Escuela Profesional de Ciencias de la Computación

UNSA

CURSO:

Ciencia de la computación II

DOCENTE:

Enzo Edir Velásquez Lobatón

ESTUDIANTE:

Quicaño Miranda, Victor Alejandro

Ramirez Zarate, Patrick Rene

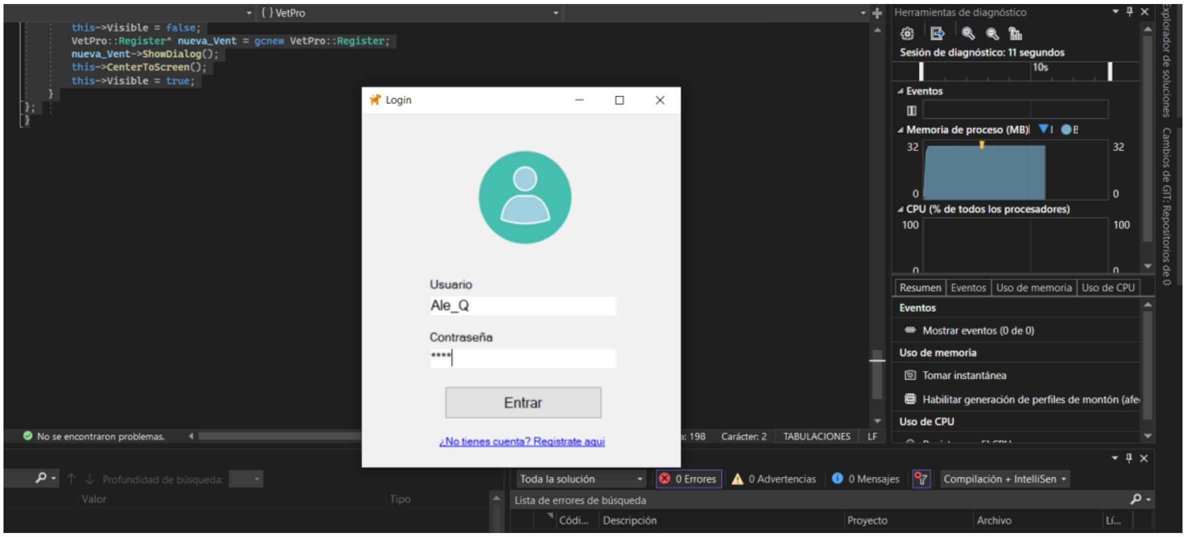
Rodriguez Cutimbo, Gabriel Fernando

AREQUIPA-PERÙ

2022

EXPLICACIÓN DEL PROGRAMA

Login



```
            this->Visible = false;
            VetPro::Register^ nueva_Vent = gcnew VetPro::Register;
            nueva_Vent->ShowDialog();
            this->CenterToScreen();
            this->Visible = true;
        }
};
}
```

Login

Usuario: Ale_Q

Contraseña: ****

No se encontraron problemas.

0 Errores · 0 Advertencias · 0 Mensajes

Sesión de diagnóstico: 11 segundos

Se establece el archivo y las dependencias que tendrá en Login, es decir, las ventanas que posterior mente implementará: El registro y el menú Principal.

```
#pragma once
#include "Register.h"
#include "Main_menu.h"

namespace VetPro {
```

Haciendo uso de using namespace System , llamamos a las librerías que usaremos en esta ventana:

System -> Contiene las clases fundamentales para el fucionamiento de Windows Forms

System::ComponentModel -> Proporcina clases para el buen funcionamiento durante el tiempo de ejecución, durante el tiempo de diseño de componente y controles dentro del programa

System::Collections -> Contiene clases especializadad y de tipeo fuerte, como diccionario de listas enlazadas (Linked List Dictionary) o vectores de bits

System::Windows::Forms -> Provee clases para creación de objetos basados en componentes Windows que aprovechan la interacción con el usuario

System::Data -> Clase especializada en el manejo de datos para enlazar propiedades a orígenes de datos.

System::Drawing -> Clase para dibujar e implementar objetos en 2D

MySql::Data::MySqlClient -> Clase especializada en la conexión a la base de datos proporcionada por MySql.

```
using namespace System;
using namespace System::ComponentModel;
using namespace System::Collections;
using namespace System::Windows::Forms;
using namespace System::Data;
using namespace System::Drawing;
using namespace MySql::Data::MySqlClient;
```

Crecaión de la clase Login la cuál heredará las propiedades y atributos de System Windows Forms específicamente del objeto Forms. Aquí se detalla el constructor. El comenzará con la creación de los objetos que componen el Login. esto con la Función InitializeComponent. Y dando ciertas propiedades a elementos que componen el Form como txt_user y txt_cont, ambos objetos, TextBoxes, encargados de recibir, la información que proporcione el usuario.

Finalmente también se establece el destructor del Form.

```
public ref class Login : public System::Windows::Forms::Form {
public:
  Login(void) {
    InitializeComponent();
    //Corrige un bug de tamaño con las entradas de texto para usuario y contraseña
    this->txt_user->AutoSize = false;
    this->txt_cont->AutoSize = false;

    this->txt_user->Size = System::Drawing::Size(215, 22);
    this->txt_cont->Size = System::Drawing::Size(215, 22);
  }

protected:
  ~Login() {
    if (components) {
      delete components;
    }
  }
```

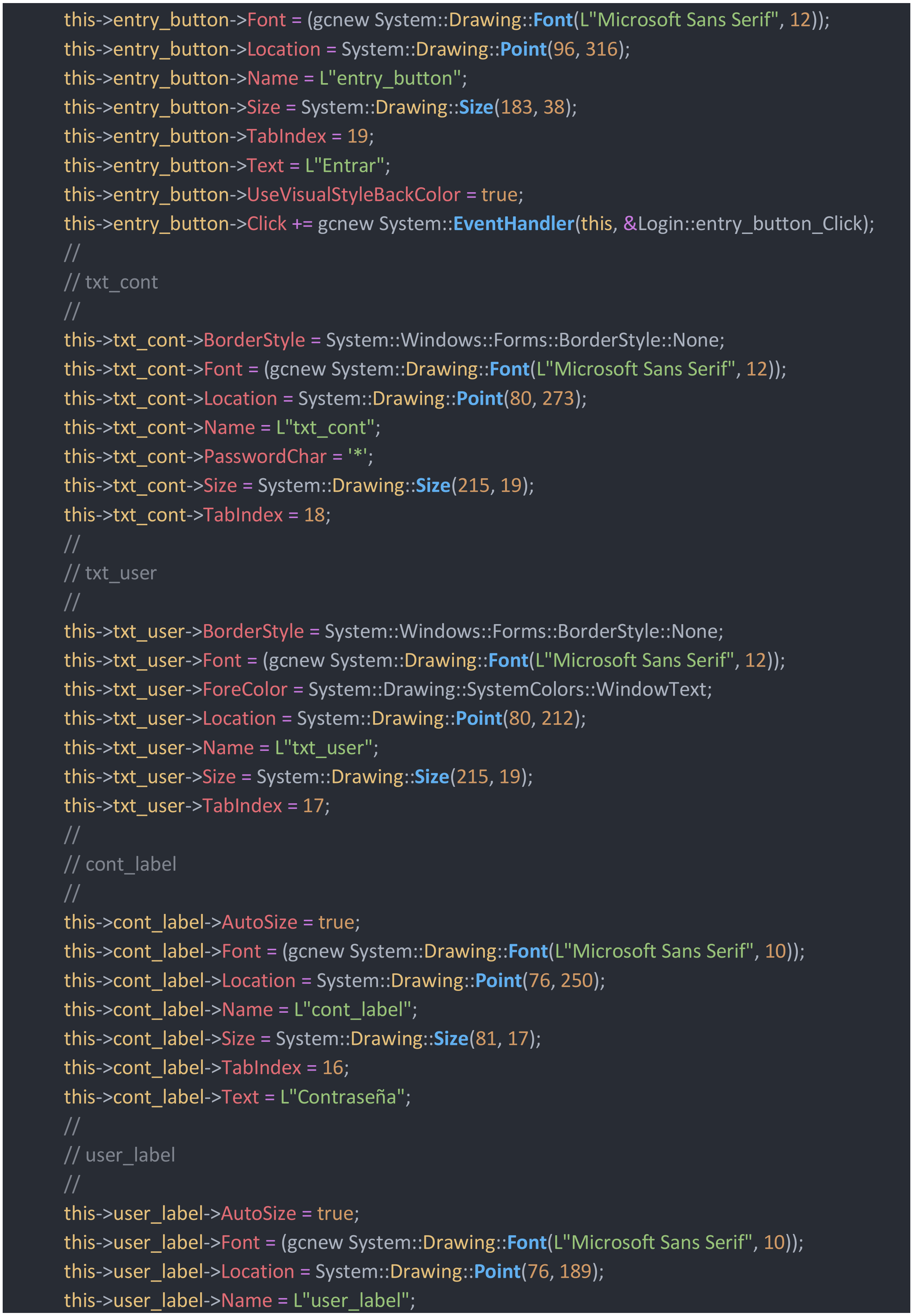
Se establece un String conectionSting, el operador ^ y gcnew son implementados por .NET, todos los objetos creadoscon estos proporcionan un mejor manejo de memoria, colectando automáticamente la basura. Con la ayuda deconectionStrign se establece la conexión con la base de datos, estos haciendo uso del objeto MySqlConnection. Finalmente se procede a crear todos los objetos que se usarán dentro del Login, junto con el establecimiento de cada uno de los objetos del programa.

```

  private: String^ connectionString = "datasource=localhost; username=root; password=1234; 
database=veterinaria";
  private: MySqlConnection^ conn = gcnew MySqlConnection(connectionString);
  private: System::Windows::Forms::LinkLabel^ regist_label;
```

```
    private: System::Windows::Forms::Button^ entry_button;
    private: System::Windows::Forms::TextBox^ txt_cont;
    private: System::Windows::Forms::TextBox^ txt_user;
    private: System::Windows::Forms::Label^ cont_label;
    private: System::Windows::Forms::Label^ user_label;
    private: System::Windows::Forms::PictureBox^ login_img;

    private: System::ComponentModel::Container ^components;

#pragma region Windows Form Designer generated code
        /// <summary>
        /// Método necesario para admitir el Diseñador. No se puede modificar
        /// el contenido de este método con el editor de código.
        /// </summary>
```

InitializaComponent es la función que se utilizó en el inicio y es encargada de crear cada uno de los elementos del programa, nombre tipo de dato y propiedades

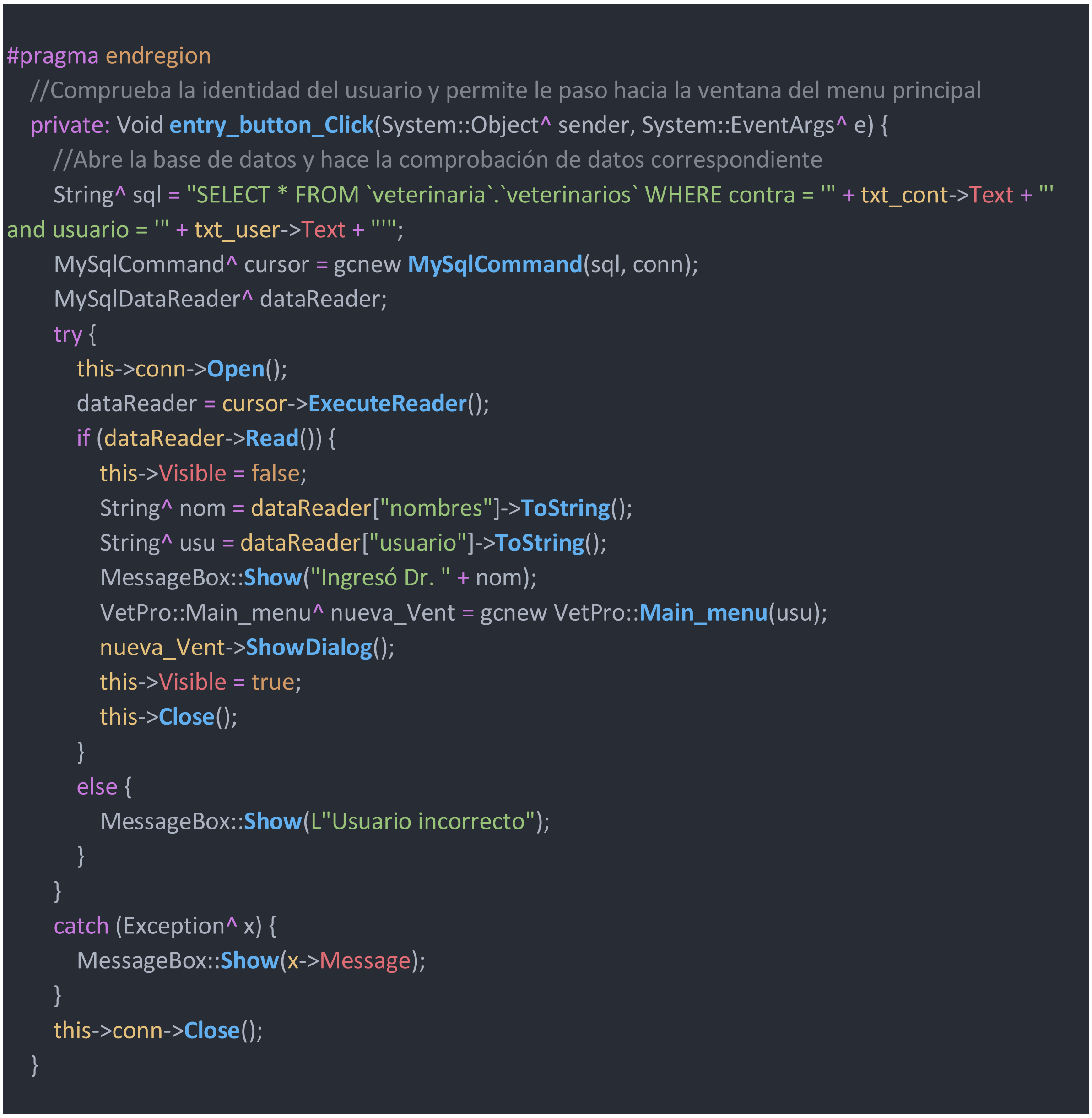
```

        void InitializeComponent(void) {
            System::ComponentModel::ComponentResourceManager^ resources = (gcnew
System::ComponentModel::ComponentResourceManager(Login::typeid));
            this->regist_label = (gcnew System::Windows::Forms::LinkLabel());
            this->entry_button = (gcnew System::Windows::Forms::Button());
            this->txt_cont = (gcnew System::Windows::Forms::TextBox());
            this->txt_user = (gcnew System::Windows::Forms::TextBox());
            this->cont_label = (gcnew System::Windows::Forms::Label());
            this->user_label = (gcnew System::Windows::Forms::Label());
            this->login_img = (gcnew System::Windows::Forms::PictureBox());
            (cli::safe_cast<System::ComponentModel::ISupportInitialize^>(this->login_img))->BeginInit();
            this->SuspendLayout();
            //
            // regist_label
            //
            this->regist_label->AutoSize = true;
            this->regist_label->Font = (gcnew System::Drawing::Font(L"Microsoft Sans Serif", 9));
            this->regist_label->Location = System::Drawing::Point(87, 371);
            this->regist_label->Name = L"regist_label";
            this->regist_label->Size = System::Drawing::Size(199, 15);
            this->regist_label->TabIndex = 20;
            this->regist_label->TabStop = true;
            this->regist_label->Text = L"¿No tienes cuenta\? Registrate aqui";
            this->regist_label->LinkClicked += gcnew
System::Windows::Forms::LinkLabelLinkClickedEventHandler(this, &Login::regist_label_LinkClicked);
            //
            // entry_button
            //
```



```
this->entry_button->Font = (gcnew System::Drawing::Font(L"Microsoft Sans Serif", 12));
this->entry_button->Location = System::Drawing::Point(96, 316);
this->entry_button->Name = L"entry_button";
this->entry_button->Size = System::Drawing::Size(183, 38);
this->entry_button->TabIndex = 19;
this->entry_button->Text = L"Entrar";
this->entry_button->UseVisualStyleBackColor = true;
this->entry_button->Click += gcnew System::EventHandler(this, &Login::entry_button_Click);
//
// txt_cont
//
this->txt_cont->BorderStyle = System::Windows::Forms::BorderStyle::None;
this->txt_cont->Font = (gcnew System::Drawing::Font(L"Microsoft Sans Serif", 12));
this->txt_cont->Location = System::Drawing::Point(80, 273);
this->txt_cont->Name = L"txt_cont";
this->txt_cont->PasswordChar = '*';
this->txt_cont->Size = System::Drawing::Size(215, 19);
this->txt_cont->TabIndex = 18;
//
// txt_user
//
this->txt_user->BorderStyle = System::Windows::Forms::BorderStyle::None;
this->txt_user->Font = (gcnew System::Drawing::Font(L"Microsoft Sans Serif", 12));
this->txt_user->ForeColor = System::Drawing::SystemColors::WindowText;
this->txt_user->Location = System::Drawing::Point(80, 212);
this->txt_user->Name = L"txt_user";
this->txt_user->Size = System::Drawing::Size(215, 19);
this->txt_user->TabIndex = 17;
//
// cont_label
//
this->cont_label->AutoSize = true;
this->cont_label->Font = (gcnew System::Drawing::Font(L"Microsoft Sans Serif", 10));
this->cont_label->Location = System::Drawing::Point(76, 250);
this->cont_label->Name = L"cont_label";
this->cont_label->Size = System::Drawing::Size(81, 17);
this->cont_label->TabIndex = 16;
this->cont_label->Text = L"Contraseña";
//
// user_label
//
this->user_label->AutoSize = true;
this->user_label->Font = (gcnew System::Drawing::Font(L"Microsoft Sans Serif", 10));
this->user_label->Location = System::Drawing::Point(76, 189);
this->user_label->Name = L"user_label";
```

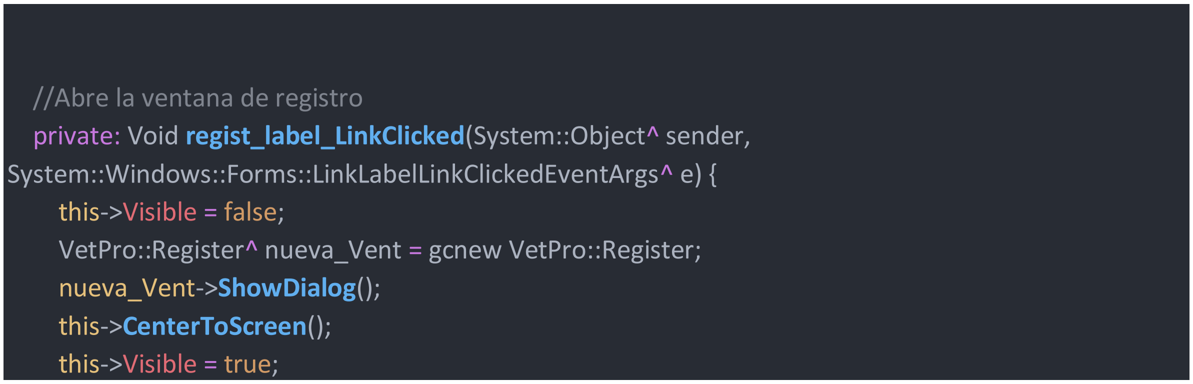
```
        this->user_label->Size = System::Drawing::Size(57, 17);
        this->user_label->TabIndex = 15;
        this->user_label->Text = L"Usuario";
        //
        // login_img
        //
        this->login_img->Image = (cli::safe_cast<System::Drawing::Image^>(resources-
>GetObject(L"login_img.Image")));
        this->login_img->Location = System::Drawing::Point(118, 24);
        this->login_img->Name = L"login_img";
        this->login_img->Size = System::Drawing::Size(143, 148);
        this->login_img->SizeMode = System::Windows::Forms::PictureBoxSizeMode::Zoom;
        this->login_img->TabIndex = 14;
        this->login_img->TabStop = false;
        //
        // Login
        //
        this->AutoScaleDimensions = System::Drawing::SizeF(6, 13);
        this->AutoScaleMode = System::Windows::Forms::AutoScaleMode::Font;
        this->ClientSize = System::Drawing::Size(370, 410);
        this->Controls->Add(this->regist_label);
        this->Controls->Add(this->entry_button);
        this->Controls->Add(this->txt_cont);
        this->Controls->Add(this->txt_user);
        this->Controls->Add(this->cont_label);
        this->Controls->Add(this->user_label);
        this->Controls->Add(this->login_img);
        this->FormBorderStyle = System::Windows::Forms::FormBorderStyle::FixedDialog;
        this->Icon = (cli::safe_cast<System::Drawing::Icon^>(resources->GetObject(L"$this.Icon")));
        this->Name = L"Login";
        this->StartPosition = System::Windows::Forms::FormStartPosition::CenterScreen;
        this->Text = L"Login";
        (cli::safe_cast<System::ComponentModel::ISupportInitialize^>(this->login_img))->EndInit();
        this->ResumeLayout(false);
        this->PerformLayout();

    }
    /// </summary>
```

