



INTRODUCCIÓN

En el siguiente manual se presentan los módulos y su funcionamiento del área de seguridad, se presentan además ilustraciones de cómo es cada módulo para mayor facilidad.

El presente módulo de seguridad está realizado para brindarle una mejor experiencia donde sus datos e información permanezca seguros.



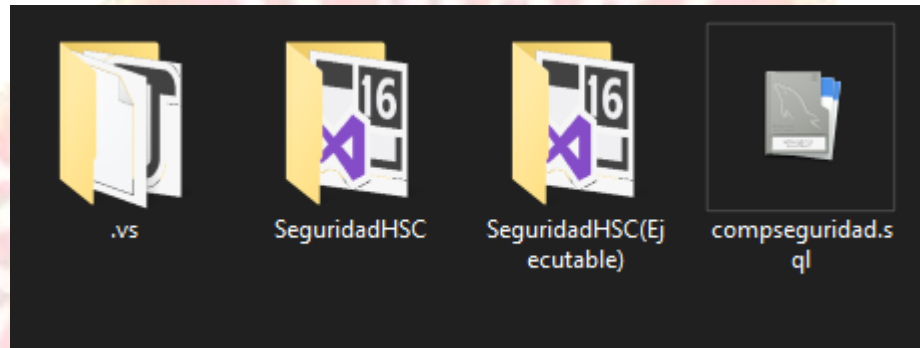
INDICE

Contenido

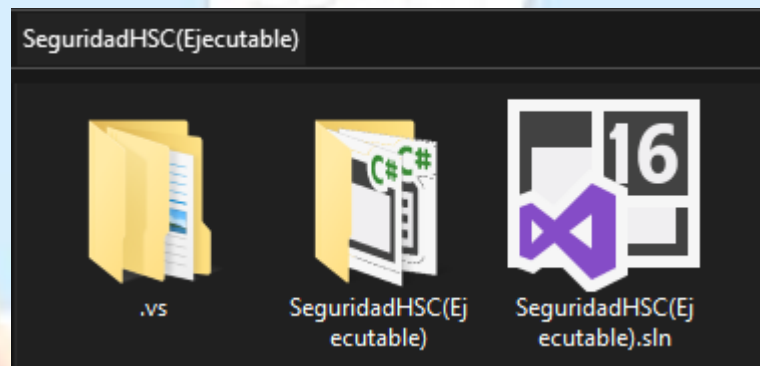
Manual de seguridad.....	1
INSTALACIÓN O USO DEL COMPONENTE.....	4
CONEXIÓN A BASE DE DATOS.....	5
OBJETIVOS.....	6
NORMAS, POLITICAS Y PROCEDIMIENTOS.....	7
ESPECIFICACION DE REQUISITOS.....	7
VISTA FUNCIONAL.....	8
CASOS DE USO.....	9
MODELO LOGICO DE DATOS	10

INSTALACIÓN O USO DEL COMPONENTE

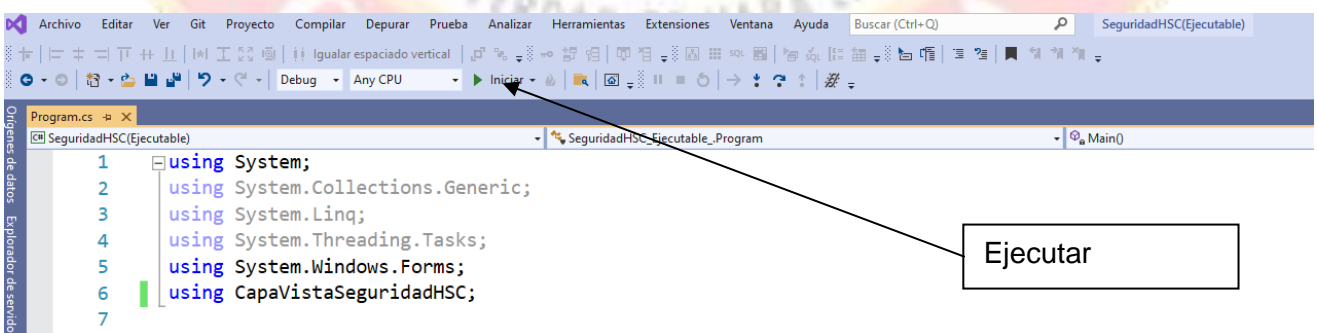
En la carpeta contenedora se encuentran dos proyectos uno donde están los DLL y el otro se encuentra el ejecutable, también el respectivo script de la base de datos del componente.



