

MEMORIA DEL PROYECTO FINAL DI

League Admin



league admin

MANUEL GÓMEZ FURONES
(2º DAM)

ÍNDICE

| | |
|--|----|
| OBJETIVOS DE LA APLICACIÓN | 3 |
| PÚBLICO OBJETIVO | 3 |
| DISEÑO DE LA APLICACIÓN | 3 |
| PANTALLA DE CARGA/SLASH SCREEN | 7 |
| PANTALLA DE INICIO | 7 |
| PANTALLA PRINCIPAL | 8 |
| CONSULTAR..... | 12 |
| MODIFICAR/BORRAR | 13 |
| AÑADIR | 14 |
| INFORMES | 16 |
| AYUDA | 18 |
| FUNCIONAMIENTO DE LA APLICACIÓN..... | 20 |
| PANTALLA DE INICIO | 20 |
| PANTALLA PRINCIPAL | 20 |
| CONSULTAR..... | 21 |
| MODIFICAR/BORRAR | 21 |
| AÑADIR | 21 |
| INFORMES | 21 |
| DIFICULTADES QUE HAN SURGIDO EN EL DESARROLLO DE LA APLICACIÓN..... | 22 |
| POSIBLES AMPLIACIONES DE LA APLICACIÓN | 22 |
| CONCLUSIÓN..... | 23 |
| REFERENCIAS CONSULTADAS | 23 |

OBJETIVOS DE LA APLICACIÓN

League Admin es una aplicación concebida para poder gestionar y administrar una liga deportiva de la manera más sencilla

Sus objetivos son los siguientes:

- Implementar una interfaz de usuario lo más sencilla e intuitiva para que hasta el usuario más principiante no tenga problemas al usar la app.
- Usar Controles de usuario propios, así como paquetes externos a la librería principal de .NET en Visual Studio para que la app se distinga del resto.
- Aprender y simular el proceso de creación de una aplicación real para la asignatura Desarrollo de Interfaces.

PÚBLICO OBJETIVO

La app está dirigida principalmente a personas que dirijan ligas deportivas, así como personas interesadas en el funcionamiento de estas por mero entretenimiento.

DISEÑO DE LA APLICACIÓN

El diseño de la aplicación empieza por la base de datos, que es local ya que el programa está pensado usarse desde un mismo equipo.

Las plataformas elegidas para su creación es *SQL Server* y *Microsoft SQL Server Management Studio 18* y la Base de datos la constituyen 3 tablas las cuales detallaré a continuación.

EQUIPOS

- **ID varchar(3)**
 - Nombre Completo varchar(50)
 - Nombre varchar (40)
 - Ciudad varchar(29)
 - → Estadio (int) – Referencia a ESTADIOS
 - AñoFundacion int
 - NumSocios int

ARBITROS

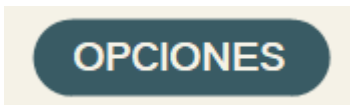
- **ID int**
 - APELLIDOS Varchar(30)
 - Nombre varchar(20)
 - FechaNacimiento Timestamp
 - Colegio Varchar(20)

ESTADIOS

- **ID int**
 - Nombre Varchar(30)
 - Capacidad int
 - Direccion Varchar(100)
 - FechaConstruccion Timestamp

En cuanto al diseño de la interfaz, se detallan ahora los formularios y sus controles personalizados:

En cuanto a controles personalizados, todos los botones en la aplicación lo son. Un ejemplo de estos es el siguiente:



A continuación mostraré el código de este control personalizado

```
using System;
using System.Collections.Generic;
using System.Linq;
using System.Text;
using System.Threading.Tasks;
using System.Windows.Forms;
using System.Drawing;
using System.Drawing.Drawing2D;
using System.ComponentModel;

namespace LeagueAdminManuelGomez
{
    public class BotonPersonalizado : Button
    {
        private int borderSize = 0;
        private int borderRadius = 20;
        private Color borderColor = Color.PaleVioletRed;

        //Properties
        [Category("RJ Code Advance")]
        public int BorderSize
        {
            get { return borderSize; }
            set
            {
                borderSize = value;
                this.Invalidate();
            }
        }
        [Category("RJ Code Advance")]
        public int BorderRadius
        {
            get { return borderRadius; }
            set
            {
                borderRadius = value;
                this.Invalidate();
            }
        }
        [Category("RJ Code Advance")]
        public Color BorderColor
        {
            get { return borderColor; }
            set
            {
                borderColor = value;
                this.Invalidate();
            }
        }
    }
}
```

```
        get { return borderColor; }
        set
        {
            borderColor = value;
            this.Invalidate();
        }
    }
    [Category("RJ Code Advance")]
    public Color BackgroundColor
    {
        get { return this.BackColor; }
        set { this.BackColor = value; }
    }
    [Category("RJ Code Advance")]
    public Color TextColor
    {
        get { return this.ForeColor; }
        set { this.ForeColor = value; }
    }

    //Constructor
    public BotonPersonalizado()
    {
        this.FlatStyle = FlatStyle.Flat;
        this.FlatAppearance.BorderSize = 0;
        this.Size = new Size(150, 40);
        this.BackColor = Color.MediumSlateBlue;
        this.ForeColor = Color.White;
        this.Resize += new EventHandler(Button_Resize);
    }
    private void Button_Resize(object sender, EventArgs e)
    {
        if (borderRadius > this.Height)
            borderRadius = this.Height;
    }

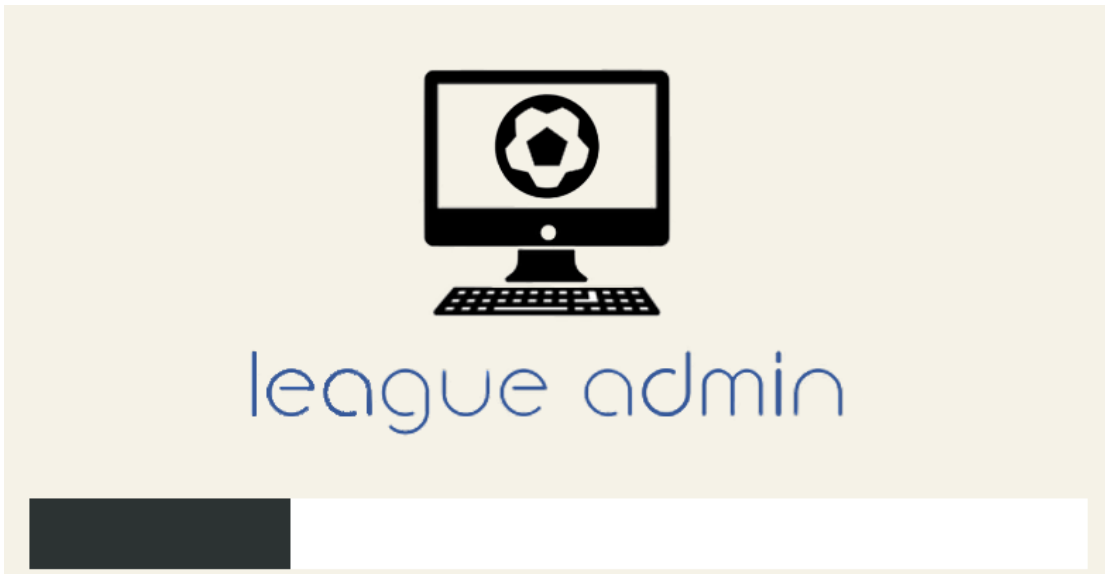
    private GraphicsPath GetFigurePath(Rectangle rect, float radius)
    {
        GraphicsPath path = new GraphicsPath();
        float curveSize = radius * 2F;
        path.StartFigure();
        path.AddArc(rect.X, rect.Y, curveSize, curveSize, 180, 90);
        path.AddArc(rect.Right - curveSize, rect.Y, curveSize, curveSize, 270,
90);
        path.AddArc(rect.Right - curveSize, rect.Bottom - curveSize, curveSize,
curveSize, 0, 90);
        path.AddArc(rect.X, rect.Bottom - curveSize, curveSize, curveSize, 90,
90);
        path.CloseFigure();
        return path;
    }

    protected override void OnPaint(PaintEventArgs pevent)
    {
        base.OnPaint(pevent);
        Rectangle rectSurface = this.ClientRectangle;
        Rectangle rectBorder = Rectangle.Inflate(rectSurface, -borderSize, -
borderSize);
        int smoothSize = 2;
        if (borderSize > 0)
            smoothSize = borderSize;
        if (borderRadius > 2) //Rounded button
        {
            using (GraphicsPath pathSurface = GetFigurePath(rectSurface,
borderRadius))
```

```
        using (GraphicsPath pathBorder = GetFigurePath(rectBorder,
borderRadius - borderSize))
        using (Pen penSurface = new Pen(this.Parent.BackColor, smoothSize))
        using (Pen penBorder = new Pen(borderColor, borderSize))
        {
            pevent.Graphics.SmoothingMode = SmoothingMode.AntiAlias;
            //Button surface
            this.Region = new Region(pathSurface);
            //Draw surface border for HD result
            pevent.Graphics.DrawPath(penSurface, pathSurface);
            //Button border
            if (borderSize >= 1)
                //Draw control border
                pevent.Graphics.DrawPath(penBorder, pathBorder);
        }
    }
    else //Normal button
    {
        pevent.Graphics.SmoothingMode = SmoothingMode.None;
        //Button surface
        this.Region = new Region(rectSurface);
        //Button border
        if (borderSize >= 1)
        {
            using (Pen penBorder = new Pen(borderColor, borderSize))
            {
                penBorder.Alignment = PenAlignment.Inset;
                pevent.Graphics.DrawRectangle(penBorder, 0, 0, this.Width -
1, this.Height - 1);
            }
        }
    }
}

protected override void OnHandleCreated(EventArgs e)
{
    base.OnHandleCreated(e);
    this.Parent.BackColorChanged += new
EventHandler(Container_BackColorChanged);
}
private void Container_BackColorChanged(object sender, EventArgs e)
{
    this.Invalidate();
}
}
```