En InitializaComponent también se detallan funciones propias a cada uno de los objetos. Una de las primeras funciones es Click en entry_button. Aquí se comprueba que existe el usuario (veterinario) en la base de datos. Generando código, **METAPROGRAMACIÓN,** en SQL WORKBENCH con las entradas del usuario. En caso de encontrar el usuario se mostrará un mensaje de bienvenida, inicia el Main_manu y cierra la ventana Login. Caso contrario mostrara si se ingresó mal alguno de los datos o hubo algún error con la base de datos.



```
#pragma endregion
   //Comprueba la identidad del usuario y permite le paso hacia la ventana del menu principal
   private: Void entry_button_Click(System::Object^ sender, System::EventArgs^ e) {
      //Abre la base de datos y hace la comprobación de datos correspondiente
      String^ sql = "SELECT * FROM `veterinaria`.`veterinarios` WHERE contra = '" + txt_cont->Text + "'
and usuario = '" + txt_user->Text + "'";
      MySqlCommand^ cursor = gcnew MySqlCommand(sql, conn);
      MySqlDataReader^ dataReader;
      try {
         this->conn->Open();
         dataReader = cursor->ExecuteReader();
         if (dataReader->Read()) {
            this->Visible = false;
            String^ nom = dataReader["nombres"]->ToString();
            String^ usu = dataReader["usuario"]->ToString();
            MessageBox::Show("Ingresó Dr. " + nom);
            VetPro::Main_menu^ nueva_Vent = gcnew VetPro::Main_menu(usu);
            nueva_Vent->ShowDialog();
            this->Visible = true;
            this->Close();
         }
         else {
            MessageBox::Show(L"Usuario incorrecto");
         }
      }
      catch (Exception^ x) {
         MessageBox::Show(x->Message);
      }
      this->conn->Close();
   }
```

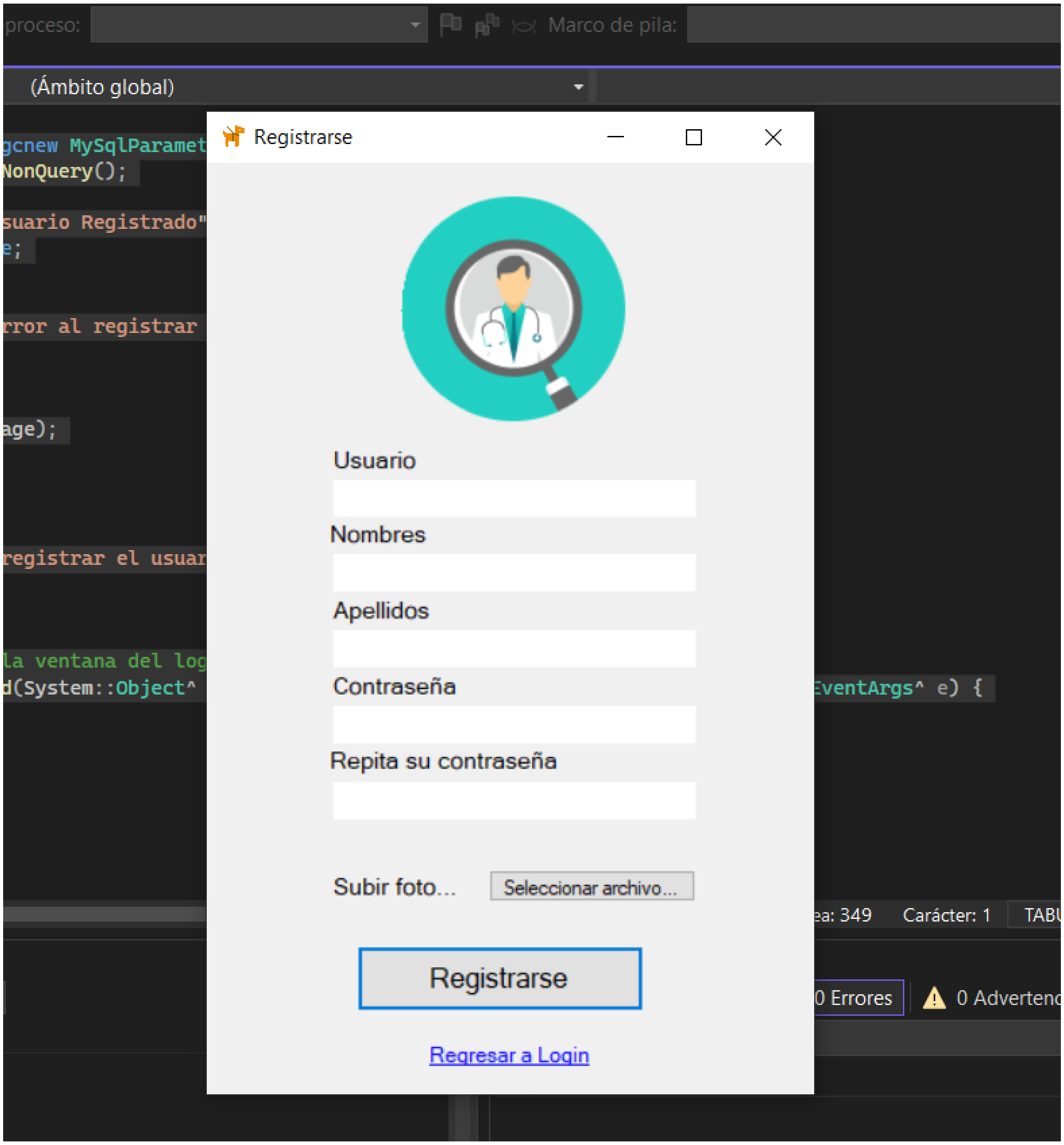
También está la posibilidad de registrarse, al hacer click en register_label, la ventana login se oculta y se genera la ventana de registro , mediante Windows Forms, se hace visible y se coloca en el centro de la pantalla.



```
   //Abre la ventana de registro
   private: Void regist_label_LinkClicked(System::Object^ sender,
System::Windows::Forms::LinkLabelLinkClickedEventArgs^ e) {
      this->Visible = false;
      VetPro::Register^ nueva_Vent = gcnew VetPro::Register;
      nueva_Vent->ShowDialog();
      this->CenterToScreen();
      this->Visible = true;
```

```
  }
};
}
```

Register

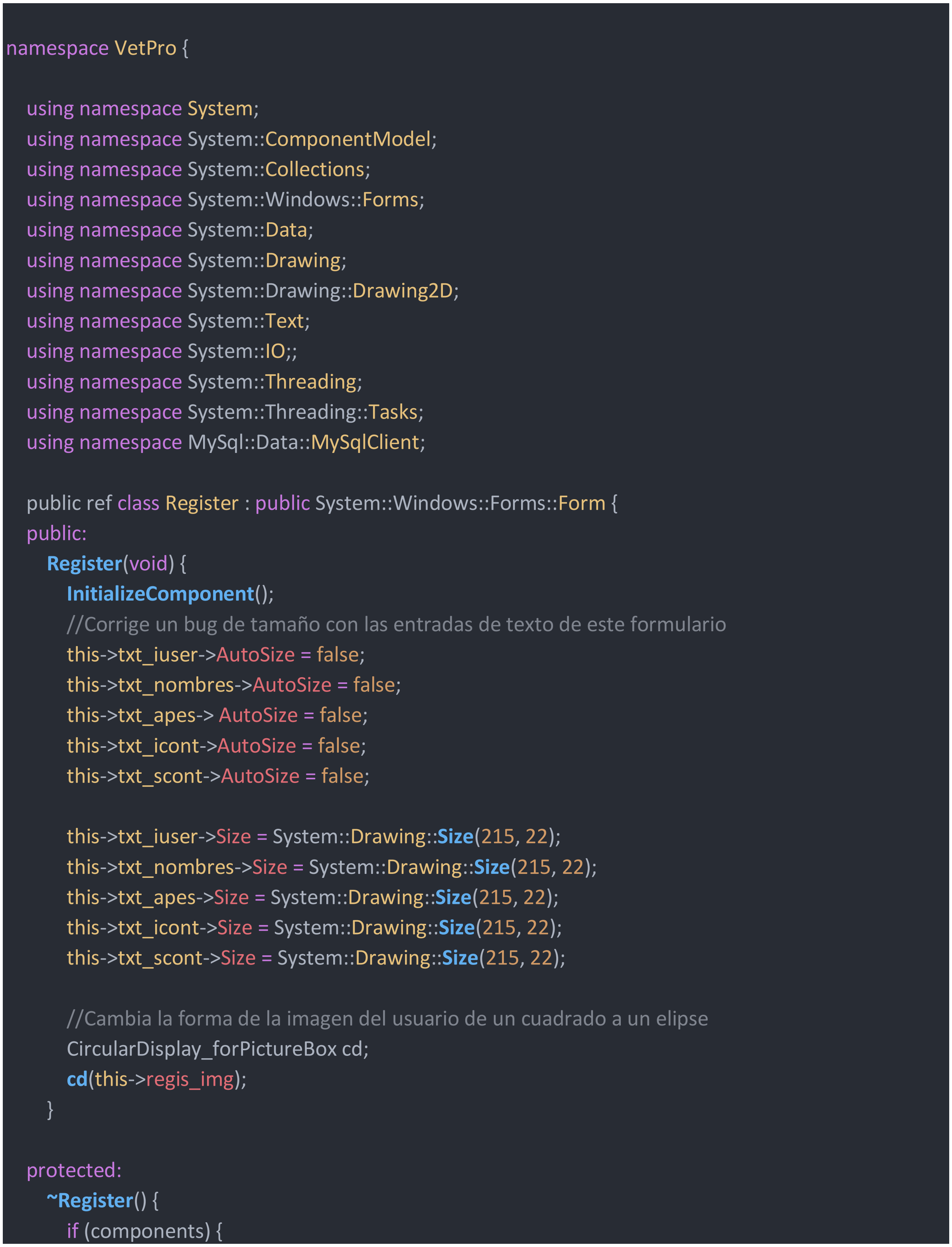


```
gcnew MySqlParamet
NonQuery();

suario Registrado"
e;

rror al registrar

age);

registrar el usuar

la ventana del log
d(System::Object^                                   EventArgs^ e) {
```

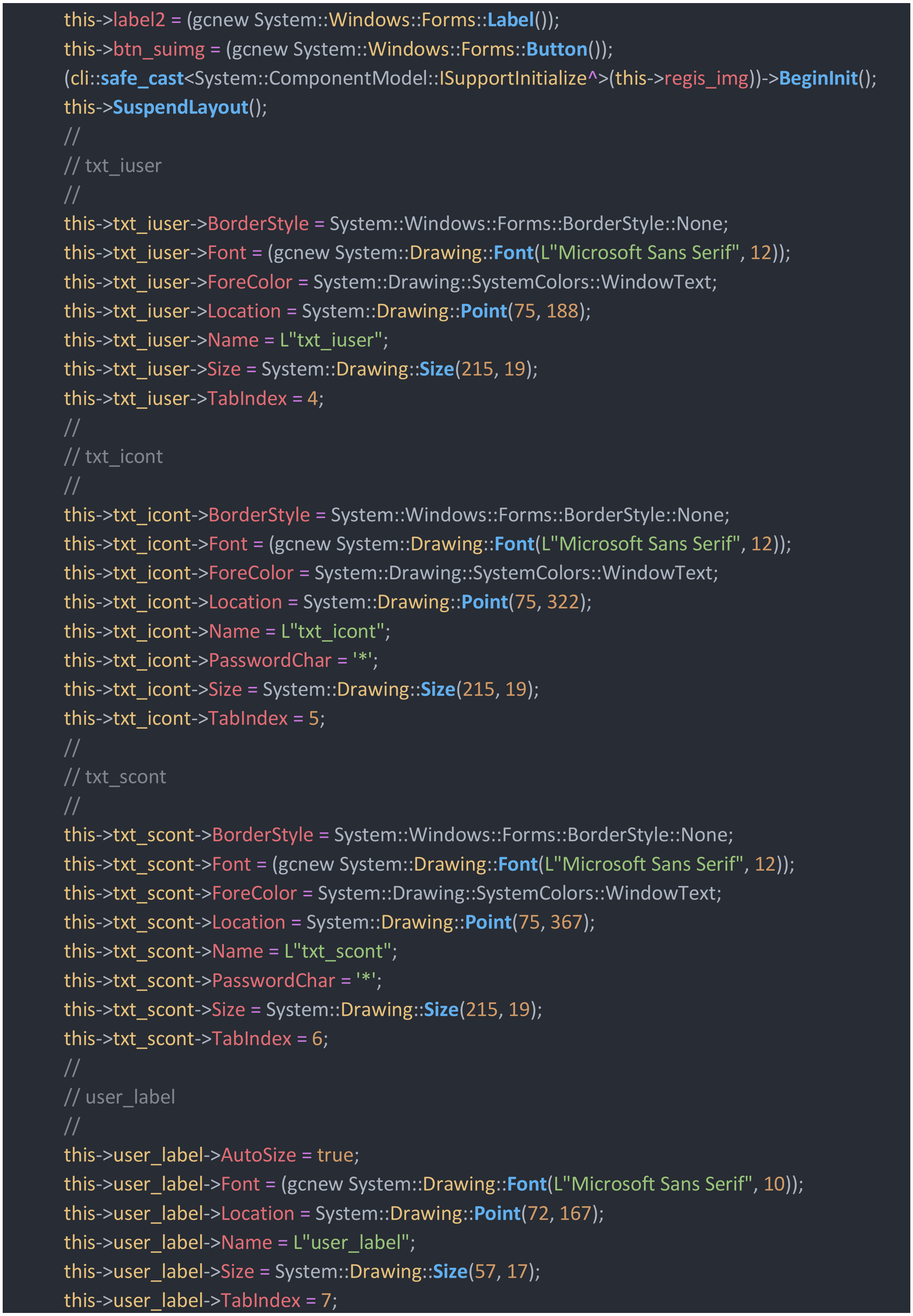
La inicialización de esta ventana es similar al resto de ventanas que forman parte del proyeto. En esta ocasión se hace uso de una clase, CicrularDisplay_forPictureBox.

```
#pragma once
#include "CircularDisplay_forPictureBox.h"
```

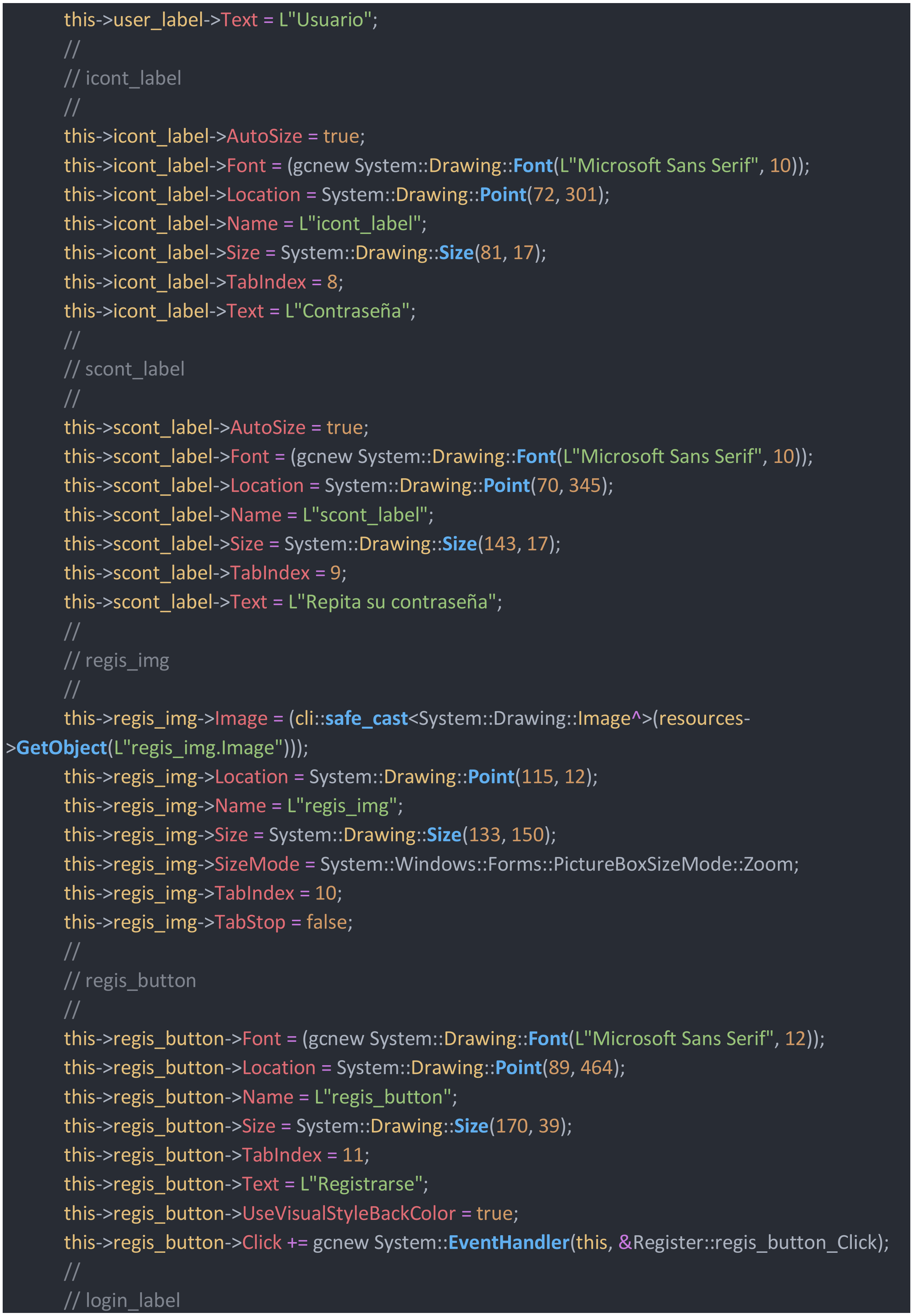
De forma nativa, la clase PictureBox no dispone de formas más que la cuadrada y rectangular. La clase Circular mediante functores, permite cambiar esto.



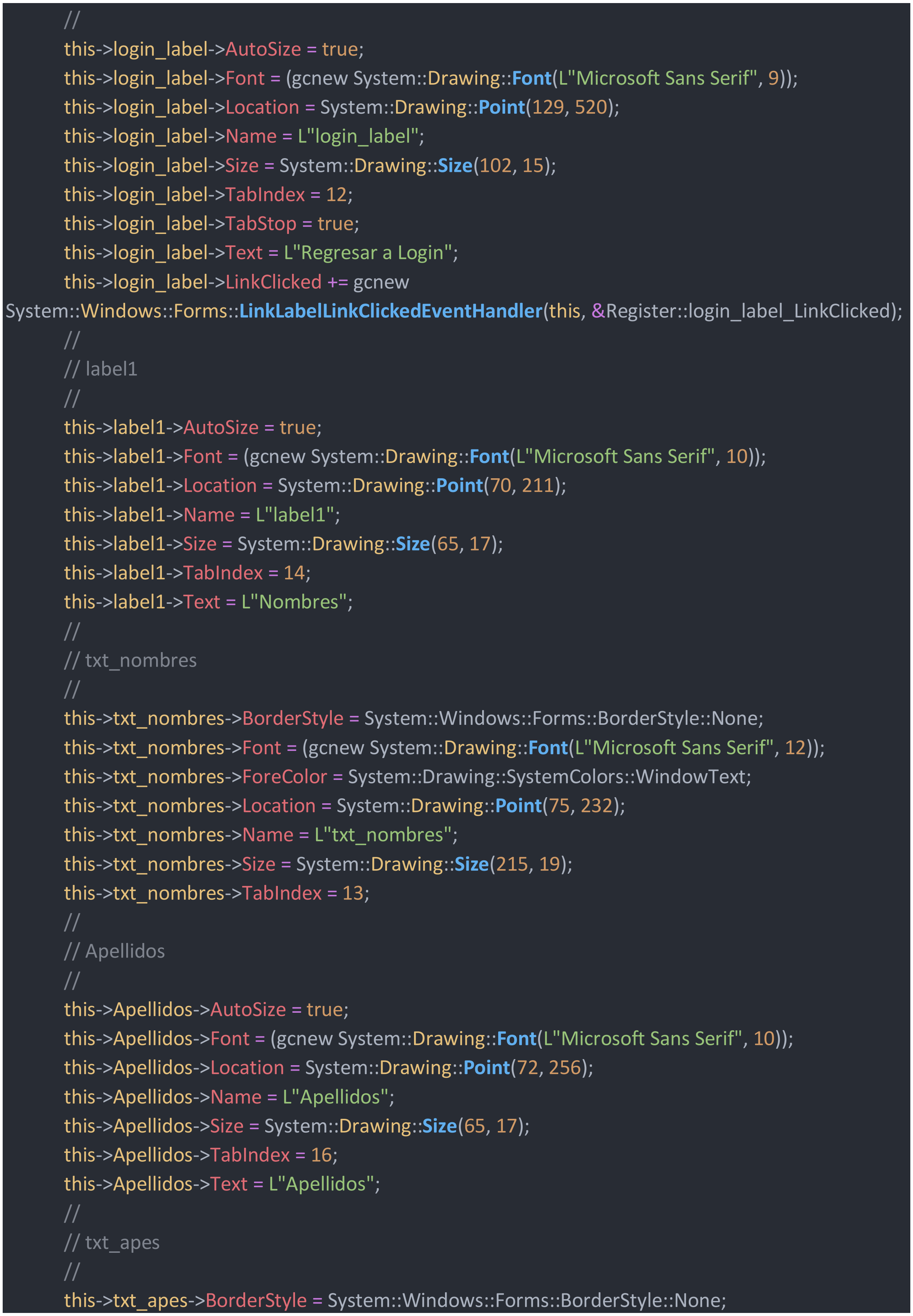
```
namespace VetPro {

  using namespace System;
  using namespace System::ComponentModel;
  using namespace System::Collections;
  using namespace System::Windows::Forms;
  using namespace System::Data;
  using namespace System::Drawing;
  using namespace System::Drawing::Drawing2D;
  using namespace System::Text;
  using namespace System::IO;;
  using namespace System::Threading;
  using namespace System::Threading::Tasks;
  using namespace MySql::Data::MySqlClient;

  public ref class Register : public System::Windows::Forms::Form {
  public:
    Register(void) {
      InitializeComponent();
      //Corrige un bug de tamaño con las entradas de texto de este formulario
      this->txt_iuser->AutoSize = false;
      this->txt_nombres->AutoSize = false;
      this->txt_apes-> AutoSize = false;
      this->txt_icont->AutoSize = false;
      this->txt_scont->AutoSize = false;

      this->txt_iuser->Size = System::Drawing::Size(215, 22);
      this->txt_nombres->Size = System::Drawing::Size(215, 22);
      this->txt_apes->Size = System::Drawing::Size(215, 22);
      this->txt_icont->Size = System::Drawing::Size(215, 22);
      this->txt_scont->Size = System::Drawing::Size(215, 22);

      //Cambia la forma de la imagen del usuario de un cuadrado a un elipse
      CircularDisplay_forPictureBox cd;
      cd(this->regis_img);
    }

  protected:
    ~Register() {
      if (components) {
           delete components;
      }
    }
  private: String^ connectionString = "datasource=localhost; username=root; password=1234;
database=veterinaria";
  private: MySqlConnection^ conn = gcnew MySqlConnection(connectionString);
  private: System::Windows::Forms::TextBox^ txt_iuser;
  private: System::Windows::Forms::TextBox^ txt_icont;
  private: System::Windows::Forms::TextBox^ txt_scont;
  private: System::Windows::Forms::Label^ user_label;
  private: System::Windows::Forms::Label^ icont_label;
  private: System::Windows::Forms::Label^ scont_label;
  private: System::Windows::Forms::PictureBox^ regis_img;
  private: System::Windows::Forms::Button^ regis_button;
  private: System::Windows::Forms::LinkLabel^ login_label;
  private: System::Windows::Forms::Label^ label1;
  private: System::Windows::Forms::TextBox^ txt_nombres;
  private: System::Windows::Forms::Label^ Apellidos;
  private: System::Windows::Forms::TextBox^ txt_apes;
  private: System::Windows::Forms::Label^ label2;
  private: System::Windows::Forms::Button^ btn_suimg;

  private: System::ComponentModel::Container ^components;

#pragma region Windows Form Designer generated code
    /// <summary>
    /// Método necesario para admitir el Diseñador. No se puede modificar
    /// el contenido de este método con el editor de código.
    /// </summary>
    void InitializeComponent(void) {
      System::ComponentModel::ComponentResourceManager^ resources = (gcnew
System::ComponentModel::ComponentResourceManager(Register::typeid));
      this->txt_iuser = (gcnew System::Windows::Forms::TextBox());
      this->txt_icont = (gcnew System::Windows::Forms::TextBox());
      this->txt_scont = (gcnew System::Windows::Forms::TextBox());
      this->user_label = (gcnew System::Windows::Forms::Label());
      this->icont_label = (gcnew System::Windows::Forms::Label());
      this->scont_label = (gcnew System::Windows::Forms::Label());
      this->regis_img = (gcnew System::Windows::Forms::PictureBox());
      this->regis_button = (gcnew System::Windows::Forms::Button());
      this->login_label = (gcnew System::Windows::Forms::LinkLabel());
      this->label1 = (gcnew System::Windows::Forms::Label());
      this->txt_nombres = (gcnew System::Windows::Forms::TextBox());
      this->Apellidos = (gcnew System::Windows::Forms::Label());
      this->txt_apes = (gcnew System::Windows::Forms::TextBox());
```