Para iniciar la ejecución se inicia el proyecto **SeguridadHSC(ejecutable)**.

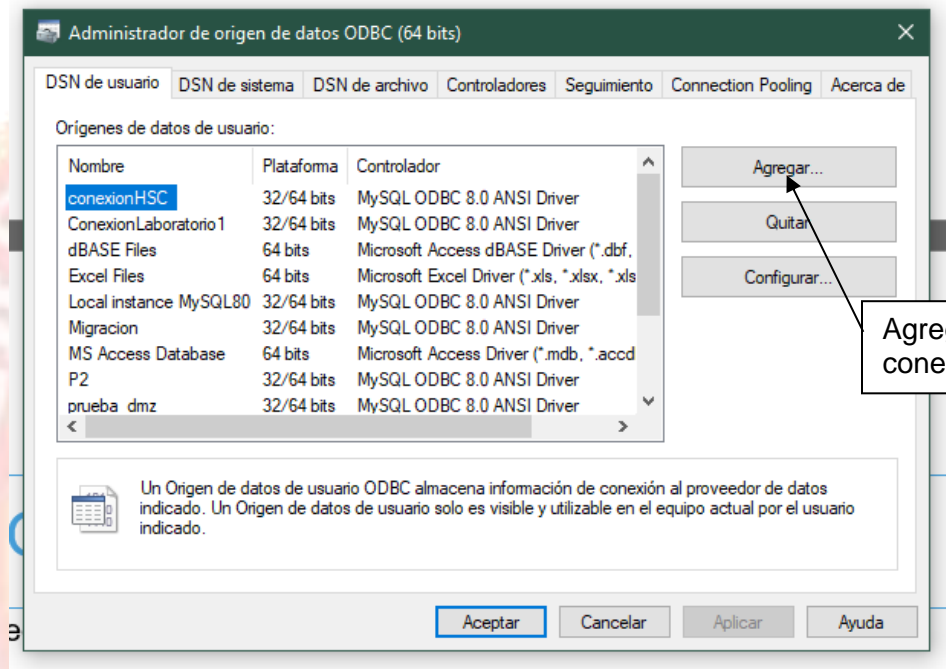


Donde procedemos a su ejecución.

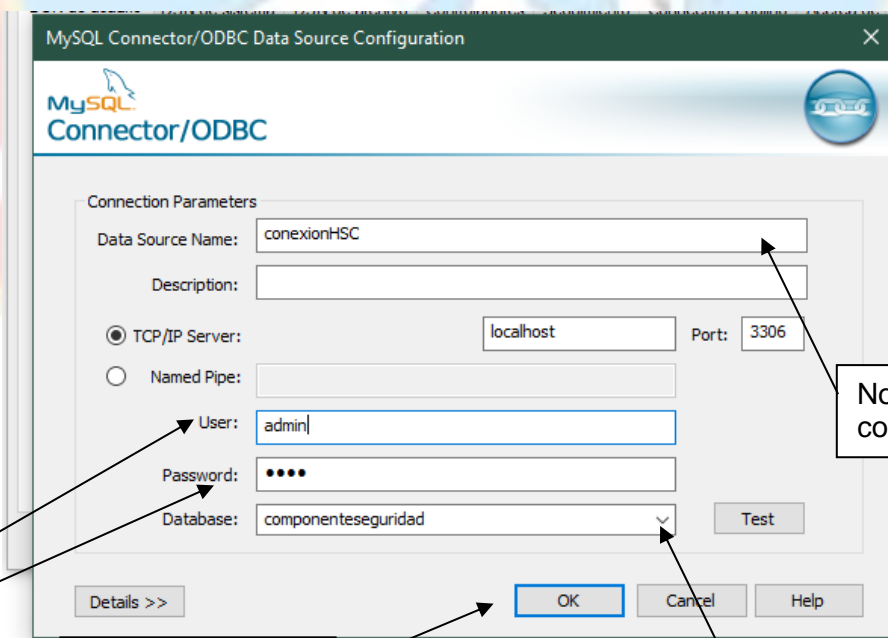


CONEXIÓN A BASE DE DATOS

Para conectar con la base de datos se realiza una conexión por ODBC con el nombre **conexionHSC**.



