# **Manual de Usuario - Sistema de Administración de Empleados**

## **Descripción General**

Este sistema permite gestionar empleados, sus teléfonos, direcciones y puestos en una empresa, con funcionalidades de autenticación de usuarios, un menú de navegación y operaciones CRUD (Crear, Leer, Actualizar y Eliminar) en cada sección. La interfaz incluye una página de inicio de sesión, una página de inicio principal y enlaces de navegación para acceder a las secciones específicas.

### **Requisitos previos**

1. **Visual Studio** (con soporte para ASP.NET Web Forms).
2. **SQL Server** y SQL Server Management Studio (para gestionar la base de datos).
3. **Base de datos**: un esquema configurado con tablas de empleados, teléfonos, direcciones, puestos y usuarios.
4. **Configuración de conexión** en el archivo Web.config de la aplicación para conectarse a la base de datos.

## **Pasos para Crear el Programa**

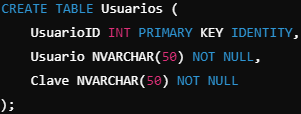
### **1. Creación del Proyecto en Visual Studio**

1. Abre Visual Studio y crea un **nuevo proyecto** de tipo **ASP.NET Web Application (.NET Framework)**.
2. Selecciona **Web Forms** para desarrollar el sistema utilizando formularios.

### **2. Configuración de la Base de Datos en SQL Server**

1. En SQL Server, crea una nueva base de datos llamada AdminEmpleadosDB.
2. Crea las tablas necesarias para el sistema, como Usuarios, Empleados, TelefonosEmpleado, DireccionesEmpleado, PuestosEmpleado.

**Ejemplo de la tabla Usuarios:**



1. Configura la conexión a la base de datos en el archivo Web.config:

<connectionStrings>

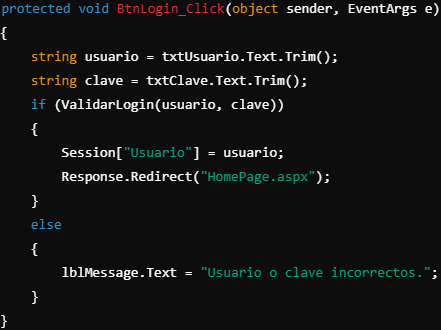
<add name="DBConnectionString" connectionString="Data Source=localhost;Initial Catalog=AdminEmpleadosDB;Integrated Security=True" providerName="System.Data.SqlClient" />

</connectionStrings>

### **3. Creación de Páginas Web**

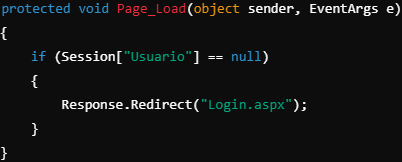
#### **Login.aspx y Login.aspx.cs**

* **Login.aspx**: crea un formulario de inicio de sesión con campos de texto para Usuario y Clave, y un botón de inicio de sesión.
* **Login.aspx.cs**: escribe el código de C# para validar las credenciales del usuario ingresado.



#### **HomePage.aspx y HomePage.aspx.cs**

* **HomePage.aspx**: crea un mensaje de bienvenida y un menú de opciones para navegar a las páginas CRUD de empleados, teléfonos, direcciones y puestos.
* **HomePage.aspx.cs**: protege la página, redirigiendo a Login.aspx si no hay una sesión activa.



#### **Site.Master y Site.Master.cs**

* **Site.Master**: incluye el menú de navegación principal con enlaces a cada sección del sistema.
* **Site.Master.cs**: verifica la sesión de usuario en Page\_Load. Si no hay sesión activa, redirige a Login.aspx.

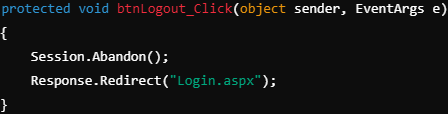
### **4. Creación de las Páginas CRUD para Empleados**

#### **Ejemplo: Empleados.aspx y Empleados.aspx.cs**

* **Empleados.aspx**: crea un formulario para agregar o actualizar empleados y una tabla para mostrar la lista de empleados.
* **Empleados.aspx.cs**: incluye código para operaciones CRUD (agregar, editar, eliminar) conectándose a SQL Server.

### **5. Implementación del Cierre de Sesión**

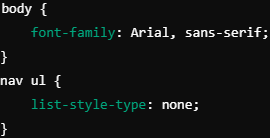
Agrega un botón de "Cerrar sesión" en **Site.Master** y escribe el código para eliminar la sesión:



### **6. Estilos y Diseño**

Crea un archivo CSS para darle estilo a los formularios, tablas y el menú de navegación.

Ejemplo básico:



### **7. Pruebas y Verificación**

1. Inicia el sistema y prueba iniciar sesión con las credenciales configuradas en la tabla Usuarios.
2. Verifica cada operación CRUD en las secciones de empleados, teléfonos, direcciones y puestos.
3. Asegúrate de que el sistema redirija correctamente a Login.aspx si no hay una sesión activa.