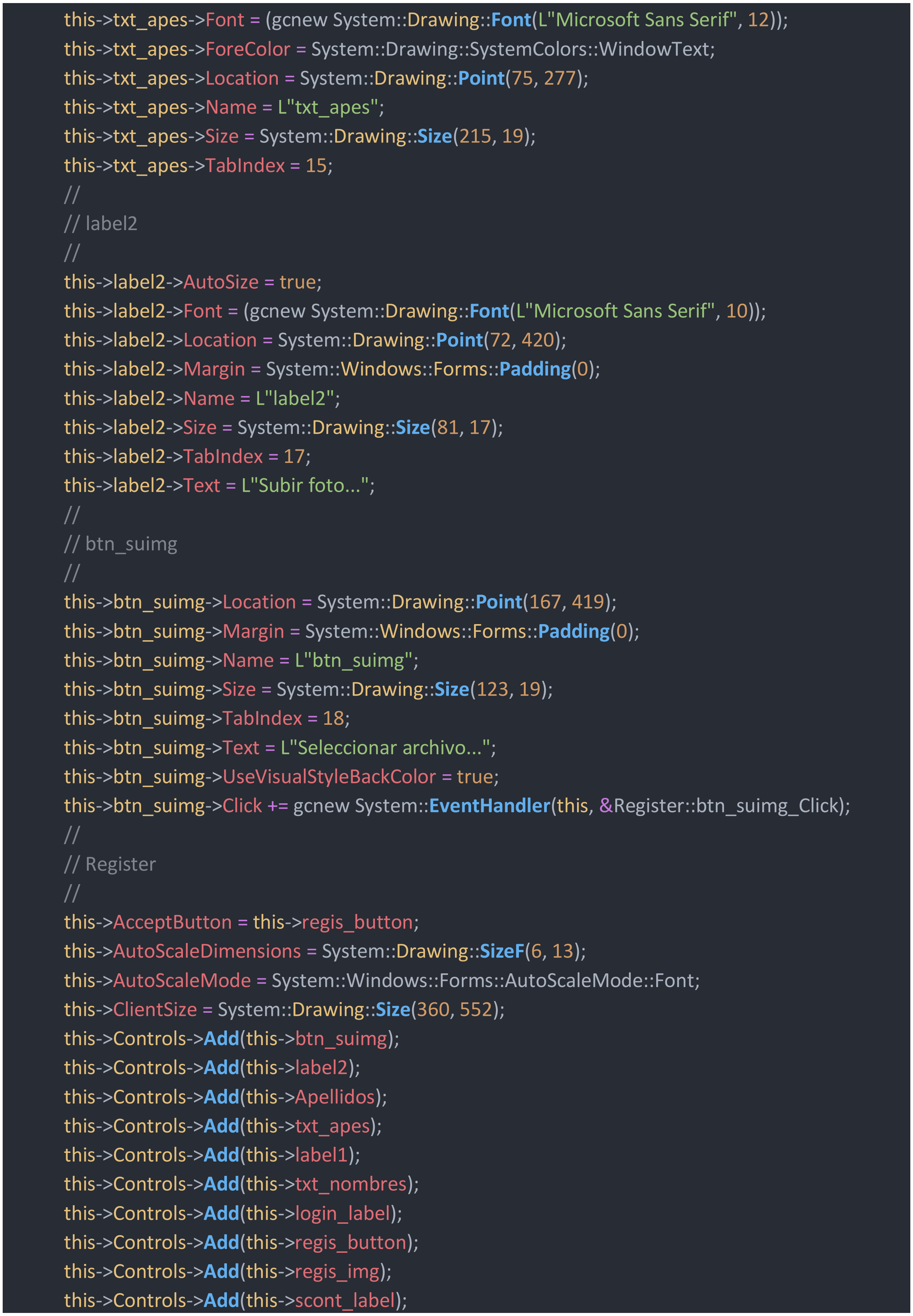
```
this->label2 = (gcnew System::Windows::Forms::Label());
this->btn_suimg = (gcnew System::Windows::Forms::Button());
(cli::safe_cast<System::ComponentModel::ISupportInitialize^>(this->regis_img))->BeginInit();
this->SuspendLayout();
//
// txt_iuser
//
this->txt_iuser->BorderStyle = System::Windows::Forms::BorderStyle::None;
this->txt_iuser->Font = (gcnew System::Drawing::Font(L"Microsoft Sans Serif", 12));
this->txt_iuser->ForeColor = System::Drawing::SystemColors::WindowText;
this->txt_iuser->Location = System::Drawing::Point(75, 188);
this->txt_iuser->Name = L"txt_iuser";
this->txt_iuser->Size = System::Drawing::Size(215, 19);
this->txt_iuser->TabIndex = 4;
//
// txt_icont
//
this->txt_icont->BorderStyle = System::Windows::Forms::BorderStyle::None;
this->txt_icont->Font = (gcnew System::Drawing::Font(L"Microsoft Sans Serif", 12));
this->txt_icont->ForeColor = System::Drawing::SystemColors::WindowText;
this->txt_icont->Location = System::Drawing::Point(75, 322);
this->txt_icont->Name = L"txt_icont";
this->txt_icont->PasswordChar = '*';
this->txt_icont->Size = System::Drawing::Size(215, 19);
this->txt_icont->TabIndex = 5;
//
// txt_scont
//
this->txt_scont->BorderStyle = System::Windows::Forms::BorderStyle::None;
this->txt_scont->Font = (gcnew System::Drawing::Font(L"Microsoft Sans Serif", 12));
this->txt_scont->ForeColor = System::Drawing::SystemColors::WindowText;
this->txt_scont->Location = System::Drawing::Point(75, 367);
this->txt_scont->Name = L"txt_scont";
this->txt_scont->PasswordChar = '*';
this->txt_scont->Size = System::Drawing::Size(215, 19);
this->txt_scont->TabIndex = 6;
//
// user_label
//
this->user_label->AutoSize = true;
this->user_label->Font = (gcnew System::Drawing::Font(L"Microsoft Sans Serif", 10));
this->user_label->Location = System::Drawing::Point(72, 167);
this->user_label->Name = L"user_label";
this->user_label->Size = System::Drawing::Size(57, 17);
this->user_label->TabIndex = 7;
```



```
        this->user_label->Text = L"Usuario";
        //
        // icont_label
        //
        this->icont_label->AutoSize = true;
        this->icont_label->Font = (gcnew System::Drawing::Font(L"Microsoft Sans Serif", 10));
        this->icont_label->Location = System::Drawing::Point(72, 301);
        this->icont_label->Name = L"icont_label";
        this->icont_label->Size = System::Drawing::Size(81, 17);
        this->icont_label->TabIndex = 8;
        this->icont_label->Text = L"Contraseña";
        //
        // scont_label
        //
        this->scont_label->AutoSize = true;
        this->scont_label->Font = (gcnew System::Drawing::Font(L"Microsoft Sans Serif", 10));
        this->scont_label->Location = System::Drawing::Point(70, 345);
        this->scont_label->Name = L"scont_label";
        this->scont_label->Size = System::Drawing::Size(143, 17);
        this->scont_label->TabIndex = 9;
        this->scont_label->Text = L"Repita su contraseña";
        //
        // regis_img
        //
        this->regis_img->Image = (cli::safe_cast<System::Drawing::Image^>(resources-
>GetObject(L"regis_img.Image")));
        this->regis_img->Location = System::Drawing::Point(115, 12);
        this->regis_img->Name = L"regis_img";
        this->regis_img->Size = System::Drawing::Size(133, 150);
        this->regis_img->SizeMode = System::Windows::Forms::PictureBoxSizeMode::Zoom;
        this->regis_img->TabIndex = 10;
        this->regis_img->TabStop = false;
        //
        // regis_button
        //
        this->regis_button->Font = (gcnew System::Drawing::Font(L"Microsoft Sans Serif", 12));
        this->regis_button->Location = System::Drawing::Point(89, 464);
        this->regis_button->Name = L"regis_button";
        this->regis_button->Size = System::Drawing::Size(170, 39);
        this->regis_button->TabIndex = 11;
        this->regis_button->Text = L"Registrarse";
        this->regis_button->UseVisualStyleBackColor = true;
        this->regis_button->Click += gcnew System::EventHandler(this, &Register::regis_button_Click);
        //
        // login_label
```



```
        //
        this->login_label->AutoSize = true;
        this->login_label->Font = (gcnew System::Drawing::Font(L"Microsoft Sans Serif", 9));
        this->login_label->Location = System::Drawing::Point(129, 520);
        this->login_label->Name = L"login_label";
        this->login_label->Size = System::Drawing::Size(102, 15);
        this->login_label->TabIndex = 12;
        this->login_label->TabStop = true;
        this->login_label->Text = L"Regresar a Login";
        this->login_label->LinkClicked += gcnew System::Windows::Forms::LinkLabelLinkClickedEventHandler(this, &Register::login_label_LinkClicked);
        //
        // label1
        //
        this->label1->AutoSize = true;
        this->label1->Font = (gcnew System::Drawing::Font(L"Microsoft Sans Serif", 10));
        this->label1->Location = System::Drawing::Point(70, 211);
        this->label1->Name = L"label1";
        this->label1->Size = System::Drawing::Size(65, 17);
        this->label1->TabIndex = 14;
        this->label1->Text = L"Nombres";
        //
        // txt_nombres
        //
        this->txt_nombres->BorderStyle = System::Windows::Forms::BorderStyle::None;
        this->txt_nombres->Font = (gcnew System::Drawing::Font(L"Microsoft Sans Serif", 12));
        this->txt_nombres->ForeColor = System::Drawing::SystemColors::WindowText;
        this->txt_nombres->Location = System::Drawing::Point(75, 232);
        this->txt_nombres->Name = L"txt_nombres";
        this->txt_nombres->Size = System::Drawing::Size(215, 19);
        this->txt_nombres->TabIndex = 13;
        //
        // Apellidos
        //
        this->Apellidos->AutoSize = true;
        this->Apellidos->Font = (gcnew System::Drawing::Font(L"Microsoft Sans Serif", 10));
        this->Apellidos->Location = System::Drawing::Point(72, 256);
        this->Apellidos->Name = L"Apellidos";
        this->Apellidos->Size = System::Drawing::Size(65, 17);
        this->Apellidos->TabIndex = 16;
        this->Apellidos->Text = L"Apellidos";
        //
        // txt_apes
        //
        this->txt_apes->BorderStyle = System::Windows::Forms::BorderStyle::None;
```



```
this->txt_apes->Font = (gcnew System::Drawing::Font(L"Microsoft Sans Serif", 12));
this->txt_apes->ForeColor = System::Drawing::SystemColors::WindowText;
this->txt_apes->Location = System::Drawing::Point(75, 277);
this->txt_apes->Name = L"txt_apes";
this->txt_apes->Size = System::Drawing::Size(215, 19);
this->txt_apes->TabIndex = 15;
//
// label2
//
this->label2->AutoSize = true;
this->label2->Font = (gcnew System::Drawing::Font(L"Microsoft Sans Serif", 10));
this->label2->Location = System::Drawing::Point(72, 420);
this->label2->Margin = System::Windows::Forms::Padding(0);
this->label2->Name = L"label2";
this->label2->Size = System::Drawing::Size(81, 17);
this->label2->TabIndex = 17;
this->label2->Text = L"Subir foto...";
//
// btn_suimg
//
this->btn_suimg->Location = System::Drawing::Point(167, 419);
this->btn_suimg->Margin = System::Windows::Forms::Padding(0);
this->btn_suimg->Name = L"btn_suimg";
this->btn_suimg->Size = System::Drawing::Size(123, 19);
this->btn_suimg->TabIndex = 18;
this->btn_suimg->Text = L"Seleccionar archivo...";
this->btn_suimg->UseVisualStyleBackColor = true;
this->btn_suimg->Click += gcnew System::EventHandler(this, &Register::btn_suimg_Click);
//
// Register
//
this->AcceptButton = this->regis_button;
this->AutoScaleDimensions = System::Drawing::SizeF(6, 13);
this->AutoScaleMode = System::Windows::Forms::AutoScaleMode::Font;
this->ClientSize = System::Drawing::Size(360, 552);
this->Controls->Add(this->btn_suimg);
this->Controls->Add(this->label2);
this->Controls->Add(this->Apellidos);
this->Controls->Add(this->txt_apes);
this->Controls->Add(this->label1);
this->Controls->Add(this->txt_nombres);
this->Controls->Add(this->login_label);
this->Controls->Add(this->regis_button);
this->Controls->Add(this->regis_img);
this->Controls->Add(this->scont_label);
```

```
        this->Controls->Add(this->icont_label);
        this->Controls->Add(this->user_label);
        this->Controls->Add(this->txt_scont);
        this->Controls->Add(this->txt_icont);
        this->Controls->Add(this->txt_iuser);
        this->FormBorderStyle = System::Windows::Forms::FormBorderStyle::FixedSingle;
        this->Icon = (cli::safe_cast<System::Drawing::Icon^>(resources->GetObject(L"$this.Icon")));
        this->Name = L"Register";
        this->StartPosition = System::Windows::Forms::FormStartPosition::CenterScreen;
        this->Text = L"Registrarse";
        (cli::safe_cast<System::ComponentModel::ISupportInitialize^>(this->regis_img))->EndInit();
        this->ResumeLayout(false);
        this->PerformLayout();

    }

#pragma endregion
```

Esta función de tipo booleano comprueba que tanto el nombre de usuario y contraseña no esten vacíos

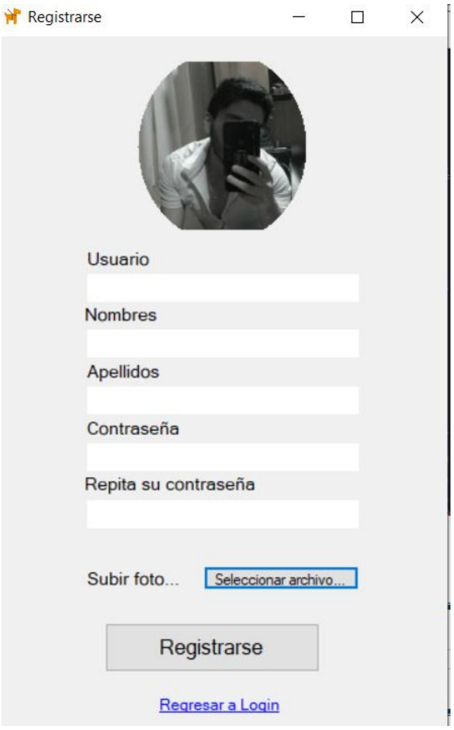
```
   private: System::Boolean checkTextBox_Empty() {
      if (System::String::IsNullOrEmpty(txt_iuser->Text) || System::String::IsNullOrEmpty(txt_icont-
>Text)) {
         return false;
      }
      else {
         return true;
      }
   }
```

Bloque encargado de almacenar la ubicacion de un archivo y actualizar la imagen de regis_img. Conociendo la ubicación, el archivo será convertido a un array de bytes haciendo uso de una transformación de Filestream con un Binary Reader

```

   private: String^ filelocation;
   private: array<Byte>^ imagen;
   private: FileStream^ fst;
   private: BinaryReader^ br;
```

Acción del botón btn_suimg al hacerle click, esta función abrirá una ventana para seleccionar archivos. Se establece un filtro para que solo reconozca archivos tipo jpeg, png y jpeg. Al ser seleccionado un archivo y no ocurrir errores se almacena su ubicación en la variable que implementamos antes. Se convierte la ubicación a un filestream, este se pasa por un BinaryReader, se almacena su contenido en una array de bytes, este array en un MemoryStream que es leído por el pictureBox el que setea ese memory stream como el valor de imagen que mostrará.(Resultado adjunto al final)

```
private: Void btn_suimg_Click(System::Object^ sender, System::EventArgs^ e) {
   //Almacenar la ubicacion de la imagen
   OpenFileDialog^ openFile1 = gcnew OpenFileDialog;
   openFile1->Filter = "Imagenes|*.png;*.jpeg;*.jpg";
   if (System::Windows::Forms::DialogResult::OK == openFile1->ShowDialog()) {
      filelocation = openFile1->FileName->ToString();
   }

   //Actualiza la imagen de masc_Picture
   imagen = gcnew array <Byte>(3000);
   FileStream^ fst = gcnew FileStream(filelocation, FileMode::Open, FileAccess::Read);
   BinaryReader^ br = gcnew BinaryReader(fst);
   imagen = br->ReadBytes((int)fst->Length);
   MemoryStream^ ms = gcnew MemoryStream(imagen);
   this->regis_img->Image = Drawing::Image::FromStream(ms);
}
```