Agregar nueva conexión



Nombre de la conexión

Usuario y contraseña de la base de datos

Agregar la conexión creada

Nombre de la base de datos

OBJETIVOS

General

Proporcionar al usuario la seguridad que necesita un sistema de gestión de servicios de hotelería, en este caso, el Hotel San Carlos.

Mantener al usuario protegido y a la vez brindarle comodidad en la realización de sus tareas a través de un diseño moderno, fresco e intuitivo.

Específicos

- Implementar encriptado y desencriptado de contraseñas para los usuarios.
- Registrar la actividad de los usuarios dentro del sistema a través de una bitácora.
- Ingreso de usuarios que poseen un perfil, a través del cual se les concede permisos.
- Los permisos se conceden tanto a nivel de usuario, como de perfil para cada aplicación dentro del sistema.

NORMAS, POLITICAS Y PROCEDIMIENTOS

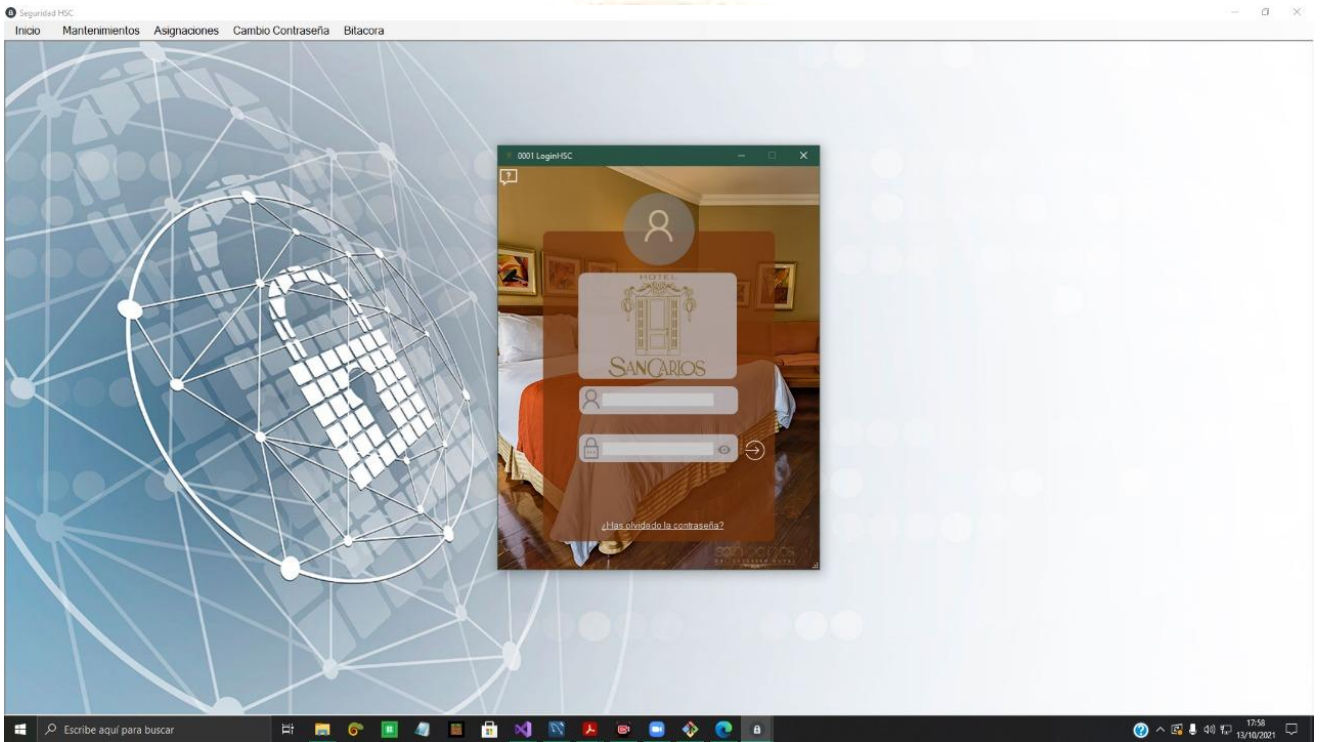
El sistema de seguridad desarrollado para HSC se basa en ISO 27001, la cual es una norma internacional que permite el aseguramiento, la confidencialidad e integridad de los datos y de la información, así como de los sistemas que la presentan.