PANTALLA DE CARGA/SLASH SCREEN



En esta pantalla de carga nos encontramos el logo de la aplicación así como la barra de progreso que al finalizar nos nos mostrará la pantalla de inicio.

La barra de progreso realmente está formada por dos paneles. Uno blanco que ocupa la totalidad de la barra y otro Negro que aumenta su anchura cada vez que se da el evento "tick" de un timer.

PANTALLA DE INICIO

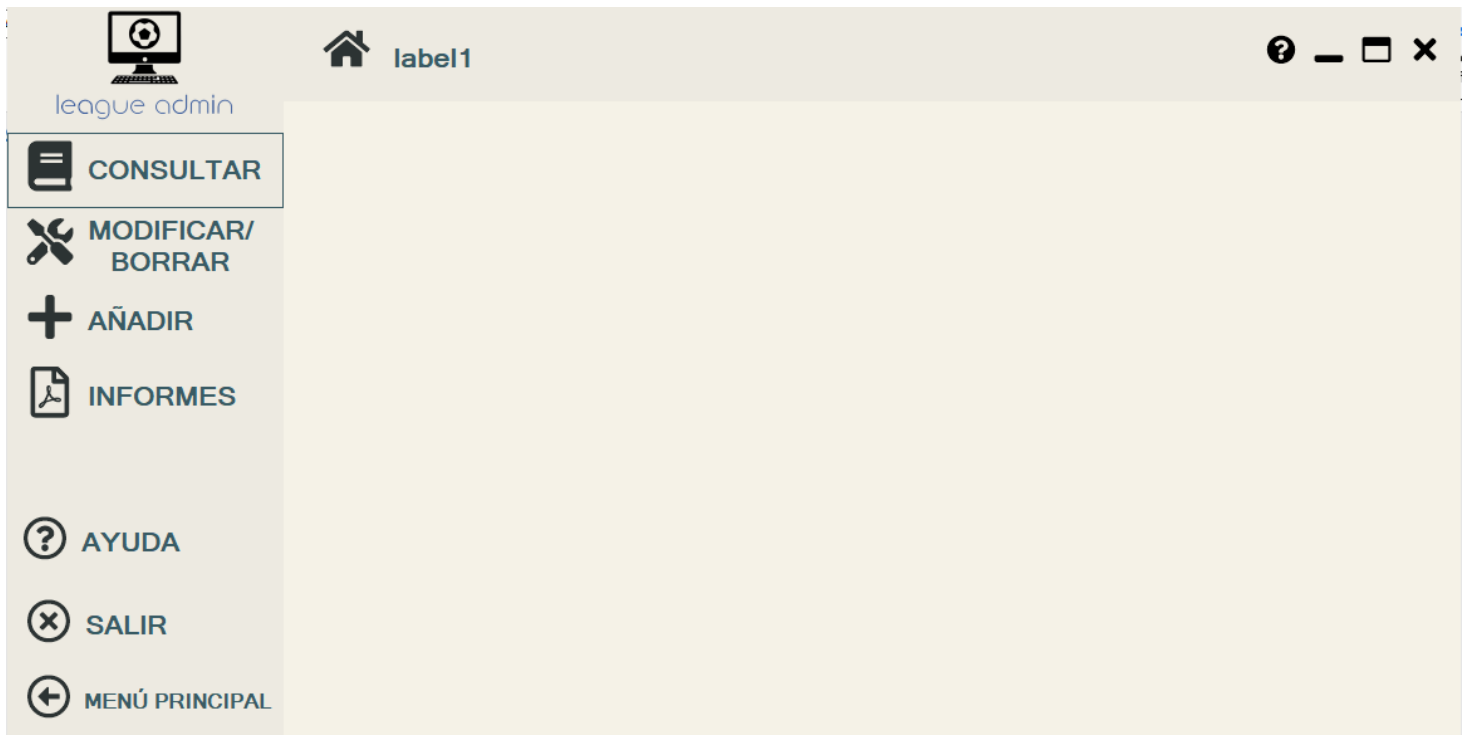


En esta pantalla nos encontramos de nuevo el logo de la aplicación así como dos botones personalizados.

El primero, SALIR, nos permite salir del programa sin haber iniciado el formulario principal.