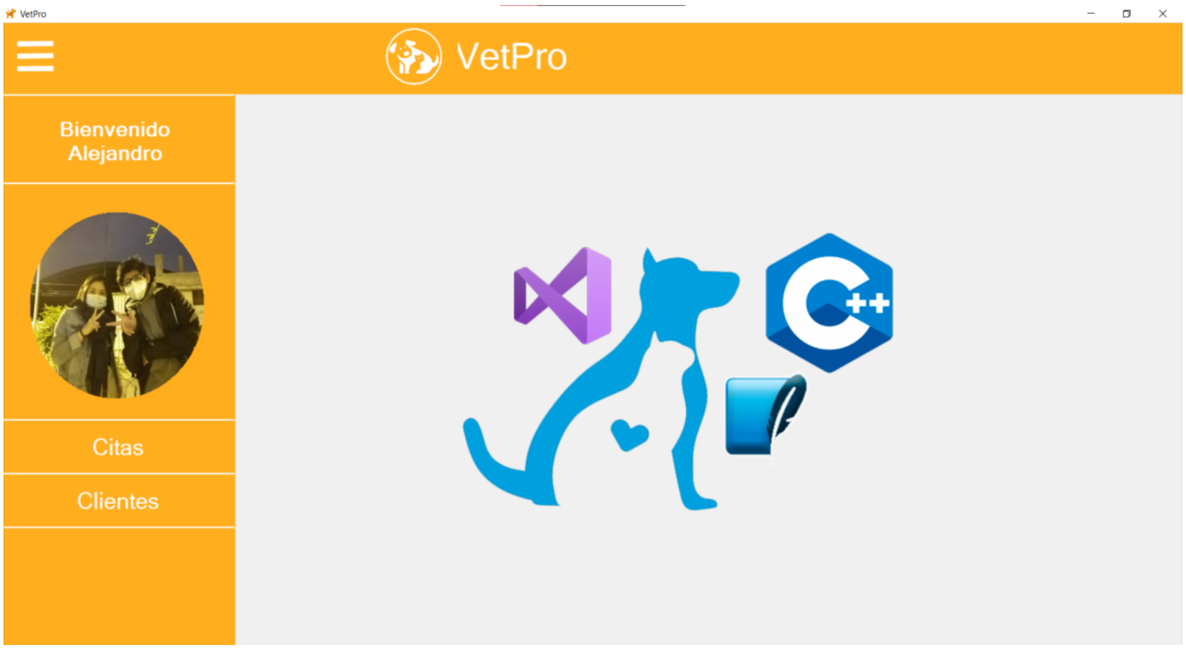


Esta es la funcionalidad principal, el click en el botón de registro, este sube la información de los textbox y los sube a la base de datos. Para evitar confusiones con el array de bits se usa un pseudónimo @IMG, y gracias a la librería MySq::Data::MySqlClient con la función Parameters.

```
private: Void regis_button_Click(System::Object^ sender, System::EventArgs^ e) {
   //Este if hace un chequeo para evitar que la contraseñas sean diferentes y que existan campos
vacios
```

```
      if (txt_icont->Text == txt_scont->Text && checkTextBox_Empty()) {
         //Apertura, ingreso de datos del usuario y cierre de la base de datos
         String^ sql = "INSERT INTO `veterinaria`.`veterinarios` (`usuario`,`contra`,
`nombres`,`apellidos`,`Foto`) VALUE ('" + txt_iuser->Text + "', '" + txt_icont->Text + "', '" + txt_nombres-
>Text + "', '" + txt_apes->Text + "', @IMG)";
         MySqlCommand^ cursor = gcnew MySqlCommand(sql, conn);
         try {
            this->conn->Open();
            cursor->Parameters->Add(gcnew MySqlParameter("@IMG", imagen));
            bool i = cursor->ExecuteNonQuery();
            if (i) {
               MessageBox::Show(L"Usuario Registrado");
               this->Visible = false;
            }
            else {
               MessageBox::Show(L"Error al registrar el usuario");
            }
         }
         catch (Exception^ x) {
            MessageBox::Show(x->Message);
         }
         this->conn->Close();
      }
      else {
         MessageBox::Show(L"Error al registrar el usuario");
      }
   }

   //Esta funcion permite a regresar a la ventana del login
   private: Void login_label_LinkClicked(System::Object^ sender,
System::Windows::Forms::LinkLabelLinkClickedEventArgs^ e) {
      this->Visible = false;
   }
};
}
```

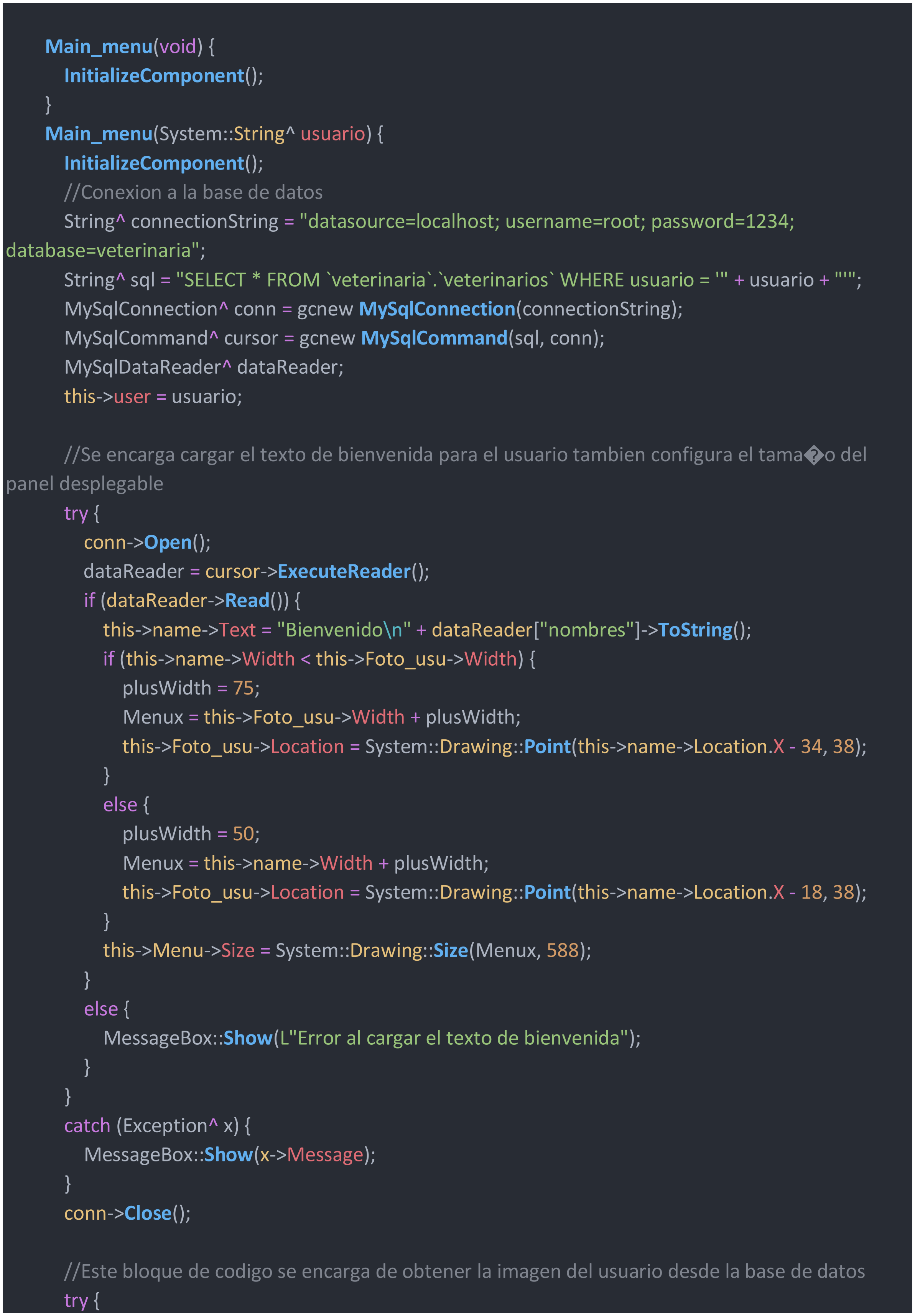
Main_menu



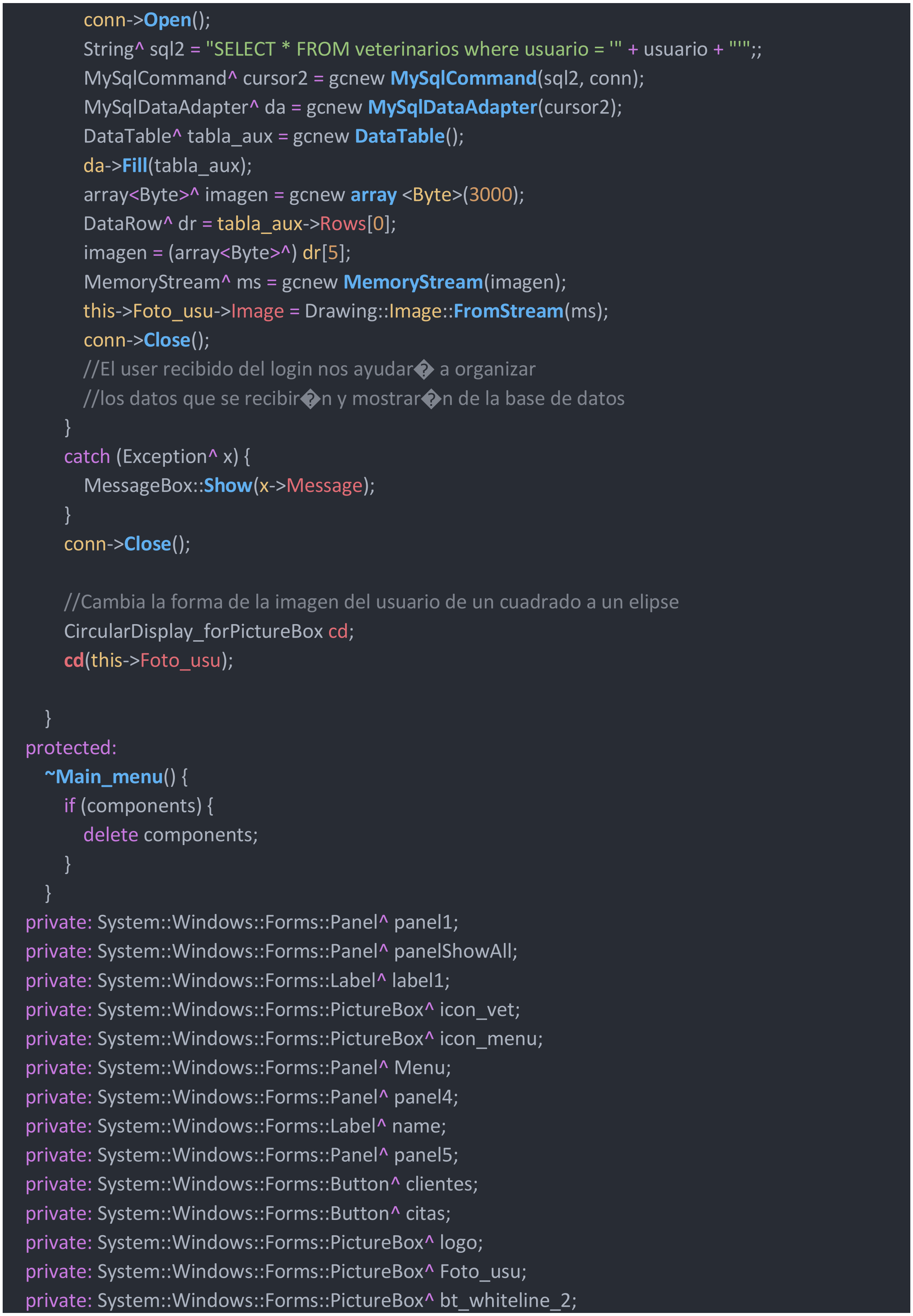
En esta venta se implemntan las clases de 2 ventanas que llamaremos más adelante. Citas y Clientes, forms que mostrarán los clientes registrados, las mascotas y las citas.

```
#pragma once
#include "Cita.h"
#include "Cliente.h"
#include "CircularDisplay_forPictureBox.h"

namespace VetPro {

    using namespace System;
    using namespace System::ComponentModel;
    using namespace System::Collections;
    using namespace System::Windows::Forms;
    using namespace System::Data;
    using namespace System::Drawing;
    using namespace System::Text;
    using namespace System::IO;
    using namespace System::Threading;
    using namespace System::Threading::Tasks;
    using namespace System::Threading;
    using namespace MySql::Data::MySqlClient;

    public ref class Main_menu : public System::Windows::Forms::Form {
    public:
```

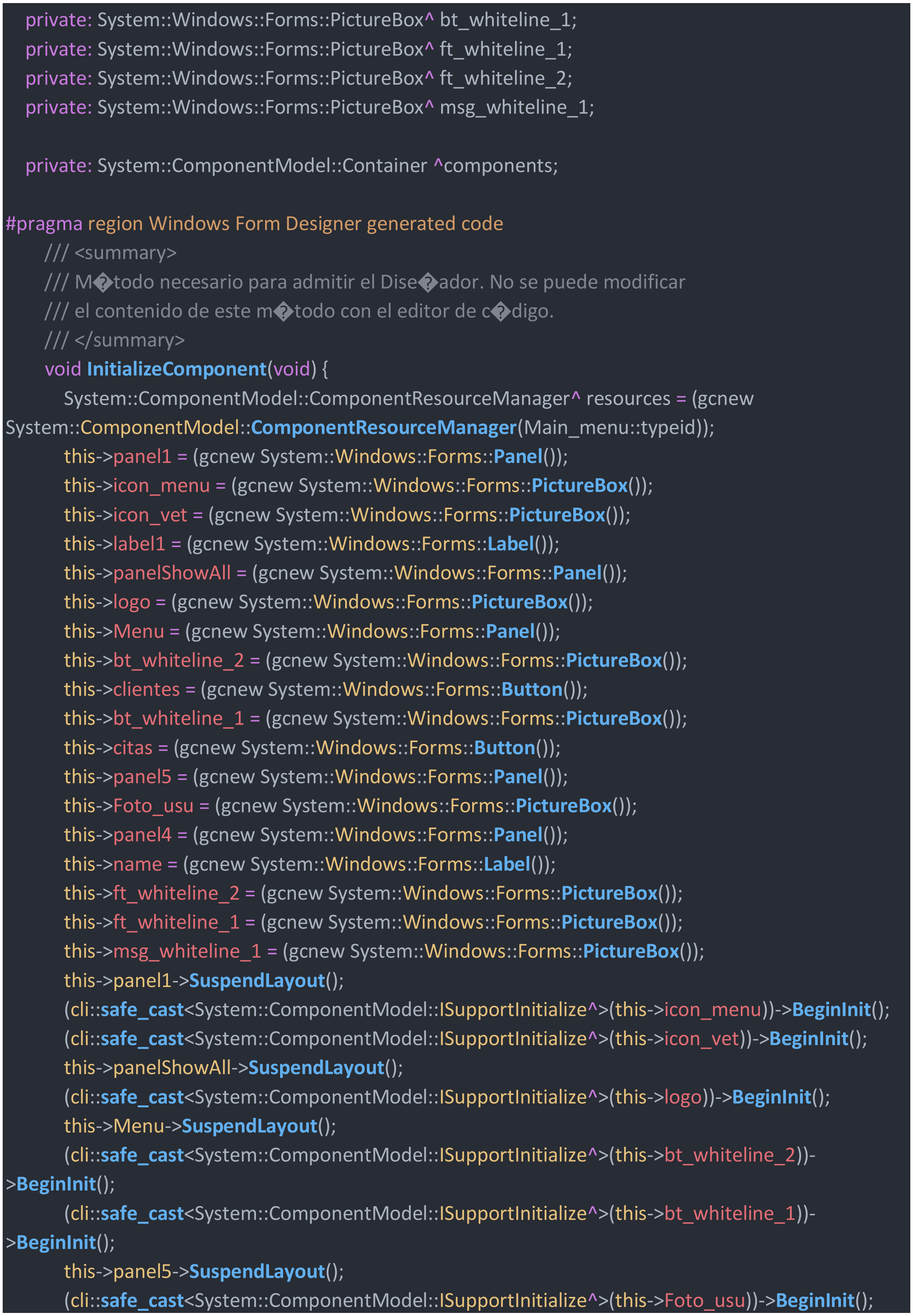
Al inicializar el formulario se pide la variable de tipo String que almacena el usuario. Esta servirá como llave para extraer el nombre y foto de la base de datos.



```
    Main_menu(void) {
      InitializeComponent();
    }
    Main_menu(System::String^ usuario) {
      InitializeComponent();
      //Conexion a la base de datos
      String^ connectionString = "datasource=localhost; username=root; password=1234;
database=veterinaria";
      String^ sql = "SELECT * FROM `veterinaria`.`veterinarios` WHERE usuario = '" + usuario + "'";
      MySqlConnection^ conn = gcnew MySqlConnection(connectionString);
      MySqlCommand^ cursor = gcnew MySqlCommand(sql, conn);
      MySqlDataReader^ dataReader;
      this->user = usuario;

      //Se encarga cargar el texto de bienvenida para el usuario tambien configura el tama�o del
panel desplegable
      try {
        conn->Open();
        dataReader = cursor->ExecuteReader();
        if (dataReader->Read()) {
          this->name->Text = "Bienvenido\n" + dataReader["nombres"]->ToString();
          if (this->name->Width < this->Foto_usu->Width) {
            plusWidth = 75;
            Menux = this->Foto_usu->Width + plusWidth;
            this->Foto_usu->Location = System::Drawing::Point(this->name->Location.X - 34, 38);
          }
          else {
            plusWidth = 50;
            Menux = this->name->Width + plusWidth;
            this->Foto_usu->Location = System::Drawing::Point(this->name->Location.X - 18, 38);
          }
          this->Menu->Size = System::Drawing::Size(Menux, 588);
        }
        else {
          MessageBox::Show(L"Error al cargar el texto de bienvenida");
        }
      }
      catch (Exception^ x) {
        MessageBox::Show(x->Message);
      }
      conn->Close();

      //Este bloque de codigo se encarga de obtener la imagen del usuario desde la base de datos
      try {
```



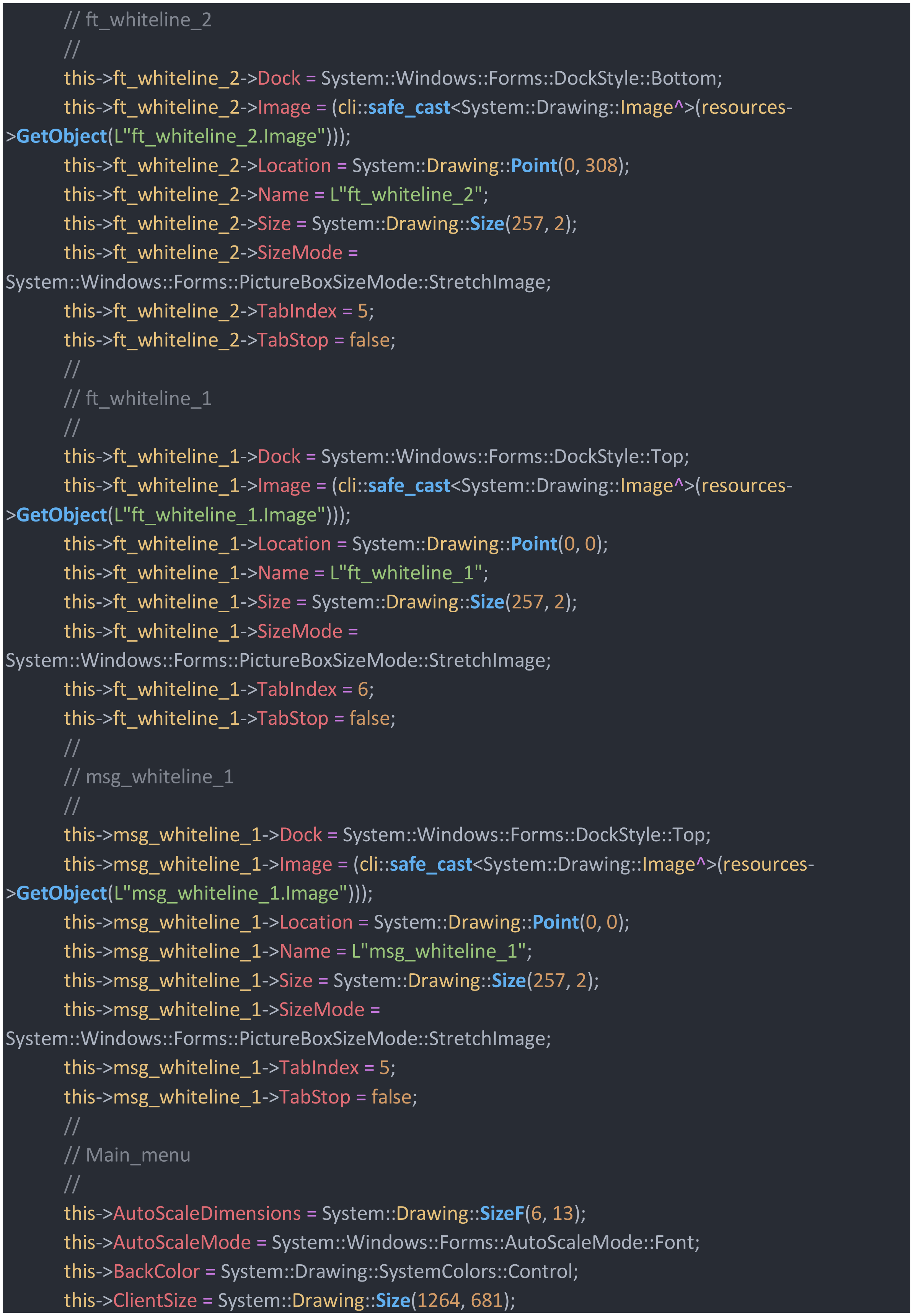
```
            conn->Open();
            String^ sql2 = "SELECT * FROM veterinarios where usuario = '" + usuario + "'";;
            MySqlCommand^ cursor2 = gcnew MySqlCommand(sql2, conn);
            MySqlDataAdapter^ da = gcnew MySqlDataAdapter(cursor2);
            DataTable^ tabla_aux = gcnew DataTable();
            da->Fill(tabla_aux);
            array<Byte>^ imagen = gcnew array <Byte>(3000);
            DataRow^ dr = tabla_aux->Rows[0];
            imagen = (array<Byte>^) dr[5];
            MemoryStream^ ms = gcnew MemoryStream(imagen);
            this->Foto_usu->Image = Drawing::Image::FromStream(ms);
            conn->Close();
            //El user recibido del login nos ayudar� a organizar
            //los datos que se recibir�n y mostrar�n de la base de datos
        }
        catch (Exception^ x) {
            MessageBox::Show(x->Message);
        }
        conn->Close();

        //Cambia la forma de la imagen del usuario de un cuadrado a un elipse
        CircularDisplay_forPictureBox cd;
        cd(this->Foto_usu);

    }
protected:
    ~Main_menu() {
        if (components) {
            delete components;
        }
    }
private: System::Windows::Forms::Panel^ panel1;
private: System::Windows::Forms::Panel^ panelShowAll;
private: System::Windows::Forms::Label^ label1;
private: System::Windows::Forms::PictureBox^ icon_vet;
private: System::Windows::Forms::PictureBox^ icon_menu;
private: System::Windows::Forms::Panel^ Menu;
private: System::Windows::Forms::Panel^ panel4;
private: System::Windows::Forms::Label^ name;
private: System::Windows::Forms::Panel^ panel5;
private: System::Windows::Forms::Button^ clientes;
private: System::Windows::Forms::Button^ citas;
private: System::Windows::Forms::PictureBox^ logo;
private: System::Windows::Forms::PictureBox^ Foto_usu;
private: System::Windows::Forms::PictureBox^ bt_whiteline_2;
```