Complementamos lo anterior con las buenas prácticas establecidas por la norma 27002, de la cual su principal objetivo es establecer directrices y principios generales para iniciar, implementar, mantener y mejorar la gestión de la seguridad de la información dentro de una organización. También incluye la selección, implementación y administración de controles teniendo en cuenta los entornos de riesgo encontrados en la empresa.

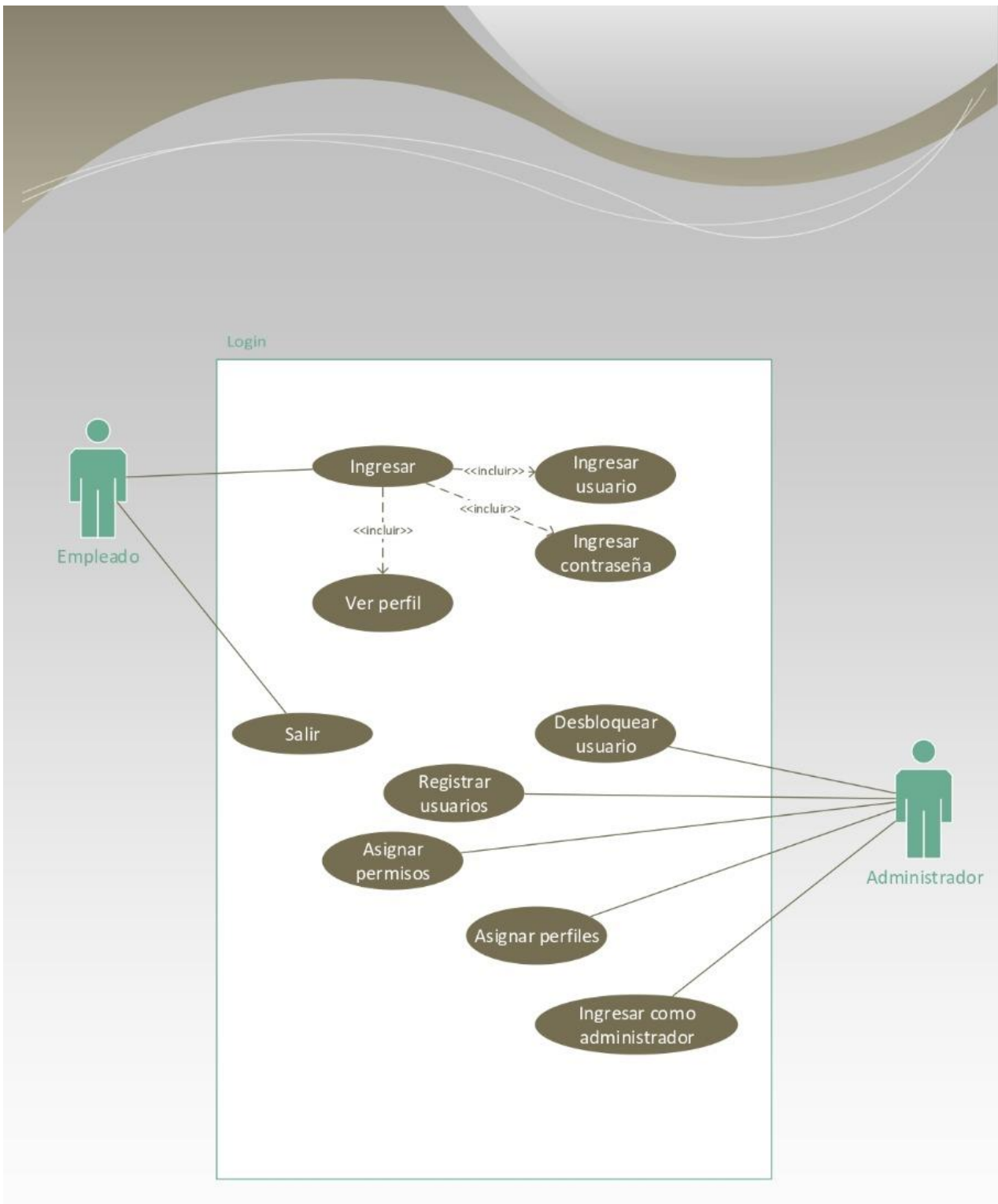
ESPECIFICACION DE REQUISITOS

- El sistema debe contar con un login que permita la recuperación de contraseña
- El sistema debe tener un registro de bitácora.
- Se debe hacer encriptado y desencriptado de contraseñas
- Se debe crear un procedimiento de asignación de permisos al usuario o perfil sobre las aplicaciones.
- Se debe crear una función que comparta con el navegador los permisos de usuario y perfil sobre las aplicaciones.
- Se debe crear una función que permita bloquear botones