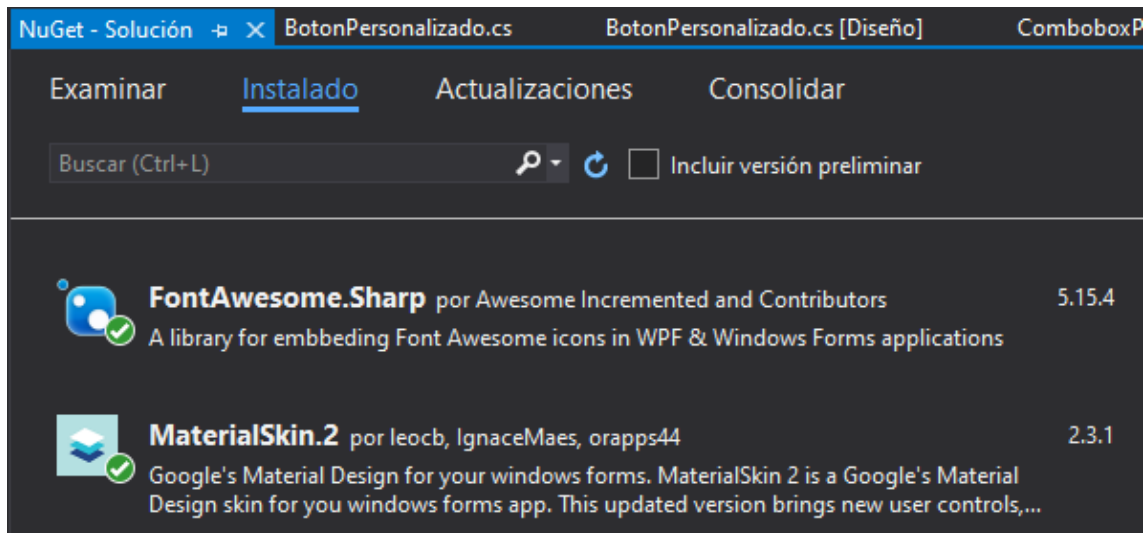
El segundo, INICIAR, nos permite acceder al formulario principal

PANTALLA PRINCIPAL



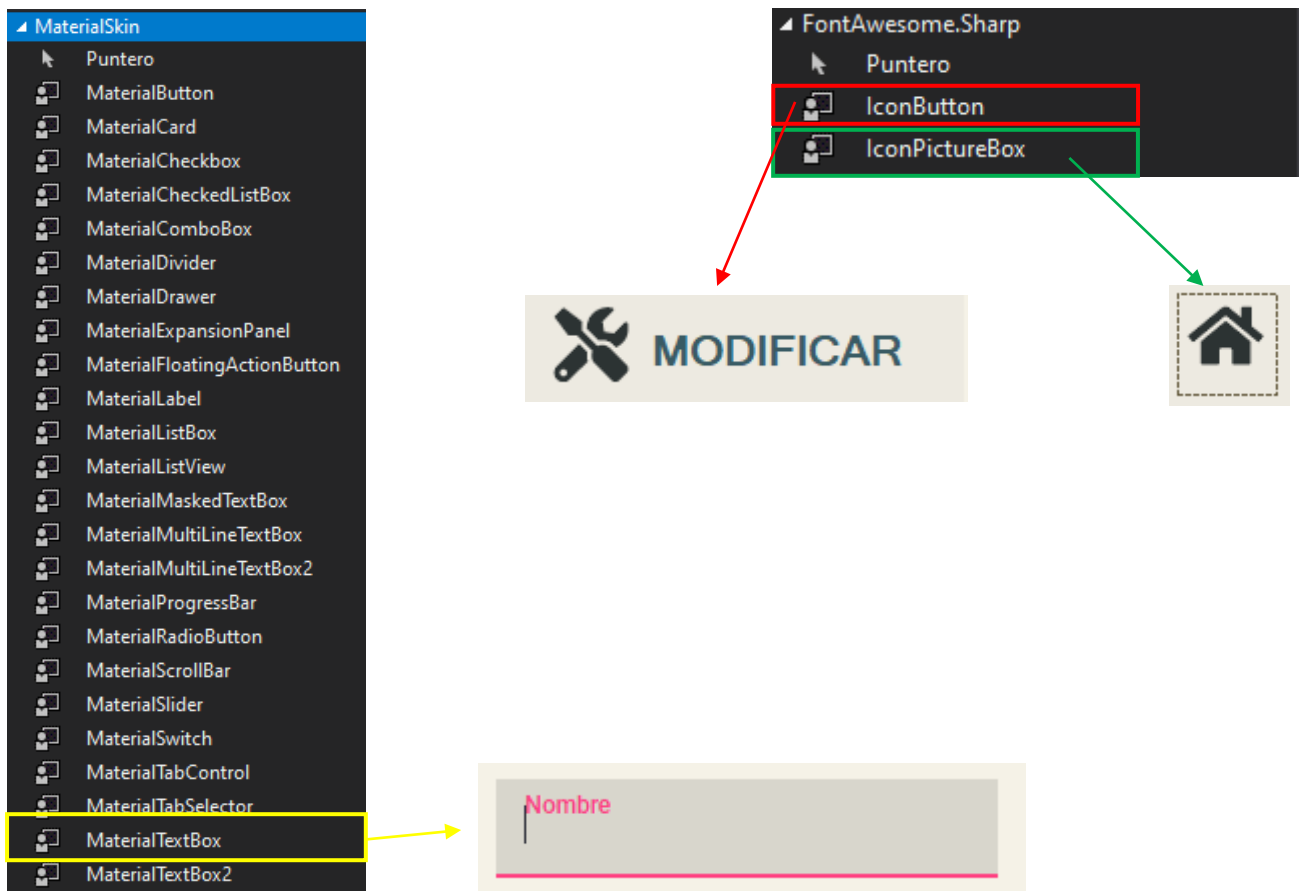
En este formulario se encuentra el grueso de funciones que se pueden realizar con la aplicación.

Lo primero que hay que explicar es que para hacer esta interfaz de una forma diferente y mejor con controles diferentes a los que vienen por defecto, he descargado dos librerías externas NuGet (*FontAwesome.Sharp* y *MaterialSkin.2*)



Al instalar estas bibliotecas se añadieron todos los controles que estas incluyen.

Los controles que he usado en el proyecto son los siguientes:



MaterialtextBox: Combobox con mejor estilo y más personalizable

IconButton: Botón al que se le fija un icono

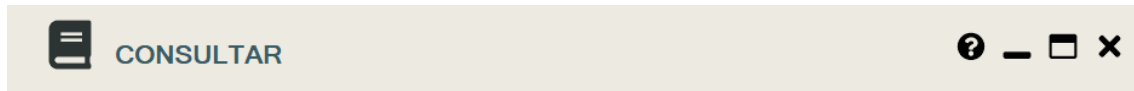
IconPictureBox: Botón al cual se le puede añadir un icono y posicionar estos de varias maneras diferentes.

Los iconos del programa son gratuitos y vienen incluidos en estos paquetes.

El enlace para ver los diferentes iconos es este: <https://fontawesome.com/search?m=free>

La pantalla principal la forman 3 paneles

❖ Panel Superior



En él se encuentra el icono de la función que se está usando, la label con dicho nombre y los botones de Ayuda, Minimizar, Maximizar y Cerrar.