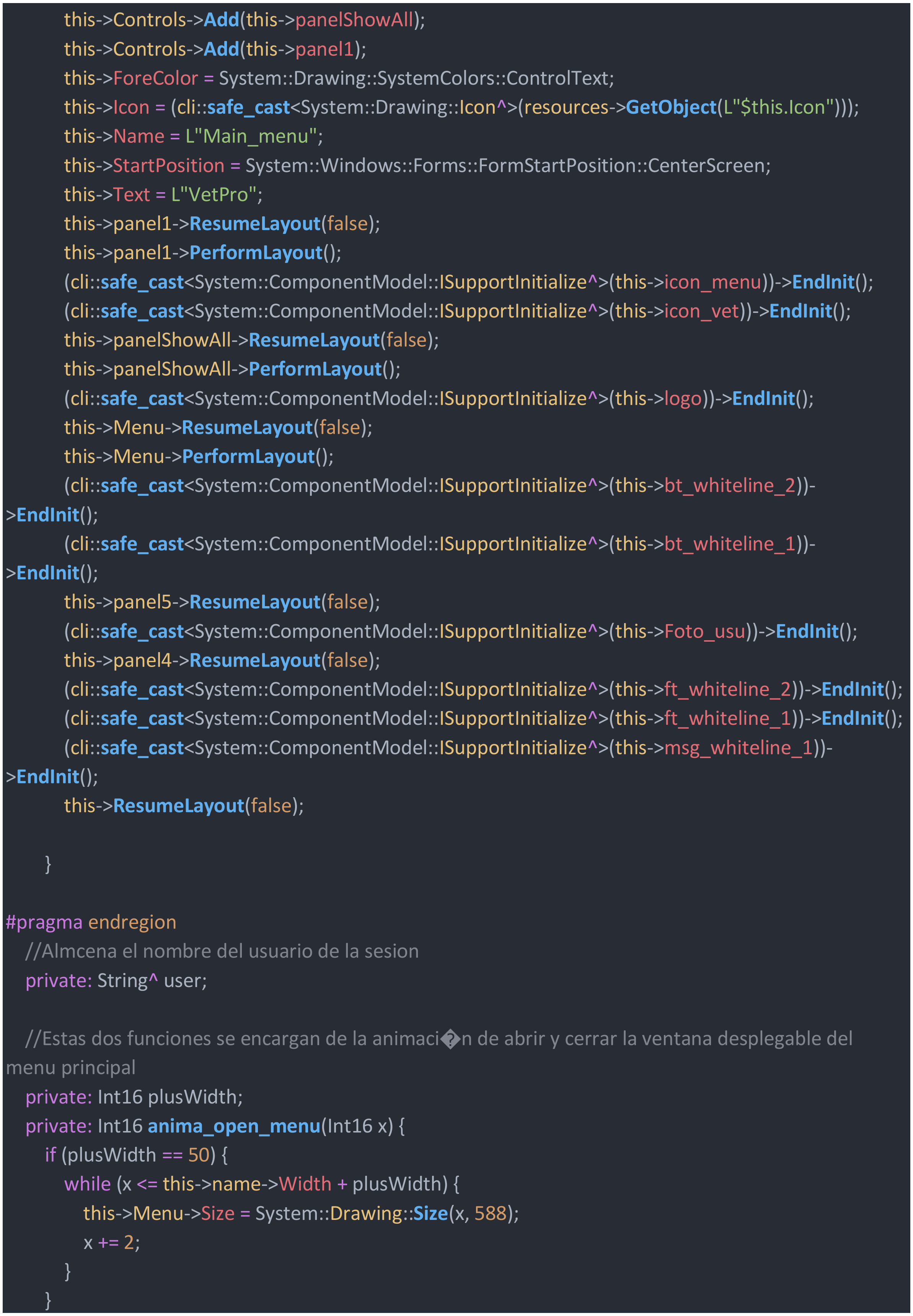
```
	private: System::Windows::Forms::PictureBox^ bt_whiteline_1;
	private: System::Windows::Forms::PictureBox^ ft_whiteline_1;
	private: System::Windows::Forms::PictureBox^ ft_whiteline_2;
	private: System::Windows::Forms::PictureBox^ msg_whiteline_1;

	private: System::ComponentModel::Container ^components;

#pragma region Windows Form Designer generated code
		/// <summary>
		/// M�todo necesario para admitir el Dise�ador. No se puede modificar
		/// el contenido de este m�todo con el editor de c�digo.
		/// </summary>
		void InitializeComponent(void) {
			System::ComponentModel::ComponentResourceManager^ resources = (gcnew
System::ComponentModel::ComponentResourceManager(Main_menu::typeid));
			this->panel1 = (gcnew System::Windows::Forms::Panel());
			this->icon_menu = (gcnew System::Windows::Forms::PictureBox());
			this->icon_vet = (gcnew System::Windows::Forms::PictureBox());
			this->label1 = (gcnew System::Windows::Forms::Label());
			this->panelShowAll = (gcnew System::Windows::Forms::Panel());
			this->logo = (gcnew System::Windows::Forms::PictureBox());
			this->Menu = (gcnew System::Windows::Forms::Panel());
			this->bt_whiteline_2 = (gcnew System::Windows::Forms::PictureBox());
			this->clientes = (gcnew System::Windows::Forms::Button());
			this->bt_whiteline_1 = (gcnew System::Windows::Forms::PictureBox());
			this->citas = (gcnew System::Windows::Forms::Button());
			this->panel5 = (gcnew System::Windows::Forms::Panel());
			this->Foto_usu = (gcnew System::Windows::Forms::PictureBox());
			this->panel4 = (gcnew System::Windows::Forms::Panel());
			this->name = (gcnew System::Windows::Forms::Label());
			this->ft_whiteline_2 = (gcnew System::Windows::Forms::PictureBox());
			this->ft_whiteline_1 = (gcnew System::Windows::Forms::PictureBox());
			this->msg_whiteline_1 = (gcnew System::Windows::Forms::PictureBox());
			this->panel1->SuspendLayout();
			(cli::safe_cast<System::ComponentModel::ISupportInitialize^>(this->icon_menu))->BeginInit();
			(cli::safe_cast<System::ComponentModel::ISupportInitialize^>(this->icon_vet))->BeginInit();
			this->panelShowAll->SuspendLayout();
			(cli::safe_cast<System::ComponentModel::ISupportInitialize^>(this->logo))->BeginInit();
			this->Menu->SuspendLayout();
			(cli::safe_cast<System::ComponentModel::ISupportInitialize^>(this->bt_whiteline_2))-
>BeginInit();
			(cli::safe_cast<System::ComponentModel::ISupportInitialize^>(this->bt_whiteline_1))-
>BeginInit();
			this->panel5->SuspendLayout();
			(cli::safe_cast<System::ComponentModel::ISupportInitialize^>(this->Foto_usu))->BeginInit();
```

```
        this->panel4->SuspendLayout();
        (cli::safe_cast<System::ComponentModel::ISupportInitialize^>(this->ft_whiteline_2))-
>BeginInit();
        (cli::safe_cast<System::ComponentModel::ISupportInitialize^>(this->ft_whiteline_1))-
>BeginInit();
        (cli::safe_cast<System::ComponentModel::ISupportInitialize^>(this->msg_whiteline_1))-
>BeginInit();
        this->SuspendLayout();
        //
        // panel1
        //
        this->panel1->BackColor =
System::Drawing::Color::FromArgb(static_cast<System::Int32>(static_cast<System::Byte>(255)),
static_cast<System::Int32>(static_cast<System::Byte>(174)),
           static_cast<System::Int32>(static_cast<System::Byte>(30)));
        this->panel1->Controls->Add(this->icon_menu);
        this->panel1->Controls->Add(this->icon_vet);
        this->panel1->Controls->Add(this->label1);
        this->panel1->Dock = System::Windows::Forms::DockStyle::Top;
        this->panel1->Location = System::Drawing::Point(0, 0);
        this->panel1->Name = L"panel1";
        this->panel1->Size = System::Drawing::Size(1264, 93);
        this->panel1->TabIndex = 0;
        //
        // icon_menu
        //
        this->icon_menu->Image = (cli::safe_cast<System::Drawing::Image^>(resources-
>GetObject(L"icon_menu.Image")));
        this->icon_menu->Location = System::Drawing::Point(12, 16);
        this->icon_menu->Name = L"icon_menu";
        this->icon_menu->Size = System::Drawing::Size(58, 55);
        this->icon_menu->SizeMode = System::Windows::Forms::PictureBoxSizeMode::StretchImage;
        this->icon_menu->TabIndex = 0;
        this->icon_menu->TabStop = false;
        this->icon_menu->Click += gcnew System::EventHandler(this, &Main_menu::icon_menu_Click);
        //
        // icon_vet
        //
        this->icon_vet->Image = (cli::safe_cast<System::Drawing::Image^>(resources-
>GetObject(L"icon_vet.Image")));
        this->icon_vet->Location = System::Drawing::Point(478, 3);
        this->icon_vet->Name = L"icon_vet";
        this->icon_vet->Size = System::Drawing::Size(114, 81);
        this->icon_vet->SizeMode = System::Windows::Forms::PictureBoxSizeMode::Zoom;
        this->icon_vet->TabIndex = 2;
        this->icon_vet->TabStop = false;
        //
        // label1
        //
        this->label1->AutoSize = true;
        this->label1->Font = (gcnew System::Drawing::Font(L"Microsoft Sans Serif", 36,
System::Drawing::FontStyle::Regular, System::Drawing::GraphicsUnit::Point,
            static_cast<System::Byte>(0)));
        this->label1->ForeColor = System::Drawing::Color::White;
        this->label1->Location = System::Drawing::Point(578, 16);
        this->label1->Name = L"label1";
        this->label1->Size = System::Drawing::Size(171, 55);
        this->label1->TabIndex = 1;
        this->label1->Text = L"VetPro";
        //
        // panelShowAll
        //
        this->panelShowAll->Controls->Add(this->logo);
        this->panelShowAll->Controls->Add(this->Menu);
        this->panelShowAll->Dock = System::Windows::Forms::DockStyle::Fill;
        this->panelShowAll->Location = System::Drawing::Point(0, 93);
        this->panelShowAll->Name = L"panelShowAll";
        this->panelShowAll->Size = System::Drawing::Size(1264, 588);
        this->panelShowAll->TabIndex = 1;
        //
        // logo
        //
        this->logo->Anchor = System::Windows::Forms::AnchorStyles::None;
        this->logo->Image = (cli::safe_cast<System::Drawing::Image^>(resources-
>GetObject(L"logo.Image")));
        this->logo->Location = System::Drawing::Point(446, 103);
        this->logo->Margin = System::Windows::Forms::Padding(2);
        this->logo->Name = L"logo";
        this->logo->Size = System::Drawing::Size(609, 390);
        this->logo->SizeMode = System::Windows::Forms::PictureBoxSizeMode::AutoSize;
        this->logo->TabIndex = 2;
        this->logo->TabStop = false;
        //
        // Menu
        //
        this->Menu->BackColor =
System::Drawing::Color::FromArgb(static_cast<System::Int32>(static_cast<System::Byte>(255)),
static_cast<System::Int32>(static_cast<System::Byte>(174)),
            static_cast<System::Int32>(static_cast<System::Byte>(30)));
        this->Menu->Controls->Add(this->bt_whiteline_2);
        this->Menu->Controls->Add(this->clientes);
        this->Menu->Controls->Add(this->bt_whiteline_1);
        this->Menu->Controls->Add(this->citas);
        this->Menu->Controls->Add(this->panel5);
        this->Menu->Controls->Add(this->panel4);
        this->Menu->Cursor = System::Windows::Forms::Cursors::Arrow;
        this->Menu->Dock = System::Windows::Forms::DockStyle::Left;
        this->Menu->Location = System::Drawing::Point(0, 0);
        this->Menu->Margin = System::Windows::Forms::Padding(2);
        this->Menu->Name = L"Menu";
        this->Menu->Size = System::Drawing::Size(257, 588);
        this->Menu->TabIndex = 1;
        //
        // bt_whiteline_2
        //
        this->bt_whiteline_2->Dock = System::Windows::Forms::DockStyle::Top;
        this->bt_whiteline_2->Image = (cli::safe_cast<System::Drawing::Image^>(resources-
>GetObject(L"bt_whiteline_2.Image")));
        this->bt_whiteline_2->Location = System::Drawing::Point(0, 563);
        this->bt_whiteline_2->Name = L"bt_whiteline_2";
        this->bt_whiteline_2->Size = System::Drawing::Size(257, 2);
        this->bt_whiteline_2->SizeMode =
System::Windows::Forms::PictureBoxSizeMode::StretchImage;
        this->bt_whiteline_2->TabIndex = 5;
        this->bt_whiteline_2->TabStop = false;
        //
        // clientes
        //
        this->clientes->Dock = System::Windows::Forms::DockStyle::Top;
        this->clientes->FlatAppearance->BorderColor =
System::Drawing::Color::FromArgb(static_cast<System::Int32>(static_cast<System::Byte>(255)),
            static_cast<System::Int32>(static_cast<System::Byte>(174)),
static_cast<System::Int32>(static_cast<System::Byte>(30)));
        this->clientes->FlatStyle = System::Windows::Forms::FlatStyle::Flat;
        this->clientes->Font = (gcnew System::Drawing::Font(L"Microsoft Sans Serif", 22.2F,
System::Drawing::FontStyle::Regular, System::Drawing::GraphicsUnit::Point,
            static_cast<System::Byte>(129)));
        this->clientes->ForeColor = System::Drawing::Color::White;
        this->clientes->Location = System::Drawing::Point(0, 495);
        this->clientes->Margin = System::Windows::Forms::Padding(0);
        this->clientes->Name = L"clientes";
        this->clientes->Size = System::Drawing::Size(257, 68);
        this->clientes->TabIndex = 3;
        this->clientes->Text = L"Clientes";
        this->clientes->UseVisualStyleBackColor = true;
        this->clientes->Click += gcnew System::EventHandler(this, &Main_menu::clientes_Click);
        //
        // bt_whiteline_1
        //
        this->bt_whiteline_1->Dock = System::Windows::Forms::DockStyle::Top;
        this->bt_whiteline_1->Image = (cli::safe_cast<System::Drawing::Image^>(resources-
>GetObject(L"bt_whiteline_1.Image")));
        this->bt_whiteline_1->Location = System::Drawing::Point(0, 493);
        this->bt_whiteline_1->Name = L"bt_whiteline_1";
        this->bt_whiteline_1->Size = System::Drawing::Size(257, 2);
        this->bt_whiteline_1->SizeMode =
System::Windows::Forms::PictureBoxSizeMode::StretchImage;
        this->bt_whiteline_1->TabIndex = 4;
        this->bt_whiteline_1->TabStop = false;
        //
        // citas
        //
        this->citas->AutoSize = true;
        this->citas->Dock = System::Windows::Forms::DockStyle::Top;
        this->citas->FlatAppearance->BorderColor =
System::Drawing::Color::FromArgb(static_cast<System::Int32>(static_cast<System::Byte>(255)),
           static_cast<System::Int32>(static_cast<System::Byte>(174)),
static_cast<System::Int32>(static_cast<System::Byte>(30)));
        this->citas->FlatStyle = System::Windows::Forms::FlatStyle::Flat;
        this->citas->Font = (gcnew System::Drawing::Font(L"Microsoft Sans Serif", 22.2F,
System::Drawing::FontStyle::Regular, System::Drawing::GraphicsUnit::Point,
           static_cast<System::Byte>(129)));
        this->citas->ForeColor = System::Drawing::Color::White;
        this->citas->Location = System::Drawing::Point(0, 425);
        this->citas->Margin = System::Windows::Forms::Padding(0);
        this->citas->Name = L"citas";
        this->citas->Size = System::Drawing::Size(257, 68);
        this->citas->TabIndex = 2;
        this->citas->Text = L"Citas";
        this->citas->UseVisualStyleBackColor = true;
        this->citas->Click += gcnew System::EventHandler(this, &Main_menu::citas_Click);
        //
        // panel5
        //
        this->panel5->Controls->Add(this->ft_whiteline_1);
        this->panel5->Controls->Add(this->ft_whiteline_2);
        this->panel5->Controls->Add(this->Foto_usu);
        this->panel5->Dock = System::Windows::Forms::DockStyle::Top;
        this->panel5->Location = System::Drawing::Point(0, 115);
        this->panel5->Margin = System::Windows::Forms::Padding(2);
```