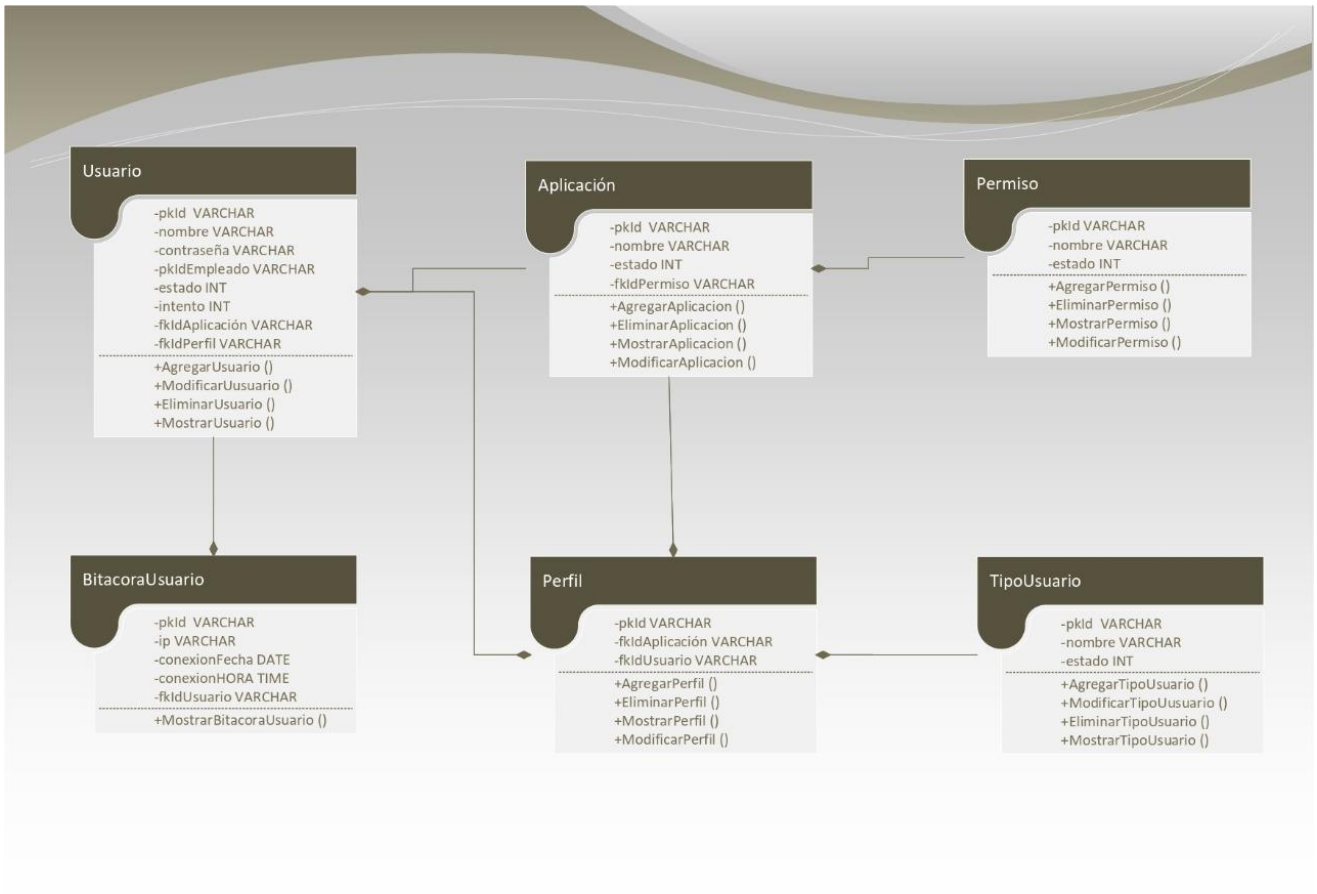
VISTA FUNCIONAL



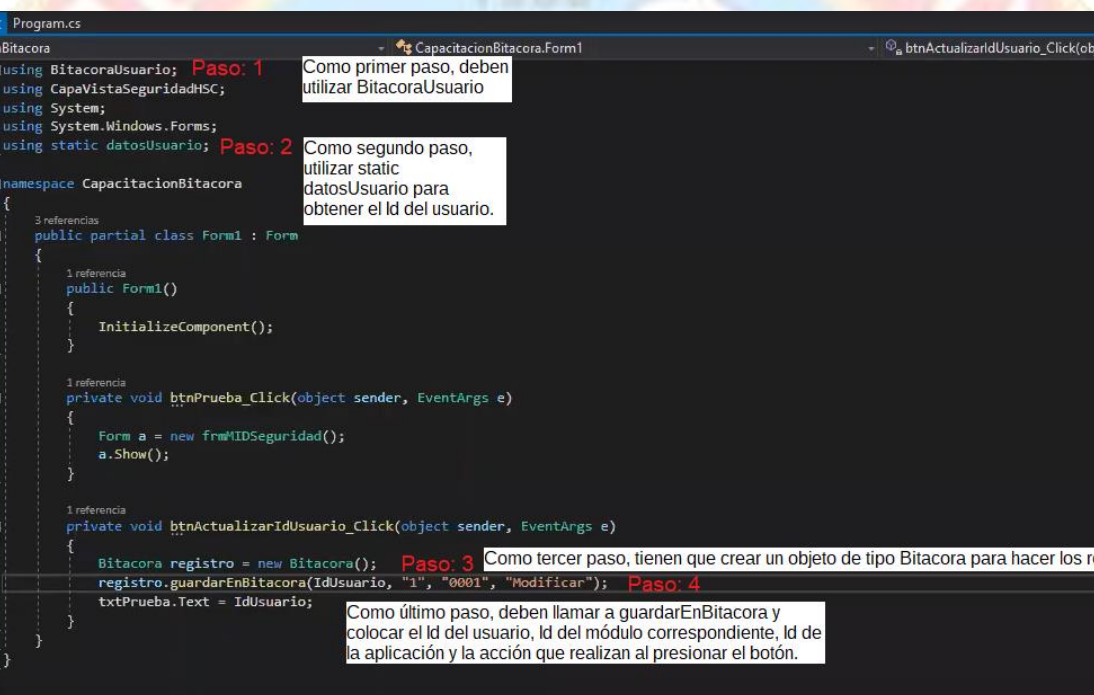
CASOS DE USO



MODELO LOGICO DE DATOS



Bitácora



```
Form1.cs Program.cs
CapitacionBitacora CapitacionBitacora.Form1 btnActualizarIdUsuario_Click(object sender, EventArgs e)

1 using BitacoraUsuario; Paso: 1 Como primer paso, deben
2 using CapaVistaSeguridadHSC; utilizar BitacoraUsuario
3 using System;
4 using System.Windows.Forms;
5 using static datosUsuario; Paso: 2 Como segundo paso,
6                                     utilizar static
7                                     datosUsuario para
8                                     obtener el Id del usuario.
9 namespace CapitacionBitacora
10 {
11     3 referencias
12     public partial class Form1 : Form
13     {
14         1 referencia
15         public Form1()
16         {
17             InitializeComponent();
18         }
19
20         1 referencia
21         private void btnPrueba_Click(object sender, EventArgs e)
22         {
23             Form a = new frmMIDSeguridad();
24             a.Show();
25         }
26
27         1 referencia
28         private void btnActualizarIdUsuario_Click(object sender, EventArgs e)
29         {
30             Bitacora registro = new Bitacora(); Paso: 3 Como tercer paso, tienen que crear un objeto de tipo Bitacora para hacer los registros
31             registro.guardarEnBitacora(IdUsuario, "1", "0001", "Modificar"); Paso: 4
32             txtPrueba.Text = IdUsuario; Como último paso, deben llamar a guardarEnBitacora y
33                                     colocar el Id del usuario, Id del módulo correspondiente, Id de
34                                     la aplicación y la acción que realizan al presionar el botón.
35         }
36     }
37 }
```