❖ Panel Lateral Izquierdo/de Menú



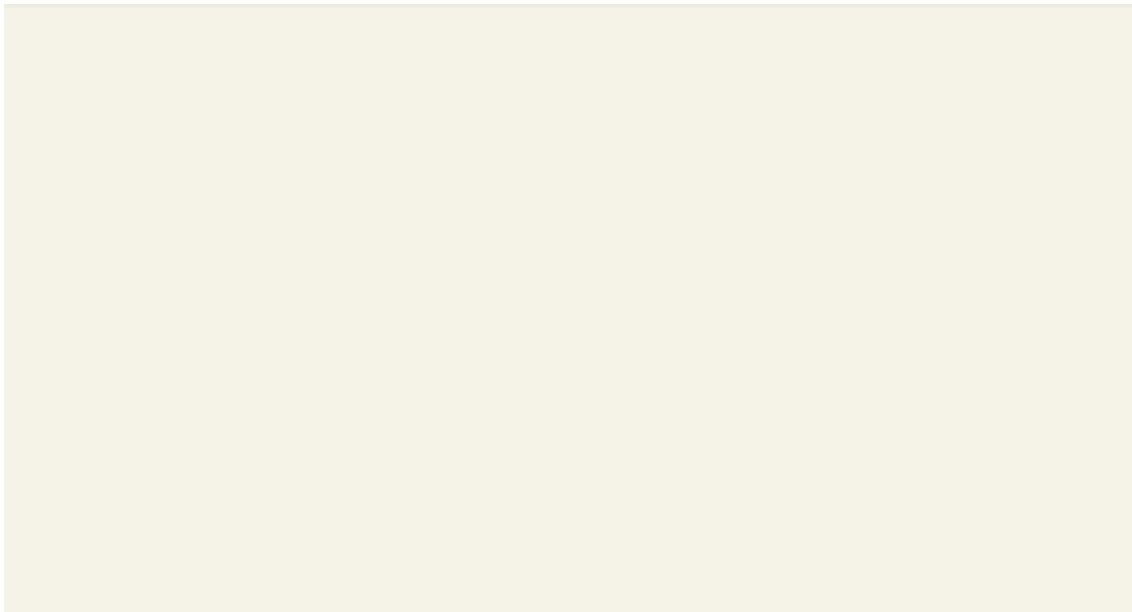
En el se encuentran los IconButtons de las diferentes funciones que ofrece el programa (CONSULTAR, MODIFICAR, AÑADIR e INFORMES) así como AYUDA, SALIR y MENÚ PRINCIPAL

Al pulsar cualquier botón no solo se actualiza el control de usuario en el panel principal. También cambia el logo y nombre en el panel superior y añade un panel pequeño a la izquierda del IconButton pulsado además de oscurecerlo gracias al método *activarBotón* (lo podemos comprobar en el botón “CONSULTAR”).

Código del botón *ActivarBoton()*:

```
private void ActivarBoton(Object senderBtn, Color color, IconChar icono, String texto)
{
    if(senderBtn != null)
    {
        desabilitarBoton();
        //Botón principal
        btnActual = (IconButton)senderBtn;
        btnActual.BackColor = Color.Silver;
        btnActual.ForeColor = color;
        btnActual.TextAlign = ContentAlignment.MiddleCenter;
        btnActual.IconColor = color;
        btnActual.TextImageRelation = TextImageRelation.TextBeforeImage;
        btnActual.ImageAlign = ContentAlignment.MiddleRight;
        //botón que aparece al pulsar el botón principal
        leftBorderBtn.BackColor = color;
        leftBorderBtn.Location = new Point(0, btnActual.Location.Y);
        leftBorderBtn.Visible = true;
        leftBorderBtn.BringToFront();
        //El menú superior mostrará el icono y el label del botón presionado
        iconoMenuSuperior.IconChar = icono;
        iconoMenuSuperior.Size = new Size(50, 50);
        lblMenuSuperior.Text = texto;
    }
}
```

❖ Panel Central/Principal



Este panel cambia de control dependiendo del botón que pulsemos en la barra lateral izquierda.

La interfaz de las diferentes funciones las veremos a continuación

CONSULTAR

Seleccione la tabla:

Palabra Clave

En esta interfaz podemos apreciar un Combobox, un MaterialTextBox y un DataGridView.

Al pulsar en uno de los elementos del combobox, que hacen referencia a las tablas de la base de datos, se mostrarán los nombres de las columnas de la tabla seleccionada.

Al introducir caracteres en el MaterialTextBox, el DataGridView se rellenará con los registros que coincidan con la cadena introducida.

He aquí un ejemplo al añadir la cadena “a” habiendo seleccionado seleccionando la tabla *equipos* en el ComboBox:

Seleccione la tabla:

Palabra Clave

| | ID | NombreCompleto | Nombre | Ciudad | Estadio | AñoFundacion | Socios |
|---|-----|----------------------|-------------------|--------------------|---------|--------------|--------|
| ▶ | ALA | Deportivo Alavés... | Alavés | Vitoria/Gasteiz | 1 | 1921 | 17600 |
| | ALC | Agrupación Depo... | Alcorcón | Alcorcón | 21 | 1971 | 2500 |
| | ALM | Unión Deportiva ... | Almería | Almería | 22 | 1989 | 9000 |
| | AMO | Sociedad Deporti... | Amorebieta | Amorebieta-Echa... | 39 | 1925 | 1000 |
| | ATB | Athletic Club Bilbao | Athletic | Bilbao | 2 | 1898 | 44560 |
| | ATM | Cub Atlético de ... | Atlético de Madrd | Madrid | 3 | 1903 | 128249 |
| | BAR | Futbol Club Barc... | Barcelona | Barcelona | 4 | 1899 | 144500 |
| | CAL | CD Calahorra | Calahorra | Calahorra | 57 | 1923 | 2000 |
| | CAR | Fútbol club Carta... | Cartagena | Cartagena | 24 | 1995 | 8200 |
| | CEL | Real club Celta S... | Celta | Vigo | 7 | 1923 | 16801 |
| | EIB | Sociedad Deporti... | Eibar | Eibar | 25 | 1940 | 5799 |
| | ESP | Real club Deporti... | Espanyol | Barcelona | 9 | 1900 | 22970 |
| | FUE | Club de Fútbol Fu... | Fuenlabrada | Fuenlabrada | 26 | 1975 | 5000 |
| | GET | Getafe club de F... | Getafe | Getafe | 10 | 1923 | 13000 |

MODIFICAR/BORRAR

ELIGE LA TABLA QUE QUIERAS MODIFICAR

GUARDAR **BORRAR**

En esta interfaz podemos ver un ComboBox, dos Botones personalizados y un DataGridView.

Al seleccionar uno de los elementos del combobox, que hacen referencia a las tablas de la base de datos, se muestran todos los registros de dicha tabla en el DataGridView.