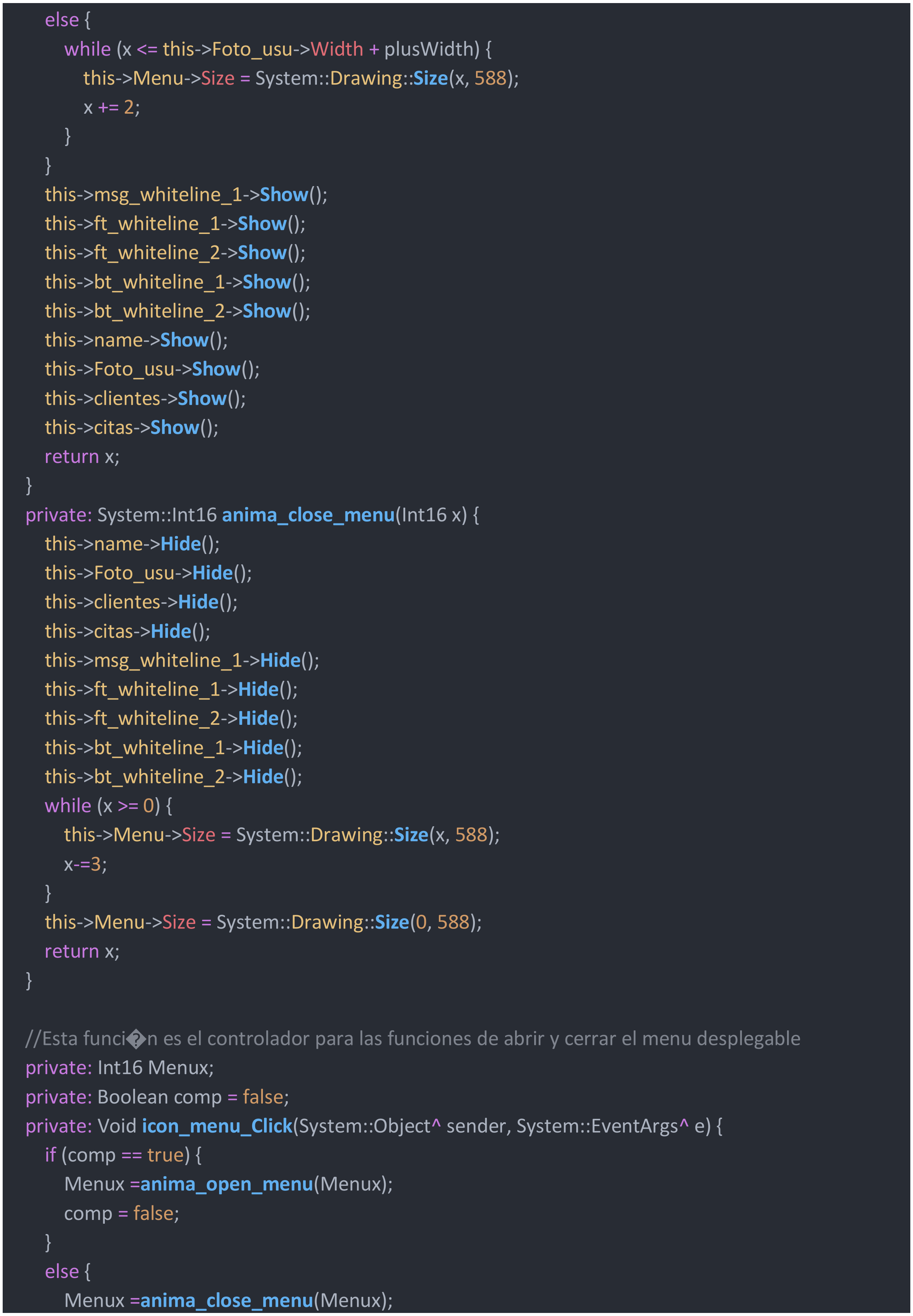
```
	this->panel5->Name = L"panel5";
	this->panel5->Size = System::Drawing::Size(257, 310);
	this->panel5->TabIndex = 1;
	// 
	// Foto_usu
	// 
	this->Foto_usu->Anchor = System::Windows::Forms::AnchorStyles::None;
	this->Foto_usu->Image = (cli::safe_cast<System::Drawing::Image^>(resources-
>GetObject(L"Foto_usu.Image")));
	this->Foto_usu->Location = System::Drawing::Point(11, 38);
	this->Foto_usu->Margin = System::Windows::Forms::Padding(0);
	this->Foto_usu->Name = L"Foto_usu";
	this->Foto_usu->Size = System::Drawing::Size(227, 243);
	this->Foto_usu->SizeMode = System::Windows::Forms::PictureBoxSizeMode::StretchImage;
	this->Foto_usu->TabIndex = 0;
	this->Foto_usu->TabStop = false;
	// 
	// panel4
	// 
	this->panel4->Controls->Add(this->msg_whiteline_1);
	this->panel4->Controls->Add(this->name);
	this->panel4->Dock = System::Windows::Forms::DockStyle::Top;
	this->panel4->Location = System::Drawing::Point(0, 0);
	this->panel4->Margin = System::Windows::Forms::Padding(2);
	this->panel4->Name = L"panel4";
	this->panel4->Size = System::Drawing::Size(257, 115);
	this->panel4->TabIndex = 0;
	// 
	// name
	// 
	this->name->Anchor = 
static_cast<System::Windows::Forms::AnchorStyles>((System::Windows::Forms::AnchorStyles::Left | 
System::Windows::Forms::AnchorStyles::Right));
	this->name->Font = (gcnew System::Drawing::Font(L"Microsoft Sans Serif", 19.8F, 
System::Drawing::FontStyle::Bold, System::Drawing::GraphicsUnit::Point,
	    static_cast<System::Byte>(129)));
	this->name->ForeColor = System::Drawing::Color::White;
	this->name->Location = System::Drawing::Point(45, 29);
	this->name->Margin = System::Windows::Forms::Padding(2, 0, 2, 0);
	this->name->Name = L"name";
	this->name->Size = System::Drawing::Size(158, 62);
	this->name->TabIndex = 1;
	this->name->Text = L"Bienvenido\n Panchito";
	this->name->TextAlign = System::Drawing::ContentAlignment::MiddleCenter;
	// 
```



```
        // ft_whiteline_2
        //
        this->ft_whiteline_2->Dock = System::Windows::Forms::DockStyle::Bottom;
        this->ft_whiteline_2->Image = (cli::safe_cast<System::Drawing::Image^>(resources-
>GetObject(L"ft_whiteline_2.Image")));
        this->ft_whiteline_2->Location = System::Drawing::Point(0, 308);
        this->ft_whiteline_2->Name = L"ft_whiteline_2";
        this->ft_whiteline_2->Size = System::Drawing::Size(257, 2);
        this->ft_whiteline_2->SizeMode =
System::Windows::Forms::PictureBoxSizeMode::StretchImage;
        this->ft_whiteline_2->TabIndex = 5;
        this->ft_whiteline_2->TabStop = false;
        //
        // ft_whiteline_1
        //
        this->ft_whiteline_1->Dock = System::Windows::Forms::DockStyle::Top;
        this->ft_whiteline_1->Image = (cli::safe_cast<System::Drawing::Image^>(resources-
>GetObject(L"ft_whiteline_1.Image")));
        this->ft_whiteline_1->Location = System::Drawing::Point(0, 0);
        this->ft_whiteline_1->Name = L"ft_whiteline_1";
        this->ft_whiteline_1->Size = System::Drawing::Size(257, 2);
        this->ft_whiteline_1->SizeMode =
System::Windows::Forms::PictureBoxSizeMode::StretchImage;
        this->ft_whiteline_1->TabIndex = 6;
        this->ft_whiteline_1->TabStop = false;
        //
        // msg_whiteline_1
        //
        this->msg_whiteline_1->Dock = System::Windows::Forms::DockStyle::Top;
        this->msg_whiteline_1->Image = (cli::safe_cast<System::Drawing::Image^>(resources-
>GetObject(L"msg_whiteline_1.Image")));
        this->msg_whiteline_1->Location = System::Drawing::Point(0, 0);
        this->msg_whiteline_1->Name = L"msg_whiteline_1";
        this->msg_whiteline_1->Size = System::Drawing::Size(257, 2);
        this->msg_whiteline_1->SizeMode =
System::Windows::Forms::PictureBoxSizeMode::StretchImage;
        this->msg_whiteline_1->TabIndex = 5;
        this->msg_whiteline_1->TabStop = false;
        //
        // Main_menu
        //
        this->AutoScaleDimensions = System::Drawing::SizeF(6, 13);
        this->AutoScaleMode = System::Windows::Forms::AutoScaleMode::Font;
        this->BackColor = System::Drawing::SystemColors::Control;
        this->ClientSize = System::Drawing::Size(1264, 681);
```



```
            this->Controls->Add(this->panelShowAll);
            this->Controls->Add(this->panel1);
            this->ForeColor = System::Drawing::SystemColors::ControlText;
            this->Icon = (cli::safe_cast<System::Drawing::Icon^>(resources->GetObject(L"$this.Icon")));
            this->Name = L"Main_menu";
            this->StartPosition = System::Windows::Forms::FormStartPosition::CenterScreen;
            this->Text = L"VetPro";
            this->panel1->ResumeLayout(false);
            this->panel1->PerformLayout();
            (cli::safe_cast<System::ComponentModel::ISupportInitialize^>(this->icon_menu))->EndInit();
            (cli::safe_cast<System::ComponentModel::ISupportInitialize^>(this->icon_vet))->EndInit();
            this->panelShowAll->ResumeLayout(false);
            this->panelShowAll->PerformLayout();
            (cli::safe_cast<System::ComponentModel::ISupportInitialize^>(this->logo))->EndInit();
            this->Menu->ResumeLayout(false);
            this->Menu->PerformLayout();
            (cli::safe_cast<System::ComponentModel::ISupportInitialize^>(this->bt_whiteline_2))-
>EndInit();
            (cli::safe_cast<System::ComponentModel::ISupportInitialize^>(this->bt_whiteline_1))-
>EndInit();
            this->panel5->ResumeLayout(false);
            (cli::safe_cast<System::ComponentModel::ISupportInitialize^>(this->Foto_usu))->EndInit();
            this->panel4->ResumeLayout(false);
            (cli::safe_cast<System::ComponentModel::ISupportInitialize^>(this->ft_whiteline_2))->EndInit();
            (cli::safe_cast<System::ComponentModel::ISupportInitialize^>(this->ft_whiteline_1))->EndInit();
            (cli::safe_cast<System::ComponentModel::ISupportInitialize^>(this->msg_whiteline_1))-
>EndInit();
            this->ResumeLayout(false);

        }

#pragma endregion
    //Almcena el nombre del usuario de la sesion
    private: String^ user;

    //Estas dos funciones se encargan de la animaci�n de abrir y cerrar la ventana desplegable del
menu principal
    private: Int16 plusWidth;
    private: Int16 anima_open_menu(Int16 x) {
        if (plusWidth == 50) {
            while (x <= this->name->Width + plusWidth) {
                this->Menu->Size = System::Drawing::Size(x, 588);
                x += 2;
            }
        }
```



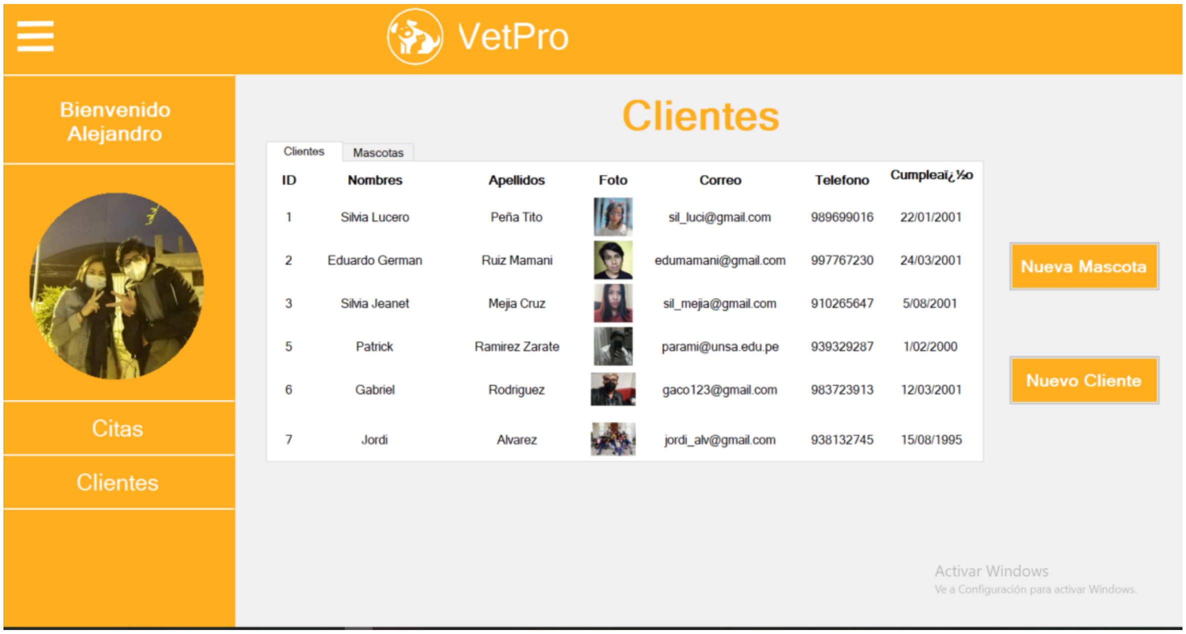
```
    else {
        while (x <= this->Foto_usu->Width + plusWidth) {
            this->Menu->Size = System::Drawing::Size(x, 588);
            x += 2;
        }
    }
    this->msg_whiteline_1->Show();
    this->ft_whiteline_1->Show();
    this->ft_whiteline_2->Show();
    this->bt_whiteline_1->Show();
    this->bt_whiteline_2->Show();
    this->name->Show();
    this->Foto_usu->Show();
    this->clientes->Show();
    this->citas->Show();
    return x;
}
private: System::Int16 anima_close_menu(Int16 x) {
    this->name->Hide();
    this->Foto_usu->Hide();
    this->clientes->Hide();
    this->citas->Hide();
    this->msg_whiteline_1->Hide();
    this->ft_whiteline_1->Hide();
    this->ft_whiteline_2->Hide();
    this->bt_whiteline_1->Hide();
    this->bt_whiteline_2->Hide();
    while (x >= 0) {
        this->Menu->Size = System::Drawing::Size(x, 588);
        x-=3;
    }
    this->Menu->Size = System::Drawing::Size(0, 588);
    return x;
}

//Esta funci�n es el controlador para las funciones de abrir y cerrar el menu desplegable
private: Int16 Menux;
private: Boolean comp = false;
private: Void icon_menu_Click(System::Object^ sender, System::EventArgs^ e) {
    if (comp == true) {
        Menux =anima_open_menu(Menux);
        comp = false;
    }
    else {
        Menux =anima_close_menu(Menux);
```

```
            comp = true;
        }
    }
```

Dentro del Windows form se creó un elemento de tipo panel en el que se mostrará los formularios de clientes y citas, evitando la sobrecarga de ventanas. Se crea un puntero a un formulario que inica como nullptr. En el caso se haga click en alguno de los botos de citas o clientes, apuntará a ese nuevo formulario. En caso el panel se encuentre con un form abierto, este se cerrará y dará paso al nuevo form solicitado. Esta función es llamado al hacer click en Citas o clientes

```
    //Se encarga de gestionar las ventanas que se abren en el panelShowAll
    private: System::Windows::Forms::Form^ f_activo = nullptr;
    private: Void abrirfh(System::Windows::Forms::Form^ f_hijo) {
        if (this->f_activo != nullptr) {
            this->f_activo->Close();
        }
        this->f_activo = f_hijo;
        f_hijo->TopLevel = false;
        f_hijo->FormBorderStyle = System::Windows::Forms::FormBorderStyle::None;
        f_hijo->Dock = System::Windows::Forms::DockStyle::Fill;
        this->panelShowAll->Controls->Add(f_hijo);
        this->panelShowAll->Tag = f_hijo;
        f_hijo->BringToFront();
        f_hijo->Show();
    }

    //Abre la ventana cita dentro del panelShowAll
    private: Void citas_Click(System::Object^ sender, System::EventArgs^ e) {
        abrirfh(gcnew VetPro::Cita());

    }

    //Abre la ventana cliente dentro del panelShowAll
    private: Void clientes_Click(System::Object^ sender, System::EventArgs^ e) {
        abrirfh(gcnew VetPro::Cliente());
    }
};
}
```

Clientes



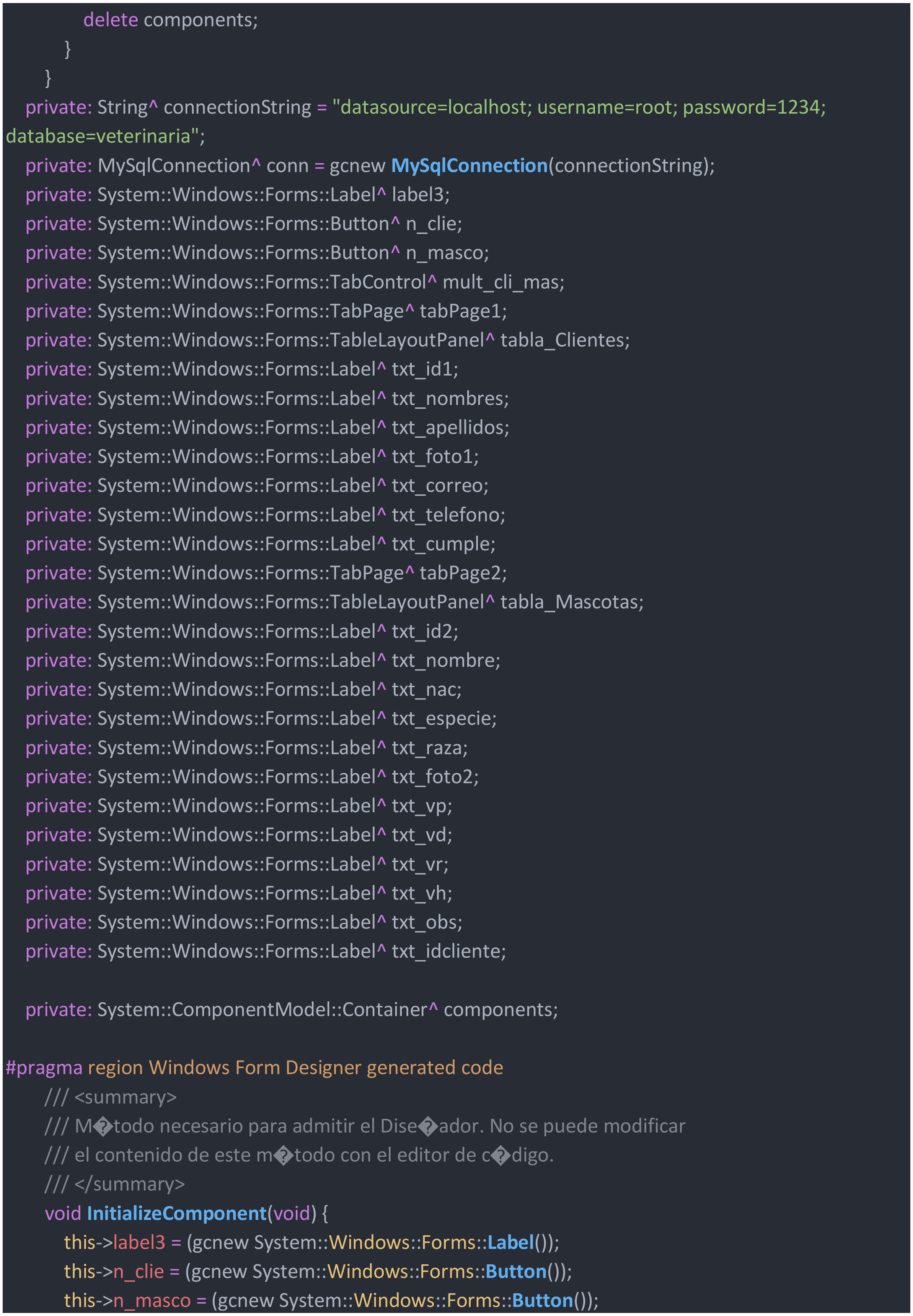
Clientes

Clientes | Mascotas

ID	Nombres	Apellidos	Foto	Correo	Telefono	Cumpleaï¿½o
1	Silvia Lucero	Peña Tito		sil_luci@gmail.com	989699016	22/01/2001
2	Eduardo German	Ruiz Mamani		edumamani@gmail.com	997767230	24/03/2001
3	Silvia Jeanet	Mejia Cruz		sil_mejia@gmail.com	910265647	5/08/2001
5	Patrick	Ramirez Zarate		parami@unsa.edu.pe	939329287	1/02/2000
6	Gabriel	Rodriguez		gaco123@gmail.com	983723913	12/03/2001
7	Jordi	Alvarez		jordi_alv@gmail.com	938132745	15/08/1995

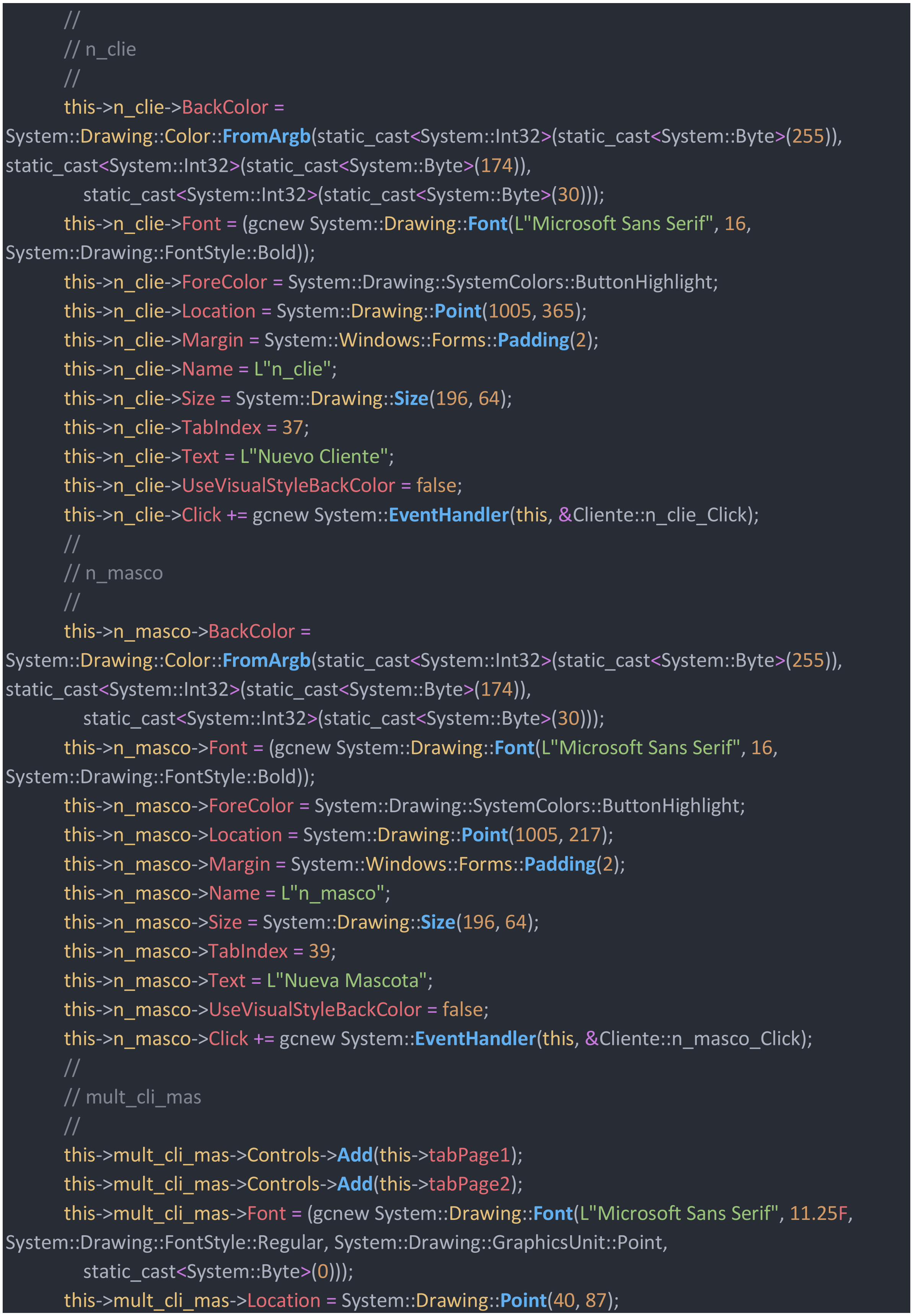
Al inicializar el form se hace el llamado a 2 funciones para cargar los datos de los clientes y mascotas en una tabla visible para le usuario.

```
#pragma once
#include "addCliente.h"
#include "addMascota.h"

namespace VetPro {

  using namespace System;
  using namespace System::ComponentModel;
  using namespace System::Collections;
  using namespace System::Windows::Forms;
  using namespace System::Data;
  using namespace System::Drawing;
  using namespace MySql::Data::MySqlClient;

  public ref class Cliente : public System::Windows::Forms::Form {
  public:
    Cliente(void) {
      InitializeComponent();
      load_TablaClientes();
      load_TablaMascotas();
    }
  protected:
    ~Cliente() {
      if (components) {
```

