificar\_Categoria(int ID\_Categoria, string nombre\_cat);

[OperationContract]

DataSet Eliminar\_Categoria(int ID\_Categoria, string nombre\_cat);

[OperationContract]

DataSet Consultar\_Categoria();

}

**ARCHIVO WEB.CONFIG**

<?xml version="1.0"?>

<configuration>

<appSettings>

<add key="aspnet:UseTaskFriendlySynchronizationContext" value="true"/>

</appSettings>

<connectionStrings>

<add name="conexionSQL" connectionString="Data Source=DESKTOP-FUSEMAO\SQLEXPRESS; Initial Catalog=ELERNING; User='sa' ; Password=''"/>

</connectionStrings>

<system.web>

<compilation debug="true" targetFramework="4.5.2"/>

<httpRuntime targetFramework="4.5.2"/>

</system.web>

<system.serviceModel>

<behaviors>

<serviceBehaviors>

<behavior>

<!-- Para evitar revelar información de los metadatos, establezca el valor siguiente en false antes de la implementación -->

<serviceMetadata httpGetEnabled="true" httpsGetEnabled="true"/>

<!-- Para recibir detalles de las excepciones en los fallos, con el fin de poder realizar la depuración, establezca el valor siguiente en true. Para no revelar información sobre las excepciones, establézcalo en false antes de la implementación -->

<serviceDebug includeExceptionDetailInFaults="false"/>

</behavior>

</serviceBehaviors>

</behaviors>

<protocolMapping>

<add binding="basicHttpsBinding" scheme="https"/>

</protocolMapping>

<serviceHostingEnvironment aspNetCompatibilityEnabled="true" multipleSiteBindingsEnabled="true"/>

</system.serviceModel>

<system.webServer>

<modules runAllManagedModulesForAllRequests="true"/>

<!--

Para examinar el directorio raíz de la aplicación web durante la depuración, establezca el valor siguiente en true.

Establézcalo en false antes de la implementación para evitar revelar información sobre la carpeta de aplicación web.

-->

<directoryBrowse enabled="true"/>

</system.webServer>

</configuration>