Aquí muestro un ejemplo seleccionando la tabla *Arbitros*:

ELIGE LA TABLA QUE QUIERAS MODIFICAR

GUARDAR **BORRAR**

| | ID | Apellidos | Nombre | FechaNacimiento | Colegio |
|---|----|--------------------|----------------|-----------------|--------------------|
| ▶ | 1 | Alberola Rojas | Javier | 22/06/1991 | Castilla la Mancha |
| | 2 | Cordero Vega | Adrián | 15/06/1984 | Cantabria |
| | 3 | Cuadra Fernández | Guillermo | 25/04/1984 | Islas Baleares |
| | 4 | De Burgos Beng... | Ricardo Peña | 16/03/1986 | Vizcaya |
| | 5 | Del Cero Grande | Carlos | 13/03/1976 | Madrid |
| | 6 | Díaz de Mera Es... | Isidro | 24/08/1989 | Castilla la Mancha |
| | 7 | Figueroa Vázquez | Jorge | 05/02/1980 | Andalucía |
| | 9 | González Fuertes | Pablo | 08/06/1980 | Asturias |
| | 10 | Hernández Hem... | Alejandro José | 10/11/1982 | Canarias |
| | 11 | Jaime Latre | Santiago | 17/06/1979 | Aragón |
| | 12 | Martínez Munuera | Juan | 13/07/1982 | Valencia |
| | 13 | Mateu Lahoz | Miguel | 12/03/1977 | Valencia |
| | 14 | Melero López | Mario | 01/01/1900 | |
| | 15 | Munuera Montero | José Luis | 19/05/1983 | Andalucía |
| | 16 | Muñiz Ruiz | Alejandro | 02/07/1979 | Andalucía |
| | 17 | Ortiz Arias | Miguel Ángel | 23/11/1984 | Madrid |

AÑADIR

Seleccione la tabla:

AÑADIR **LIMPIAR CAMPOS**

Aquí se pueden apreciar dos botones personalizados y un ComboBox

Al seleccionar uno de los elementos del ComboBox, que coinciden con las tablas de la base de datos, se muestran varios MaterialTextBox con los nombres de la tabla seleccionada

Las interfaces dependiendo de la tabla seleccionada son las siguientes

ARBITROS

Seleccione la tabla:

AÑADIR **LIMPIAR CAMPOS**

| | |
|----------|---------------------|
| ID (Num) | Apellidos |
| Nombre | Fecha de Nacimiento |
| Colegio | |

ESTADIOS

Seleccione la tabla:

AÑADIR

LIMPIAR CAMPOS

ID (Num)

Nombre

Capacidad

Dirección

Año de Construcción

EQUIPOS

Seleccione la tabla:

AÑADIR

LIMPIAR CAMPOS

ID (Siglas)