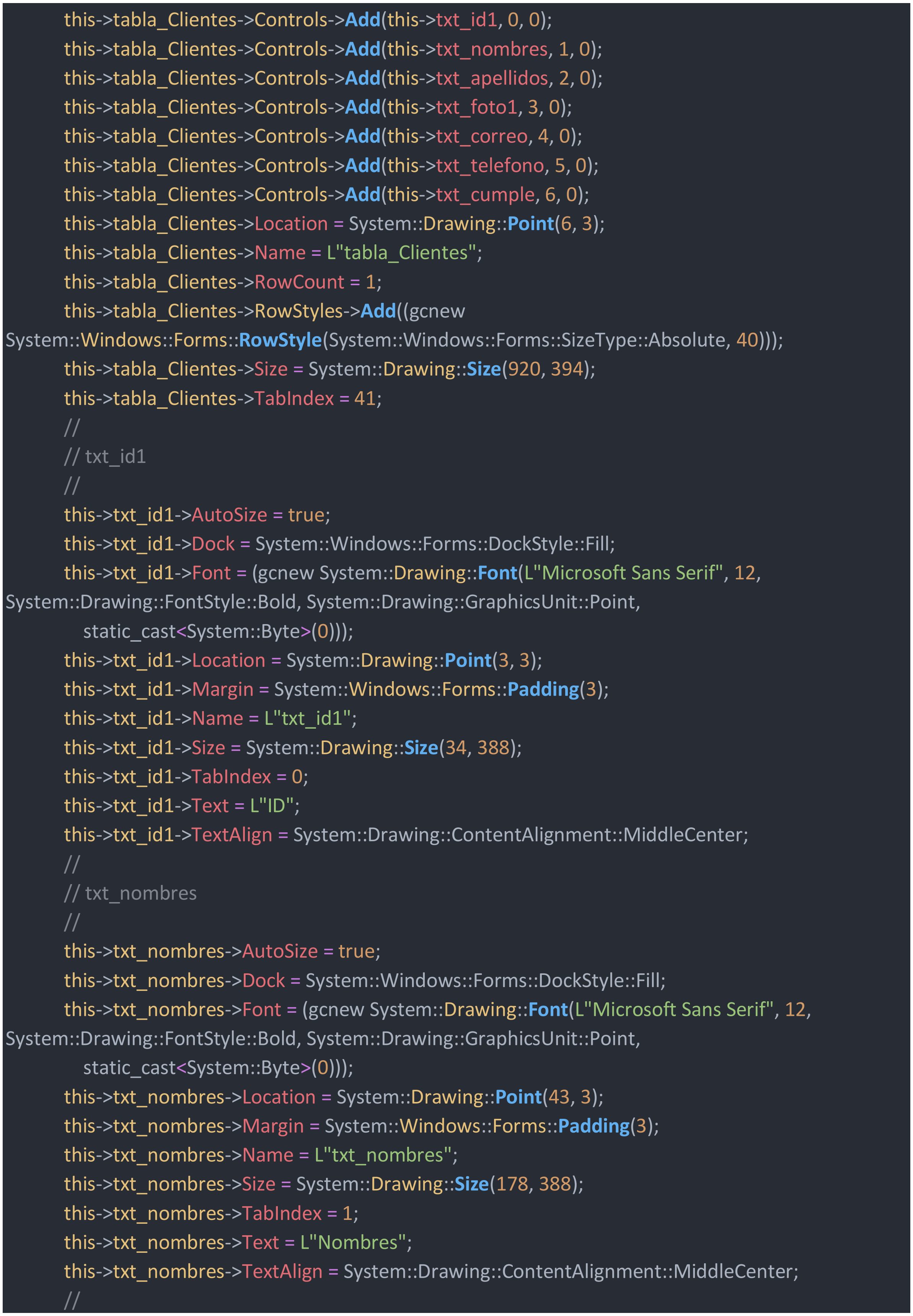


```
        delete components;
      }
    }
  private: String^ connectionString = "datasource=localhost; username=root; password=1234;
database=veterinaria";
  private: MySqlConnection^ conn = gcnew MySqlConnection(connectionString);
  private: System::Windows::Forms::Label^ label3;
  private: System::Windows::Forms::Button^ n_clie;
  private: System::Windows::Forms::Button^ n_masco;
  private: System::Windows::Forms::TabControl^ mult_cli_mas;
  private: System::Windows::Forms::TabPage^ tabPage1;
  private: System::Windows::Forms::TableLayoutPanel^ tabla_Clientes;
  private: System::Windows::Forms::Label^ txt_id1;
  private: System::Windows::Forms::Label^ txt_nombres;
  private: System::Windows::Forms::Label^ txt_apellidos;
  private: System::Windows::Forms::Label^ txt_foto1;
  private: System::Windows::Forms::Label^ txt_correo;
  private: System::Windows::Forms::Label^ txt_telefono;
  private: System::Windows::Forms::Label^ txt_cumple;
  private: System::Windows::Forms::TabPage^ tabPage2;
  private: System::Windows::Forms::TableLayoutPanel^ tabla_Mascotas;
  private: System::Windows::Forms::Label^ txt_id2;
  private: System::Windows::Forms::Label^ txt_nombre;
  private: System::Windows::Forms::Label^ txt_nac;
  private: System::Windows::Forms::Label^ txt_especie;
  private: System::Windows::Forms::Label^ txt_raza;
  private: System::Windows::Forms::Label^ txt_foto2;
  private: System::Windows::Forms::Label^ txt_vp;
  private: System::Windows::Forms::Label^ txt_vd;
  private: System::Windows::Forms::Label^ txt_vr;
  private: System::Windows::Forms::Label^ txt_vh;
  private: System::Windows::Forms::Label^ txt_obs;
  private: System::Windows::Forms::Label^ txt_idcliente;

  private: System::ComponentModel::Container^ components;

#pragma region Windows Form Designer generated code
    /// <summary>
    /// M�todo necesario para admitir el Dise�ador. No se puede modificar
    /// el contenido de este m�todo con el editor de c�digo.
    /// </summary>
    void InitializeComponent(void) {
      this->label3 = (gcnew System::Windows::Forms::Label());
      this->n_clie = (gcnew System::Windows::Forms::Button());
      this->n_masco = (gcnew System::Windows::Forms::Button());
```

```
        this->mult_cli_mas = (gcnew System::Windows::Forms::TabControl());
        this->tabPage1 = (gcnew System::Windows::Forms::TabPage());
        this->tabla_Clientes = (gcnew System::Windows::Forms::TableLayoutPanel());
        this->txt_id1 = (gcnew System::Windows::Forms::Label());
        this->txt_nombres = (gcnew System::Windows::Forms::Label());
        this->txt_apellidos = (gcnew System::Windows::Forms::Label());
        this->txt_foto1 = (gcnew System::Windows::Forms::Label());
        this->txt_correo = (gcnew System::Windows::Forms::Label());
        this->txt_telefono = (gcnew System::Windows::Forms::Label());
        this->txt_cumple = (gcnew System::Windows::Forms::Label());
        this->tabPage2 = (gcnew System::Windows::Forms::TabPage());
        this->tabla_Mascotas = (gcnew System::Windows::Forms::TableLayoutPanel());
        this->txt_id2 = (gcnew System::Windows::Forms::Label());
        this->txt_nombre = (gcnew System::Windows::Forms::Label());
        this->txt_nac = (gcnew System::Windows::Forms::Label());
        this->txt_especie = (gcnew System::Windows::Forms::Label());
        this->txt_raza = (gcnew System::Windows::Forms::Label());
        this->txt_foto2 = (gcnew System::Windows::Forms::Label());
        this->txt_vp = (gcnew System::Windows::Forms::Label());
        this->txt_vd = (gcnew System::Windows::Forms::Label());
        this->txt_vr = (gcnew System::Windows::Forms::Label());
        this->txt_vh = (gcnew System::Windows::Forms::Label());
        this->txt_obs = (gcnew System::Windows::Forms::Label());
        this->txt_idcliente = (gcnew System::Windows::Forms::Label());
        this->mult_cli_mas->SuspendLayout();
        this->tabPage1->SuspendLayout();
        this->tabla_Clientes->SuspendLayout();
        this->tabPage2->SuspendLayout();
        this->tabla_Mascotas->SuspendLayout();
        this->SuspendLayout();
        //
        // label3
        //
        this->label3->Font = (gcnew System::Drawing::Font(L"Microsoft Sans Serif", 40.2F,
System::Drawing::FontStyle::Bold, System::Drawing::GraphicsUnit::Point,
            static_cast<System::Byte>(0)));
        this->label3->ForeColor =
System::Drawing::Color::FromArgb(static_cast<System::Int32>(static_cast<System::Byte>(255)),
static_cast<System::Int32>(static_cast<System::Byte>(174)),
            static_cast<System::Int32>(static_cast<System::Byte>(30)));
        this->label3->Location = System::Drawing::Point(491, 21);
        this->label3->Name = L"label3";
        this->label3->Size = System::Drawing::Size(234, 64);
        this->label3->TabIndex = 38;
        this->label3->Text = L"Clientes";
```



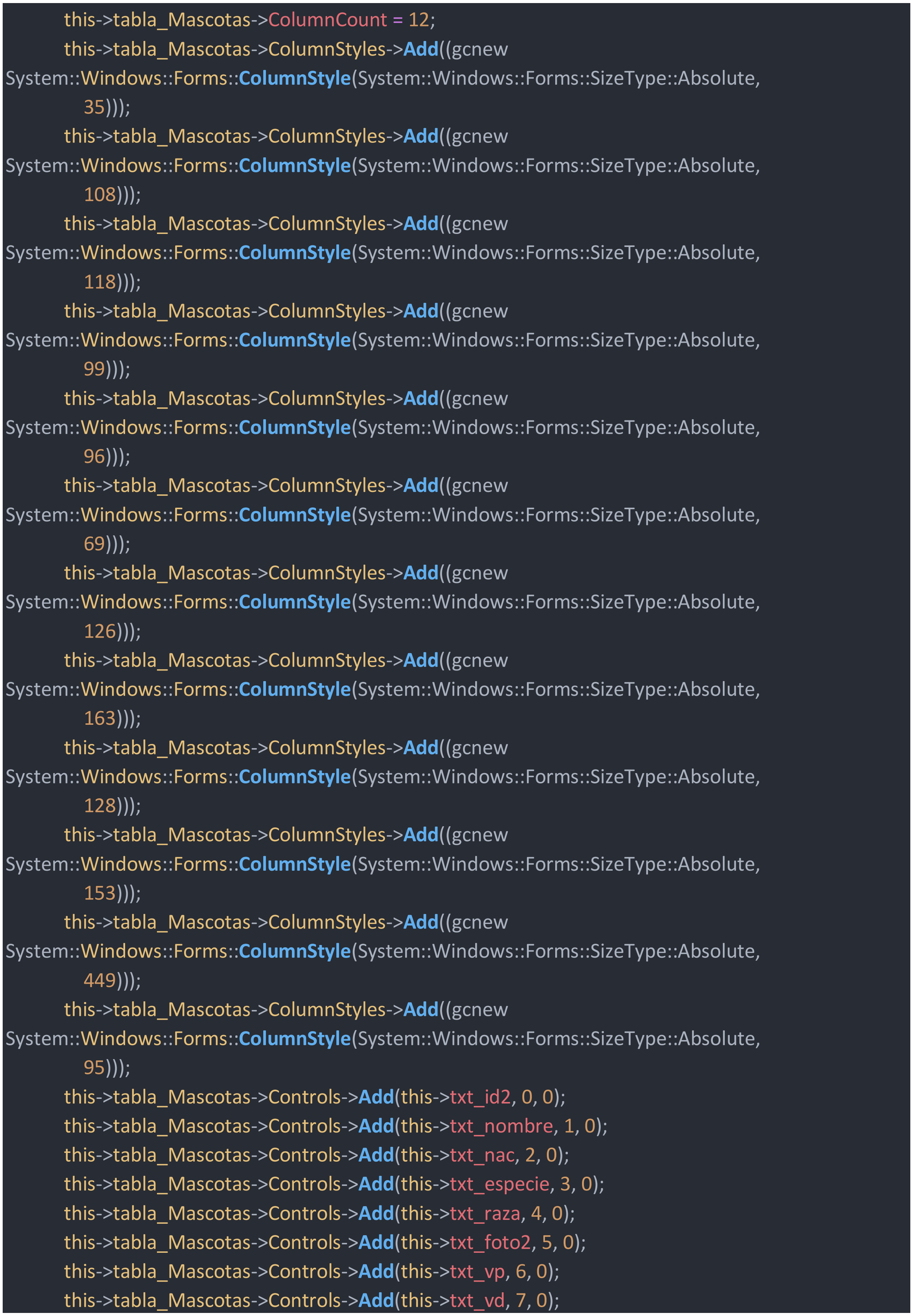
```
        //
        // n_clie
        //
        this->n_clie->BackColor =
System::Drawing::Color::FromArgb(static_cast<System::Int32>(static_cast<System::Byte>(255)),
static_cast<System::Int32>(static_cast<System::Byte>(174)),
            static_cast<System::Int32>(static_cast<System::Byte>(30)));
        this->n_clie->Font = (gcnew System::Drawing::Font(L"Microsoft Sans Serif", 16,
System::Drawing::FontStyle::Bold));
        this->n_clie->ForeColor = System::Drawing::SystemColors::ButtonHighlight;
        this->n_clie->Location = System::Drawing::Point(1005, 365);
        this->n_clie->Margin = System::Windows::Forms::Padding(2);
        this->n_clie->Name = L"n_clie";
        this->n_clie->Size = System::Drawing::Size(196, 64);
        this->n_clie->TabIndex = 37;
        this->n_clie->Text = L"Nuevo Cliente";
        this->n_clie->UseVisualStyleBackColor = false;
        this->n_clie->Click += gcnew System::EventHandler(this, &Cliente::n_clie_Click);
        //
        // n_masco
        //
        this->n_masco->BackColor =
System::Drawing::Color::FromArgb(static_cast<System::Int32>(static_cast<System::Byte>(255)),
static_cast<System::Int32>(static_cast<System::Byte>(174)),
            static_cast<System::Int32>(static_cast<System::Byte>(30)));
        this->n_masco->Font = (gcnew System::Drawing::Font(L"Microsoft Sans Serif", 16,
System::Drawing::FontStyle::Bold));
        this->n_masco->ForeColor = System::Drawing::SystemColors::ButtonHighlight;
        this->n_masco->Location = System::Drawing::Point(1005, 217);
        this->n_masco->Margin = System::Windows::Forms::Padding(2);
        this->n_masco->Name = L"n_masco";
        this->n_masco->Size = System::Drawing::Size(196, 64);
        this->n_masco->TabIndex = 39;
        this->n_masco->Text = L"Nueva Mascota";
        this->n_masco->UseVisualStyleBackColor = false;
        this->n_masco->Click += gcnew System::EventHandler(this, &Cliente::n_masco_Click);
        //
        // mult_cli_mas
        //
        this->mult_cli_mas->Controls->Add(this->tabPage1);
        this->mult_cli_mas->Controls->Add(this->tabPage2);
        this->mult_cli_mas->Font = (gcnew System::Drawing::Font(L"Microsoft Sans Serif", 11.25F,
System::Drawing::FontStyle::Regular, System::Drawing::GraphicsUnit::Point,
            static_cast<System::Byte>(0)));
        this->mult_cli_mas->Location = System::Drawing::Point(40, 87);
        this->mult_cli_mas->Name = L"mult_cli_mas";
        this->mult_cli_mas->SelectedIndex = 0;
        this->mult_cli_mas->Size = System::Drawing::Size(934, 417);
        this->mult_cli_mas->SizeMode = System::Windows::Forms::TabSizeMode::Fixed;
        this->mult_cli_mas->TabIndex = 41;
        //
        // tabPage1
        //
        this->tabPage1->Controls->Add(this->tabla_Clientes);
        this->tabPage1->Font = (gcnew System::Drawing::Font(L"Microsoft Sans Serif", 11.25F,
System::Drawing::FontStyle::Regular, System::Drawing::GraphicsUnit::Point,
           static_cast<System::Byte>(0)));
        this->tabPage1->Location = System::Drawing::Point(4, 27);
        this->tabPage1->Name = L"tabPage1";
        this->tabPage1->Padding = System::Windows::Forms::Padding(3);
        this->tabPage1->Size = System::Drawing::Size(926, 386);
        this->tabPage1->TabIndex = 0;
        this->tabPage1->Text = L"Clientes";
        this->tabPage1->UseVisualStyleBackColor = true;
        //
        // tabla_Clientes
        //
        this->tabla_Clientes->AutoScroll = true;
        this->tabla_Clientes->ColumnCount = 7;
        this->tabla_Clientes->ColumnStyles->Add((gcnew
System::Windows::Forms::ColumnStyle(System::Windows::Forms::SizeType::Absolute,
           40)));
        this->tabla_Clientes->ColumnStyles->Add((gcnew
System::Windows::Forms::ColumnStyle(System::Windows::Forms::SizeType::Percent,
           20.94361F)));
        this->tabla_Clientes->ColumnStyles->Add((gcnew
System::Windows::Forms::ColumnStyle(System::Windows::Forms::SizeType::Percent,
           21.05869F)));
        this->tabla_Clientes->ColumnStyles->Add((gcnew
System::Windows::Forms::ColumnStyle(System::Windows::Forms::SizeType::Percent,
           7.364788F)));
        this->tabla_Clientes->ColumnStyles->Add((gcnew
System::Windows::Forms::ColumnStyle(System::Windows::Forms::SizeType::Percent,
           24.51094F)));
        this->tabla_Clientes->ColumnStyles->Add((gcnew
System::Windows::Forms::ColumnStyle(System::Windows::Forms::SizeType::Percent,
           11.62255F)));
        this->tabla_Clientes->ColumnStyles->Add((gcnew
System::Windows::Forms::ColumnStyle(System::Windows::Forms::SizeType::Percent,
           14.49942F)));
```



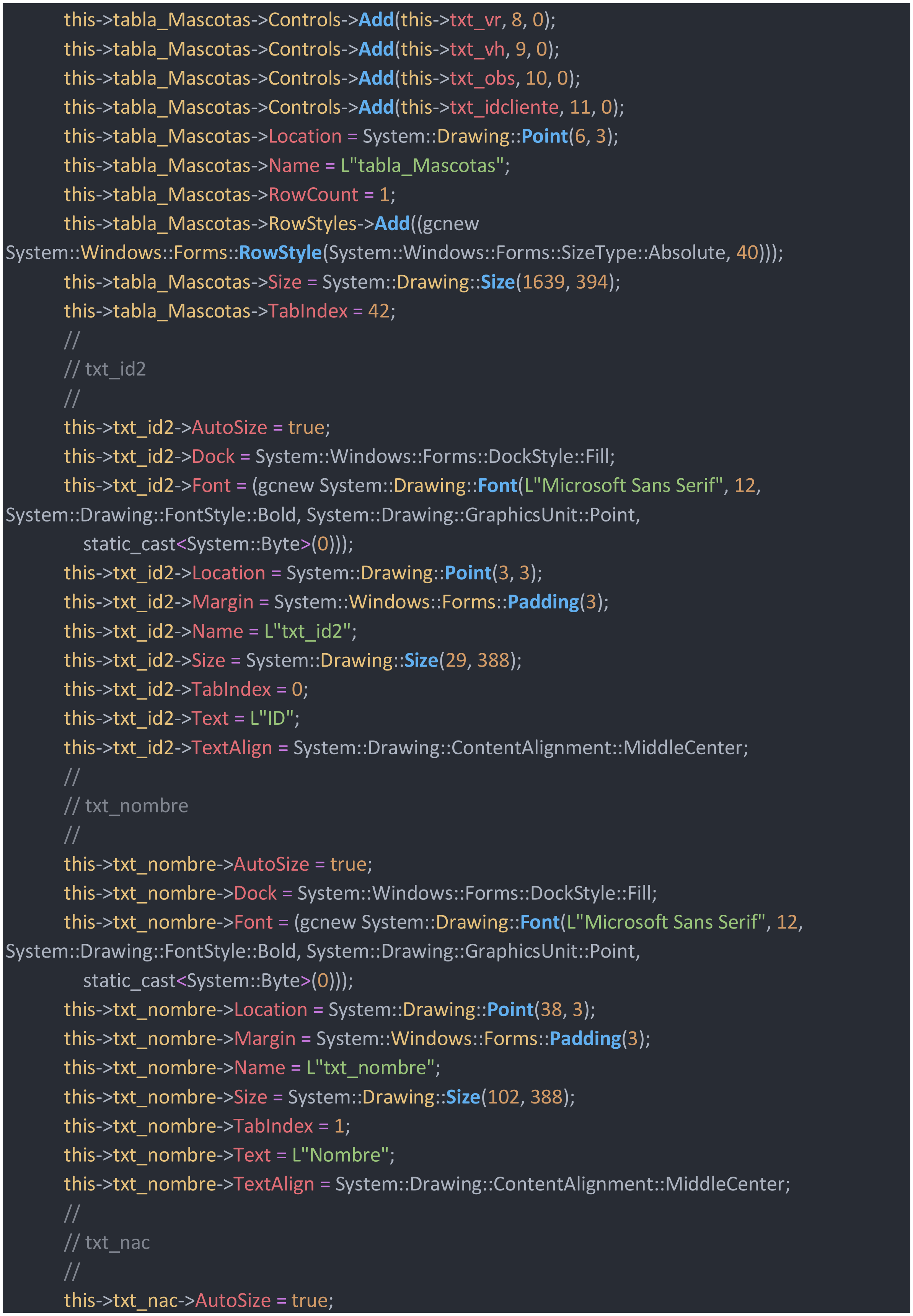
```
	this->tabla_Clientes->Controls->Add(this->txt_id1, 0, 0);
	this->tabla_Clientes->Controls->Add(this->txt_nombres, 1, 0);
	this->tabla_Clientes->Controls->Add(this->txt_apellidos, 2, 0);
	this->tabla_Clientes->Controls->Add(this->txt_foto1, 3, 0);
	this->tabla_Clientes->Controls->Add(this->txt_correo, 4, 0);
	this->tabla_Clientes->Controls->Add(this->txt_telefono, 5, 0);
	this->tabla_Clientes->Controls->Add(this->txt_cumple, 6, 0);
	this->tabla_Clientes->Location = System::Drawing::Point(6, 3);
	this->tabla_Clientes->Name = L"tabla_Clientes";
	this->tabla_Clientes->RowCount = 1;
	this->tabla_Clientes->RowStyles->Add((gcnew
System::Windows::Forms::RowStyle(System::Windows::Forms::SizeType::Absolute, 40)));
	this->tabla_Clientes->Size = System::Drawing::Size(920, 394);
	this->tabla_Clientes->TabIndex = 41;
	//
	// txt_id1
	//
	this->txt_id1->AutoSize = true;
	this->txt_id1->Dock = System::Windows::Forms::DockStyle::Fill;
	this->txt_id1->Font = (gcnew System::Drawing::Font(L"Microsoft Sans Serif", 12,
System::Drawing::FontStyle::Bold, System::Drawing::GraphicsUnit::Point,
		static_cast<System::Byte>(0)));
	this->txt_id1->Location = System::Drawing::Point(3, 3);
	this->txt_id1->Margin = System::Windows::Forms::Padding(3);
	this->txt_id1->Name = L"txt_id1";
	this->txt_id1->Size = System::Drawing::Size(34, 388);
	this->txt_id1->TabIndex = 0;
	this->txt_id1->Text = L"ID";
	this->txt_id1->TextAlign = System::Drawing::ContentAlignment::MiddleCenter;
	//
	// txt_nombres
	//
	this->txt_nombres->AutoSize = true;
	this->txt_nombres->Dock = System::Windows::Forms::DockStyle::Fill;
	this->txt_nombres->Font = (gcnew System::Drawing::Font(L"Microsoft Sans Serif", 12,
System::Drawing::FontStyle::Bold, System::Drawing::GraphicsUnit::Point,
		static_cast<System::Byte>(0)));
	this->txt_nombres->Location = System::Drawing::Point(43, 3);
	this->txt_nombres->Margin = System::Windows::Forms::Padding(3);
	this->txt_nombres->Name = L"txt_nombres";
	this->txt_nombres->Size = System::Drawing::Size(178, 388);
	this->txt_nombres->TabIndex = 1;
	this->txt_nombres->Text = L"Nombres";
	this->txt_nombres->TextAlign = System::Drawing::ContentAlignment::MiddleCenter;
	//
```

```
			// txt_apellidos
			//
			this->txt_apellidos->AutoSize = true;
			this->txt_apellidos->Dock = System::Windows::Forms::DockStyle::Fill;
			this->txt_apellidos->Font = (gcnew System::Drawing::Font(L"Microsoft Sans Serif", 12,
System::Drawing::FontStyle::Bold, System::Drawing::GraphicsUnit::Point,
				static_cast<System::Byte>(0)));
			this->txt_apellidos->Location = System::Drawing::Point(227, 3);
			this->txt_apellidos->Margin = System::Windows::Forms::Padding(3);
			this->txt_apellidos->Name = L"txt_apellidos";
			this->txt_apellidos->Size = System::Drawing::Size(179, 388);
			this->txt_apellidos->TabIndex = 2;
			this->txt_apellidos->Text = L"Apellidos";
			this->txt_apellidos->TextAlign = System::Drawing::ContentAlignment::MiddleCenter;
			//
			// txt_foto1
			//
			this->txt_foto1->AutoSize = true;
			this->txt_foto1->Dock = System::Windows::Forms::DockStyle::Fill;
			this->txt_foto1->Font = (gcnew System::Drawing::Font(L"Microsoft Sans Serif", 12,
System::Drawing::FontStyle::Bold, System::Drawing::GraphicsUnit::Point,
				static_cast<System::Byte>(0)));
			this->txt_foto1->Location = System::Drawing::Point(412, 3);
			this->txt_foto1->Margin = System::Windows::Forms::Padding(3);
			this->txt_foto1->Name = L"txt_foto1";
			this->txt_foto1->Size = System::Drawing::Size(58, 388);
			this->txt_foto1->TabIndex = 3;
			this->txt_foto1->Text = L"Foto";
			this->txt_foto1->TextAlign = System::Drawing::ContentAlignment::MiddleCenter;
			//
			// txt_correo
			//
			this->txt_correo->AutoSize = true;
			this->txt_correo->Dock = System::Windows::Forms::DockStyle::Fill;
			this->txt_correo->Font = (gcnew System::Drawing::Font(L"Microsoft Sans Serif", 12,
System::Drawing::FontStyle::Bold, System::Drawing::GraphicsUnit::Point,
				static_cast<System::Byte>(0)));
			this->txt_correo->Location = System::Drawing::Point(476, 3);
			this->txt_correo->Margin = System::Windows::Forms::Padding(3);
			this->txt_correo->Name = L"txt_correo";
			this->txt_correo->Size = System::Drawing::Size(209, 388);
			this->txt_correo->TabIndex = 4;
			this->txt_correo->Text = L"Correo";
			this->txt_correo->TextAlign = System::Drawing::ContentAlignment::MiddleCenter;
			//
```