Nombre Completo

Nombre

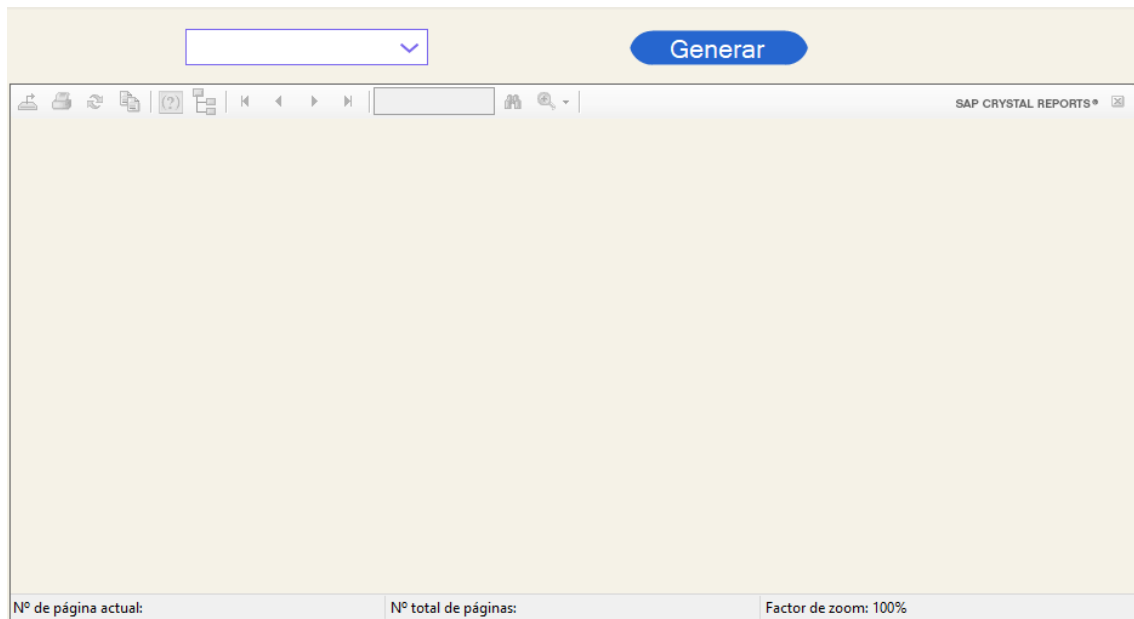
Ciudad

Estadio

Año de Fundación

Nº Socios

INFORMES

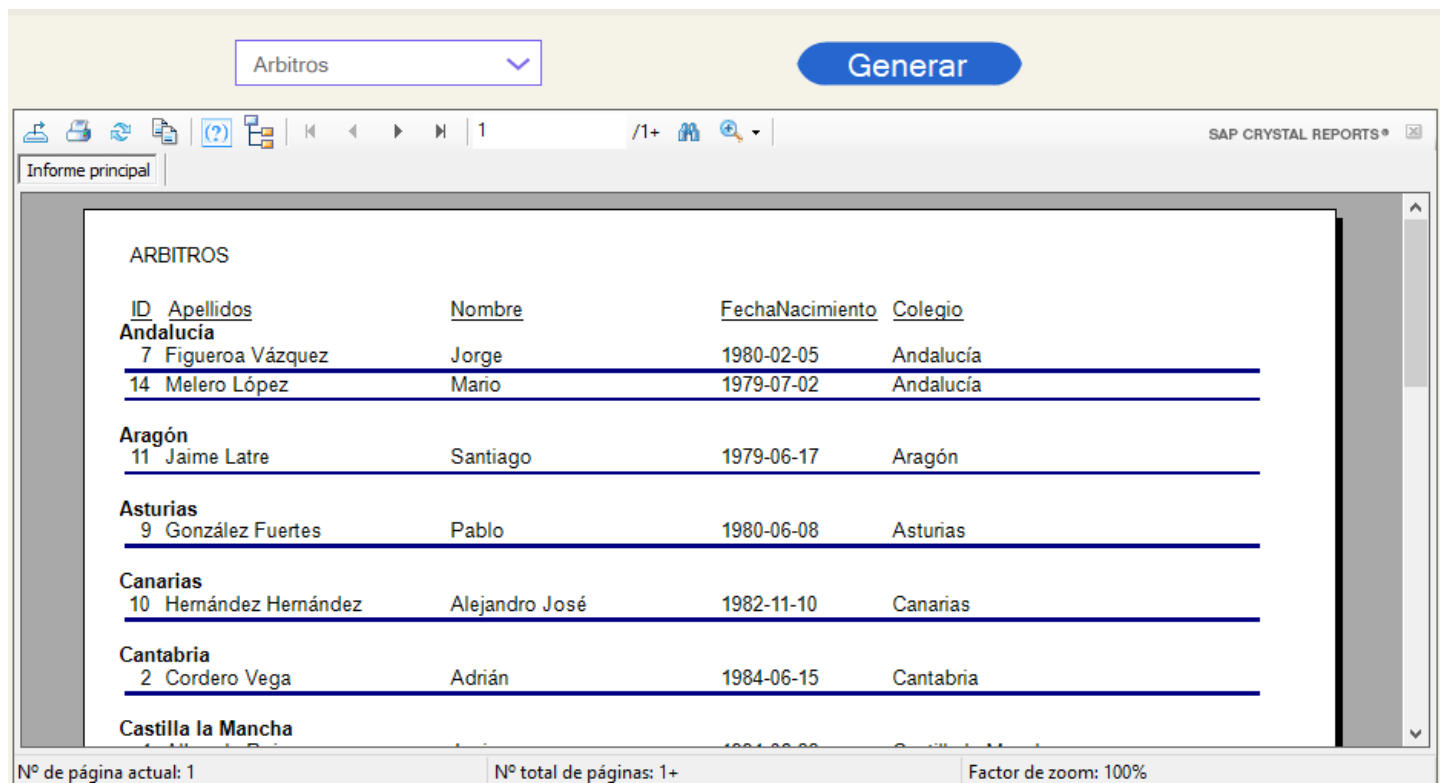


En esta interfaz podemos apreciar un ComboBox personalizado, un Botón personalizado y un CrystalReportViewer.

Al seleccionar un elemento del ComboBox (que coinciden con las tablas de la base de datos) y clicar en el botón *Generar*, se asigna el informe pertinente en el CrystalReportsViewer.

Muestro ahora el informe de cada tabla.

ARBITROS:



The screenshot shows the League Admin interface for generating reports. The dropdown menu is set to 'Arbitros' and the 'Generar' button is visible. The SAP Crystal Reports viewer displays a table of referees. The table has the following structure:

| ID | Apellidos | Nombre | FechaNacimiento | Colegio |
|---------------------------|---------------------|----------------|-----------------|-----------|
| Andalucía | | | | |
| 7 | Figueroa Vázquez | Jorge | 1980-02-05 | Andalucía |
| 14 | Melero López | Mario | 1979-07-02 | Andalucía |
| Aragón | | | | |
| 11 | Jaime Latre | Santiago | 1979-06-17 | Aragón |
| Asturias | | | | |
| 9 | González Fuertes | Pablo | 1980-06-08 | Asturias |
| Canarias | | | | |
| 10 | Hernández Hernández | Alejandro José | 1982-11-10 | Canarias |
| Cantabria | | | | |
| 2 | Cordero Vega | Adrián | 1984-06-15 | Cantabria |
| Castilla la Mancha | | | | |

The status bar at the bottom shows 'Nº de página actual: 1', 'Nº total de páginas: 1+', and 'Factor de zoom: 100%'.

ESTADIOS:

Estadios Generar

SAP CRYSTAL REPORTS®

Informe principal

| ID | Apellidos | Nombre | FechaNacimiento | Colegio |
|---------------------------|---------------------|----------------|-----------------|-----------|
| Andalucía | | | | |
| 7 | Figueroa Vázquez | Jorge | 1980-02-05 | Andalucía |
| 14 | Melero López | Mario | 1979-07-02 | Andalucía |
| Aragón | | | | |
| 11 | Jaime Latre | Santiago | 1979-06-17 | Aragón |
| Asturias | | | | |
| 9 | González Fuertes | Pablo | 1980-06-08 | Asturias |
| Canarias | | | | |
| 10 | Hernández Hernández | Alejandro José | 1982-11-10 | Canarias |
| Cantabria | | | | |
| 2 | Cordero Vega | Adrián | 1984-06-15 | Cantabria |
| Castilla la Mancha | | | | |

Nº de página actual: 1 Nº total de páginas: 1+ Factor de zoom: 100%

EQUIPOS:

Equipos Generar

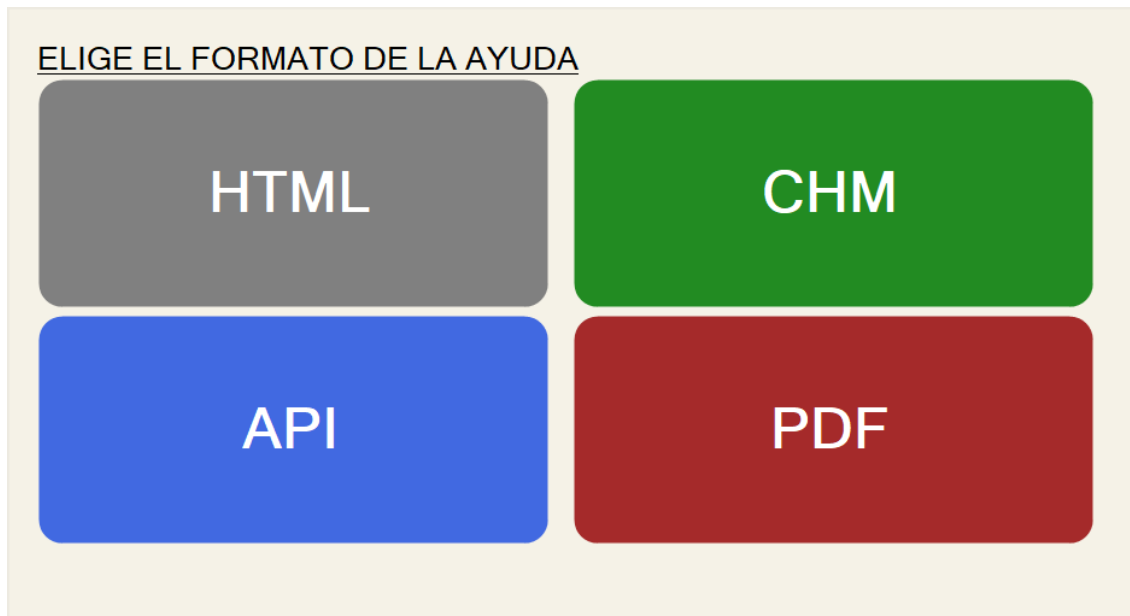
SAP CRYSTAL REPORTS®

Informe principal

| Siglas | Nombre | Estadio | Capacidad | Ciudad | Año de Fundacion | Nº Socios |
|--------|--------------------|---------------------|-----------|-----------------|------------------|-----------|
| ALA | Alavés | Mendizorroza | 19.840 | Vitoria/Gasteiz | 1.921 | 17.600 |
| ALC | Alcorcón | Santo Domingo | 5.100 | Alcorcón | 1.971 | 2.500 |
| AMO | Amorebieta | Uritxe | 3.000 | Amorebieta-Echa | 1.925 | 1.000 |
| ATB | Athletic | San Mamés | 53.332 | Bilbao | 1.898 | 44.560 |
| ATM | Atlético de Madrid | Wanda Metropolitano | 67.703 | Madrid | 1.903 | 128.249 |
| BAR | Barcelona | Camp Nou | 98.772 | Barcelona | 1.899 | 144.500 |
| BET | Betis | Benito Villamarín | 60.721 | Sevilla | 1.907 | 50.373 |
| BUR | Burgos | El Plantío | 12.194 | Burgos | 1.922 | 5.000 |

Nº de página actual: 1 Nº total de páginas: 1+ Factor de zoom: 100%

AYUDA



Esta interfaz se compone de los cuatro Botones personalizados que al pulsarlos abrirán la API, en caso de haber clicado el primer botón, o el manual de ayuda en el formato especificado.

A continuación, muestro las pestañas que se muestran al pulsar cada uno de los botones

HTML

[Skip to main content](#)

Toggle navigation

LeagueAdminHelp

- [Contents](#)
- [Index](#)
- [Search](#)
- [Introducción](#)
 - [Índice de Contenido](#)
- [Manual de Uso](#)
 - [Pantalla de Carga](#)
 - [Pantalla de Inicio](#)
 - [pantalla Principal](#)
 - [Consultar](#)
 - [Modificar](#)
 - [Añadir](#)
 - [Informes](#)
 - [Ayuda](#)
- [Sobre el Creador](#)

Search... 

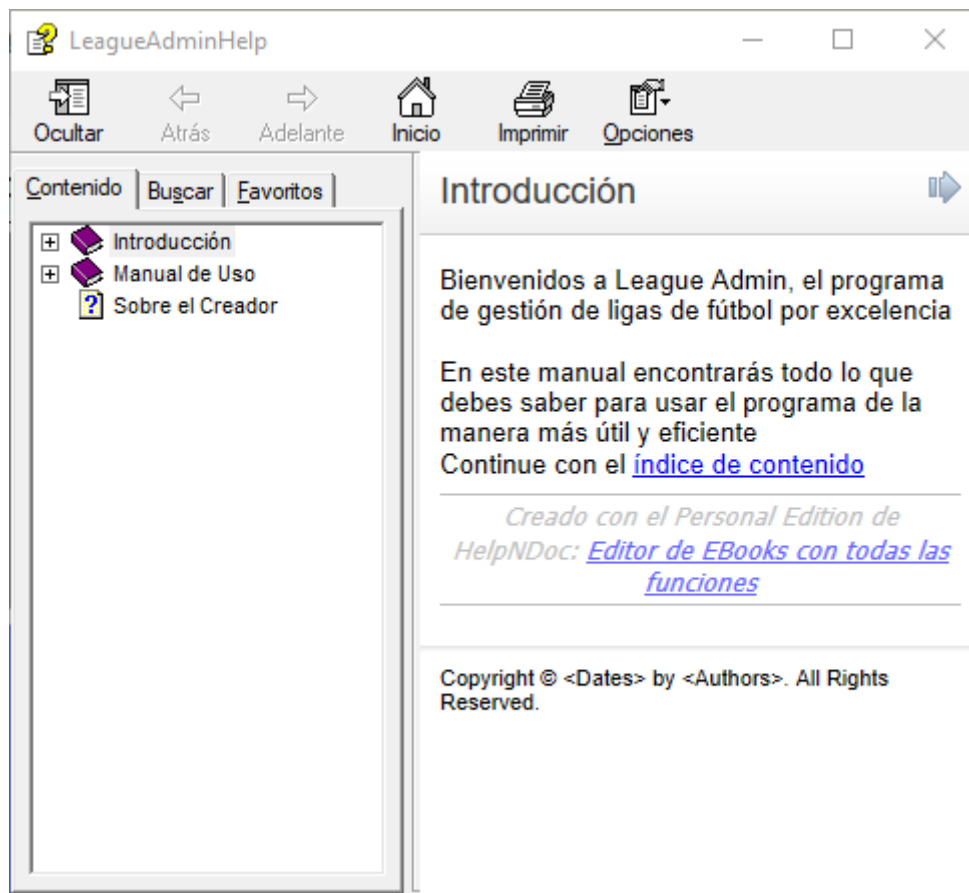
Introducción

Bienvenidos a League Admin, el programa de gestión de ligas de fútbol por excelencia

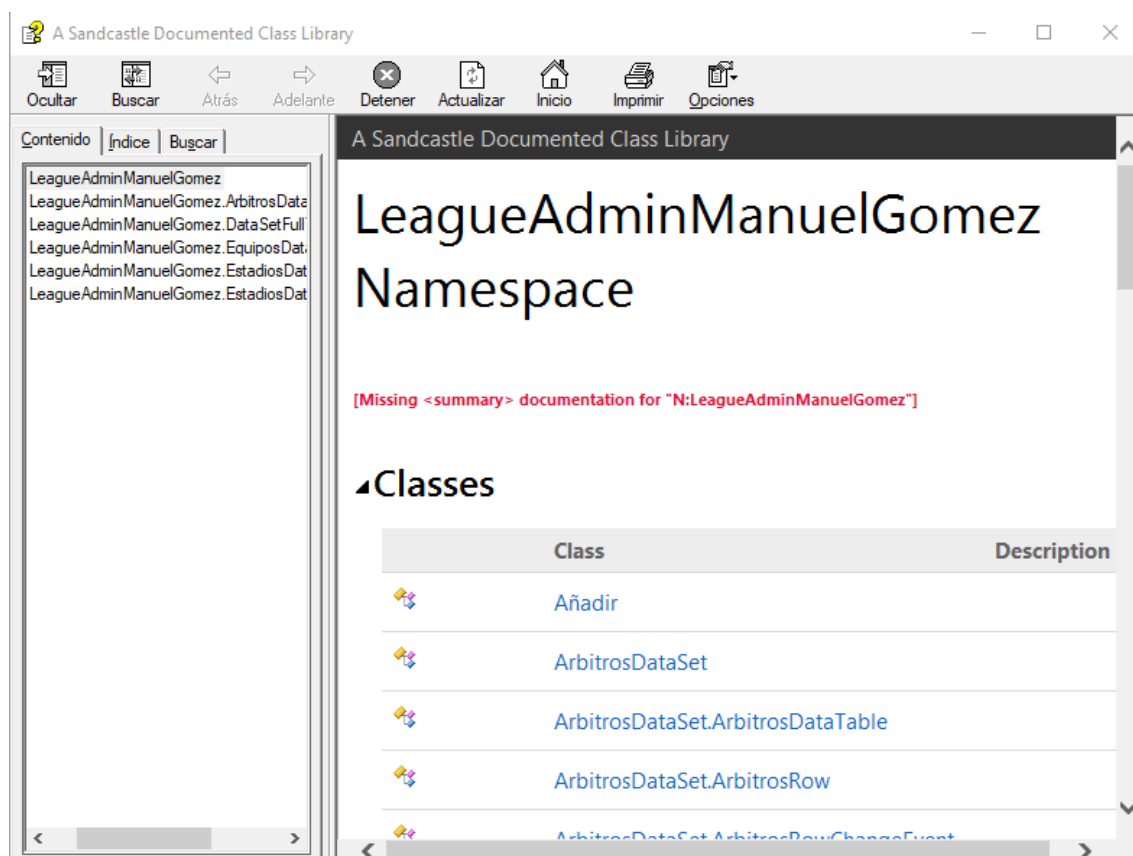
En este manual encontrarás todo lo que debes saber para usar el programa de la manera más útil y eficiente

Continúe con el [índice de contenido](#)

CHM



API



PDF

Pulsando en el siguiente enlace abriras el PDFg

PDF

FUNCIONAMIENTO DE LA APLICACIÓN

A continuación mostraré el funcionamiento de cada uno de los formularios de la aplicación así como las diferentes funciones del formulario principal.

PANTALLA DE INICIO

En esta pantalla existen dos botones:

- SALIR: Finaliza la aplicación
- INICIAR: Inicia el formulario principal

PANTALLA PRINCIPAL

En este formulario, como expliqué en el apartado de diseño, existen 3 paneles.

- ❖ PANEL SUPERIOR: En este panel se encuentran los siguientes botones:
 - Ayuda: Muestra en el panel principal el formulario *AYUDA*.
 - Minimizar: Minimiza la aplicación.
 - Maximizar: Maximiza la aplicación.
 - Cerrar: Muestra un cuadro de diálogo que nos dice si queremos cerrar o no la aplicación.
Si se pulsa sí, la aplicación se cierra.
Si pulsamos no, se cierra el cuadro de diálogo y todo sigue igual.
- ❖ PANEL LATERAL: EN él se encuentran los siguientes botones:
 - COSULTAR: Muestra en el panel principal el formulario *CONSULTAR*.
 - MODIFICAR/BORRAR: Muestra en el panel principal el formulario *MODIFICAR/BORRAR*.
 - AÑADIR: Muestra en el panel principal el formulario *AÑADIR*.
 - INFORMES: Muestra en el panel principal el formulario *INFORMES*.
 - AYUDA: Muestra en el panel principal el formulario *AYUDA*.
 - SALIR: Finaliza la aplicación.
 - MENÚ PRINCIPAL: Vuelve a la pantalla de inicio.
- ❖ PANEL PRINCIPAL: En él se mostrará el formulario cuyo botón hallamos clicado.

A continuación explicaré el funcionamiento de los diferentes formularios que se muestran al pulsar los botones del Menú lateral.

CONSULTAR

Este formulario nos permite consultar registros en la tabla que queramos que contengan la cadena introducida.

Al acceder a este formulario seleccionaremos la tabla en la cuál queremos consultar sus registros. Desués introduciremos una cadena de texto en el TextBox y se mostrarán los registros coincidentes.

MODIFICAR/BORRAR

Este formulario nos permite modificar o borrar cualquier registro de la base de datos.

Al acceder a este formulario seleccionaremos la tabla en la cual queremos consultar sus registros.

A continuación se mostrarán todos los registros de dicha tabla y al hacer doble click sobre uno de los campos de cada uno de estos podremos modificarlo a nuestro gusto.

Para guardar los cambio basta con pulsar la tecla *Intro* o el botón **GUARDAR CAMBIOS**.

Si queremos borrar cualquier registro, basta con eleccionar uno y clicar el botón **BORRAR**

AÑADIR

Este formulario nos permite añadir un registro en cualquier tabla de la base de datos

Basta con seleccionar la tabla en la que queremos añadir el registro, rellenar los campos del registro y pulsar el botón **AÑADIR**

Si algún campo no tiene el formato adecuado o falta alguno, saltará un cuadro de diálogo mostrando el error.

Si nos arrepentimos de lo escrito podemos pulsar el botón **LIMPIAR CAMPOS**

INFORMES

En este formulario podemos ver información más visual de los registros de cada tabla.

Basta con seleccionar la tabla de la cual queremos ver el informe y clicar el botón **GENERAR**

Se generará un informe de la tabla con información de los registros.

Podemos guardar esta información en varios formatos pulsando el botón **Exportar Informe**

DIFICULTADES QUE HAN SURGIDO EN EL DESARROLLO DE LA APLICACIÓN

Puedo decir que, por suerte, no he tenido grandes problemas. Pero si tuviera que documentar los que más me ha costado solucionar, han sido los siguientes:

- FALLO EN CASA DEL SQL SERVER MANAGEMENT STUDIO:
El programa de gestión de bases de datos *SQL Server Management Studio* no me funcionaba correctamente.

Resulta que iniciaba el proceso mostrando la primera ventana pero se quedaba ahí pillado en segundo plano.

Esto me obligó a hacer el grueso del proyecto en clase.

- CLONACIÓN DE LOS ORDENADORES DE CLASE:
Una semana antes de los exámenes, Álvaro, el encargado del mantenimiento de los ordenadores del IES Comercio, decidió hacer una clonación de todos los ordenadores de la clase.
El problema fue que el resultado no fue el esperado y se borraron todos los datos de los ordenadores.
Esto hizo que al volver a instalar los programas para el desarrollo del proyecto de DI algunas funciones de este no funcionaran correctamente.

Por suerte, pude arreglar los errores en poco tiempo y continuar con el proyecto.

POSIBLES AMPLIACIONES DE LA APLICACIÓN

Las dos posibles ampliaciones más realistas e interesantes que en un futuro implementaré en el proyecto son las siguientes:

- AMPLIACIÓN DE LAS FUNCIONES DE LA APLICACIÓN
Las funciones que quiero implementar ya que para el proyecto no me dio tiempo a hacerlas son las siguientes:
 - INCLUIR MÁS EQUIPOS DIFERENCIADOS EN DIVISIONES
Añadir una nueva tabla llamada Divisiones en la que añadir estas y sus datos para luego a cada equipo asignarle en su división con un nuevo campo.
 - CREACIÓN DE UN CALENDARIO DE PARTIDOS
Mediante un algoritmo, crear calendario de partidos para cada división así como simular cada uno de ellos con el número de goles, las tarjetas...
 - CREACIÓN DE UNA TABLA DE CLASIFICACIÓN
Mediante un algoritmo, crear un botón cuya función sea la de generar una tabla de clasificación en base al resultado de los partidos del calendario en cada división.

- ACCESO DESDE CUALQUIER DISPOSITIVO:

Poder acceder a tus datos desde cualquier dispositivo mediante un sistema de usuarios y contraseñas

CONCLUSIÓN

Este proyecto me ha ayudado a reforzar los contenidos dados en clase y a ponerme, más o menos, en el lugar de un desarrollador profesional ya que la aplicación que he creado podría usarse perfectamente en cualquier lugar.

También tengo que decir que el examen se podría haber cambiado por este proyecto porque hemos ido muy apretados de exámenes y proyectos este final de curso.

Por último, este proyecto me ha gustado mucho hacerlo ya que esta es una de mis asignaturas favoritas porque hemos trabajado con interfaces y hemos aprendido C#, un lenguaje muy útil a la par que interesante y sencillo (respecto a JAVA, que es el único que conocía).

REFERENCIAS CONSULTADAS

Tutorial para crear la interfaz gráfica

<https://www.youtube.com/watch?v=5AsJJl7Bhvc&t=1170s>

<https://rjcodeadvance.com/formulario-modernoiconos-font-awesome-winform-c/>

StackOverFlow → <https://es.stackoverflow.com/>

Iconos del proyecto

<https://fontawesome.com/search?m=free>

Paquete Nugget *MaterialSkin*

<https://www.nuget.org/packages/MaterialSkin.2/>

Paquete Nugget *FontAwesome.Sharp*

<https://www.nuget.org/packages/FontAwesome.Sharp/>

Compañeros que me han ayudado:

Sobre todo *Guillermo López Lozano* (ya que su interfaz era bastante similar a la mía) pero también *Juan Cruz Mencía* y *Zakaría Boukabache*