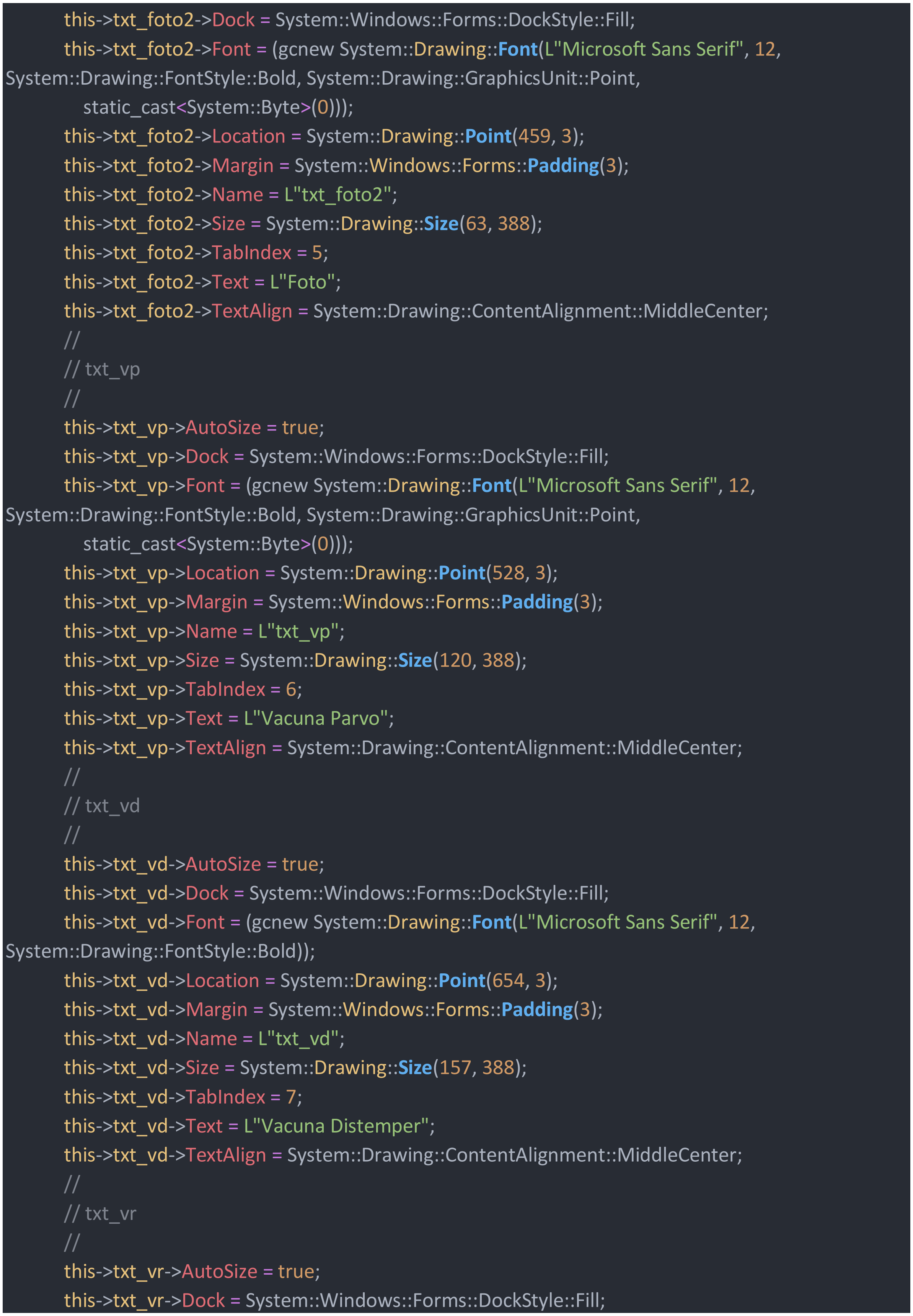
```
        // txt_telefono
        //
        this->txt_telefono->AutoSize = true;
        this->txt_telefono->Dock = System::Windows::Forms::DockStyle::Fill;
        this->txt_telefono->Font = (gcnew System::Drawing::Font(L"Microsoft Sans Serif", 12,
System::Drawing::FontStyle::Bold, System::Drawing::GraphicsUnit::Point,
          static_cast<System::Byte>(0)));
        this->txt_telefono->Location = System::Drawing::Point(691, 3);
        this->txt_telefono->Margin = System::Windows::Forms::Padding(3);
        this->txt_telefono->Name = L"txt_telefono";
        this->txt_telefono->Size = System::Drawing::Size(96, 388);
        this->txt_telefono->TabIndex = 5;
        this->txt_telefono->Text = L"Telefono";
        this->txt_telefono->TextAlign = System::Drawing::ContentAlignment::MiddleCenter;
        //
        // txt_cumple
        //
        this->txt_cumple->AutoSize = true;
        this->txt_cumple->Dock = System::Windows::Forms::DockStyle::Fill;
        this->txt_cumple->Font = (gcnew System::Drawing::Font(L"Microsoft Sans Serif", 12,
System::Drawing::FontStyle::Bold, System::Drawing::GraphicsUnit::Point,
          static_cast<System::Byte>(0)));
        this->txt_cumple->Location = System::Drawing::Point(793, 3);
        this->txt_cumple->Margin = System::Windows::Forms::Padding(3);
        this->txt_cumple->Name = L"txt_cumple";
        this->txt_cumple->Size = System::Drawing::Size(124, 388);
        this->txt_cumple->TabIndex = 6;
        this->txt_cumple->Text = L"Cumplea�os";
        this->txt_cumple->TextAlign = System::Drawing::ContentAlignment::MiddleCenter;
        //
        // tabPage2
        //
        this->tabPage2->AutoScroll = true;
        this->tabPage2->Controls->Add(this->tabla_Mascotas);
        this->tabPage2->Location = System::Drawing::Point(4, 27);
        this->tabPage2->Name = L"tabPage2";
        this->tabPage2->Padding = System::Windows::Forms::Padding(3);
        this->tabPage2->Size = System::Drawing::Size(926, 386);
        this->tabPage2->TabIndex = 1;
        this->tabPage2->Text = L"Mascotas";
        this->tabPage2->UseVisualStyleBackColor = true;
        //
        // tabla_Mascotas
        //
        this->tabla_Mascotas->AutoScroll = true;
```



```
        this->tabla_Mascotas->ColumnCount = 12;
        this->tabla_Mascotas->ColumnStyles->Add((gcnew
System::Windows::Forms::ColumnStyle(System::Windows::Forms::SizeType::Absolute,
          35)));
        this->tabla_Mascotas->ColumnStyles->Add((gcnew
System::Windows::Forms::ColumnStyle(System::Windows::Forms::SizeType::Absolute,
          108)));
        this->tabla_Mascotas->ColumnStyles->Add((gcnew
System::Windows::Forms::ColumnStyle(System::Windows::Forms::SizeType::Absolute,
          118)));
        this->tabla_Mascotas->ColumnStyles->Add((gcnew
System::Windows::Forms::ColumnStyle(System::Windows::Forms::SizeType::Absolute,
          99)));
        this->tabla_Mascotas->ColumnStyles->Add((gcnew
System::Windows::Forms::ColumnStyle(System::Windows::Forms::SizeType::Absolute,
          96)));
        this->tabla_Mascotas->ColumnStyles->Add((gcnew
System::Windows::Forms::ColumnStyle(System::Windows::Forms::SizeType::Absolute,
          69)));
        this->tabla_Mascotas->ColumnStyles->Add((gcnew
System::Windows::Forms::ColumnStyle(System::Windows::Forms::SizeType::Absolute,
          126)));
        this->tabla_Mascotas->ColumnStyles->Add((gcnew
System::Windows::Forms::ColumnStyle(System::Windows::Forms::SizeType::Absolute,
          163)));
        this->tabla_Mascotas->ColumnStyles->Add((gcnew
System::Windows::Forms::ColumnStyle(System::Windows::Forms::SizeType::Absolute,
          128)));
        this->tabla_Mascotas->ColumnStyles->Add((gcnew
System::Windows::Forms::ColumnStyle(System::Windows::Forms::SizeType::Absolute,
          153)));
        this->tabla_Mascotas->ColumnStyles->Add((gcnew
System::Windows::Forms::ColumnStyle(System::Windows::Forms::SizeType::Absolute,
          449)));
        this->tabla_Mascotas->ColumnStyles->Add((gcnew
System::Windows::Forms::ColumnStyle(System::Windows::Forms::SizeType::Absolute,
          95)));
        this->tabla_Mascotas->Controls->Add(this->txt_id2, 0, 0);
        this->tabla_Mascotas->Controls->Add(this->txt_nombre, 1, 0);
        this->tabla_Mascotas->Controls->Add(this->txt_nac, 2, 0);
        this->tabla_Mascotas->Controls->Add(this->txt_especie, 3, 0);
        this->tabla_Mascotas->Controls->Add(this->txt_raza, 4, 0);
        this->tabla_Mascotas->Controls->Add(this->txt_foto2, 5, 0);
        this->tabla_Mascotas->Controls->Add(this->txt_vp, 6, 0);
        this->tabla_Mascotas->Controls->Add(this->txt_vd, 7, 0);
```



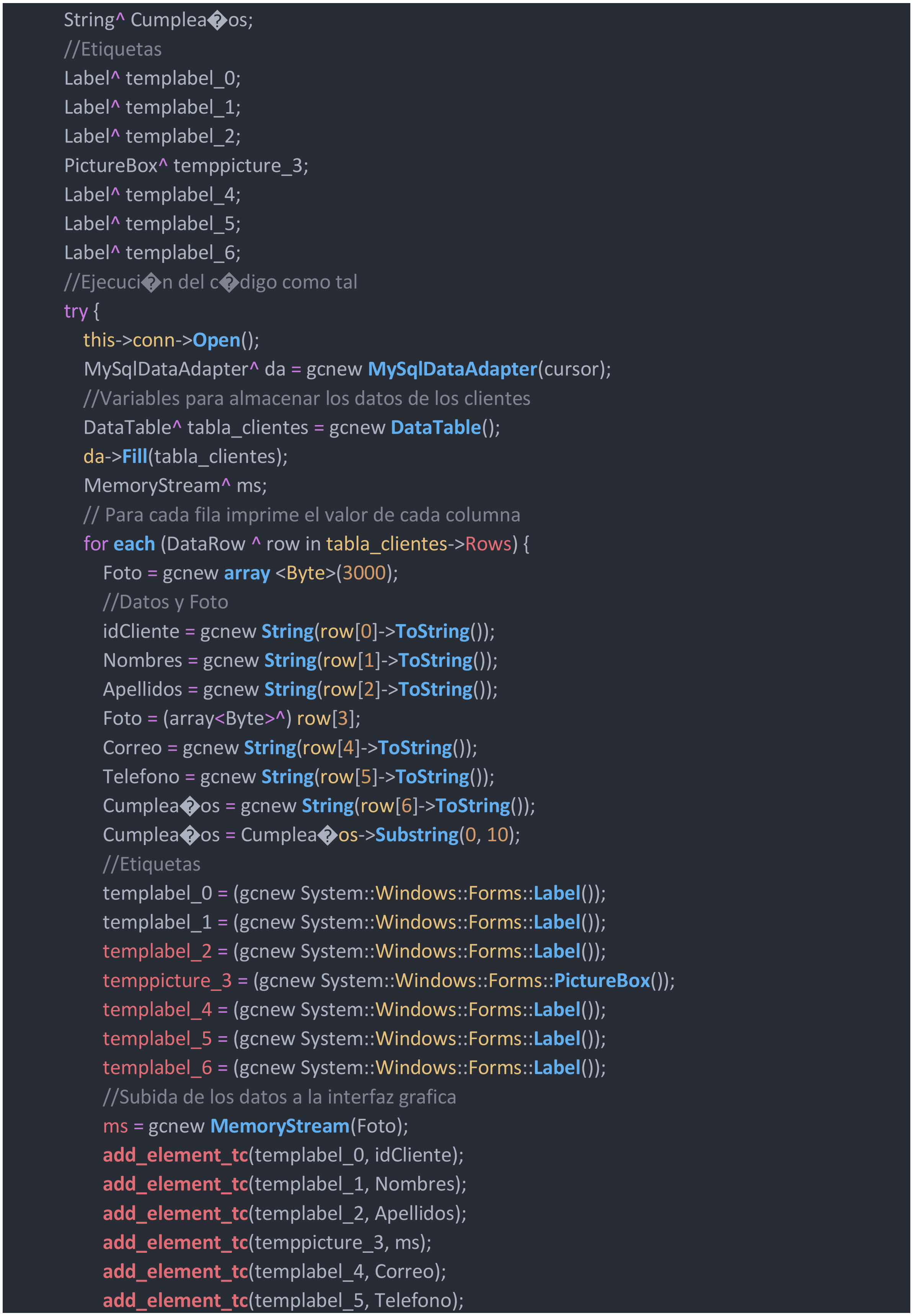
```
        this->tabla_Mascotas->Controls->Add(this->txt_vr, 8, 0);
        this->tabla_Mascotas->Controls->Add(this->txt_vh, 9, 0);
        this->tabla_Mascotas->Controls->Add(this->txt_obs, 10, 0);
        this->tabla_Mascotas->Controls->Add(this->txt_idcliente, 11, 0);
        this->tabla_Mascotas->Location = System::Drawing::Point(6, 3);
        this->tabla_Mascotas->Name = L"tabla_Mascotas";
        this->tabla_Mascotas->RowCount = 1;
        this->tabla_Mascotas->RowStyles->Add((gcnew
System::Windows::Forms::RowStyle(System::Windows::Forms::SizeType::Absolute, 40)));
        this->tabla_Mascotas->Size = System::Drawing::Size(1639, 394);
        this->tabla_Mascotas->TabIndex = 42;
        //
        // txt_id2
        //
        this->txt_id2->AutoSize = true;
        this->txt_id2->Dock = System::Windows::Forms::DockStyle::Fill;
        this->txt_id2->Font = (gcnew System::Drawing::Font(L"Microsoft Sans Serif", 12,
System::Drawing::FontStyle::Bold, System::Drawing::GraphicsUnit::Point,
            static_cast<System::Byte>(0)));
        this->txt_id2->Location = System::Drawing::Point(3, 3);
        this->txt_id2->Margin = System::Windows::Forms::Padding(3);
        this->txt_id2->Name = L"txt_id2";
        this->txt_id2->Size = System::Drawing::Size(29, 388);
        this->txt_id2->TabIndex = 0;
        this->txt_id2->Text = L"ID";
        this->txt_id2->TextAlign = System::Drawing::ContentAlignment::MiddleCenter;
        //
        // txt_nombre
        //
        this->txt_nombre->AutoSize = true;
        this->txt_nombre->Dock = System::Windows::Forms::DockStyle::Fill;
        this->txt_nombre->Font = (gcnew System::Drawing::Font(L"Microsoft Sans Serif", 12,
System::Drawing::FontStyle::Bold, System::Drawing::GraphicsUnit::Point,
            static_cast<System::Byte>(0)));
        this->txt_nombre->Location = System::Drawing::Point(38, 3);
        this->txt_nombre->Margin = System::Windows::Forms::Padding(3);
        this->txt_nombre->Name = L"txt_nombre";
        this->txt_nombre->Size = System::Drawing::Size(102, 388);
        this->txt_nombre->TabIndex = 1;
        this->txt_nombre->Text = L"Nombre";
        this->txt_nombre->TextAlign = System::Drawing::ContentAlignment::MiddleCenter;
        //
        // txt_nac
        //
        this->txt_nac->AutoSize = true;
        this->txt_nac->Dock = System::Windows::Forms::DockStyle::Fill;
        this->txt_nac->Font = (gcnew System::Drawing::Font(L"Microsoft Sans Serif", 12,
System::Drawing::FontStyle::Bold, System::Drawing::GraphicsUnit::Point,
          static_cast<System::Byte>(0)));
        this->txt_nac->Location = System::Drawing::Point(146, 3);
        this->txt_nac->Margin = System::Windows::Forms::Padding(3);
        this->txt_nac->Name = L"txt_nac";
        this->txt_nac->Size = System::Drawing::Size(112, 388);
        this->txt_nac->TabIndex = 2;
        this->txt_nac->Text = L"Nacimiento";
        this->txt_nac->TextAlign = System::Drawing::ContentAlignment::MiddleCenter;
        //
        // txt_especie
        //
        this->txt_especie->AutoSize = true;
        this->txt_especie->Dock = System::Windows::Forms::DockStyle::Fill;
        this->txt_especie->Font = (gcnew System::Drawing::Font(L"Microsoft Sans Serif", 12,
System::Drawing::FontStyle::Bold, System::Drawing::GraphicsUnit::Point,
          static_cast<System::Byte>(0)));
        this->txt_especie->Location = System::Drawing::Point(264, 3);
        this->txt_especie->Margin = System::Windows::Forms::Padding(3);
        this->txt_especie->Name = L"txt_especie";
        this->txt_especie->Size = System::Drawing::Size(93, 388);
        this->txt_especie->TabIndex = 3;
        this->txt_especie->Text = L"Especie";
        this->txt_especie->TextAlign = System::Drawing::ContentAlignment::MiddleCenter;
        //
        // txt_raza
        //
        this->txt_raza->AutoSize = true;
        this->txt_raza->Dock = System::Windows::Forms::DockStyle::Fill;
        this->txt_raza->Font = (gcnew System::Drawing::Font(L"Microsoft Sans Serif", 12,
System::Drawing::FontStyle::Bold, System::Drawing::GraphicsUnit::Point,
          static_cast<System::Byte>(0)));
        this->txt_raza->Location = System::Drawing::Point(363, 3);
        this->txt_raza->Margin = System::Windows::Forms::Padding(3);
        this->txt_raza->Name = L"txt_raza";
        this->txt_raza->Size = System::Drawing::Size(90, 388);
        this->txt_raza->TabIndex = 4;
        this->txt_raza->Text = L"Raza";
        this->txt_raza->TextAlign = System::Drawing::ContentAlignment::MiddleCenter;
        //
        // txt_foto2
        //
        this->txt_foto2->AutoSize = true;
```



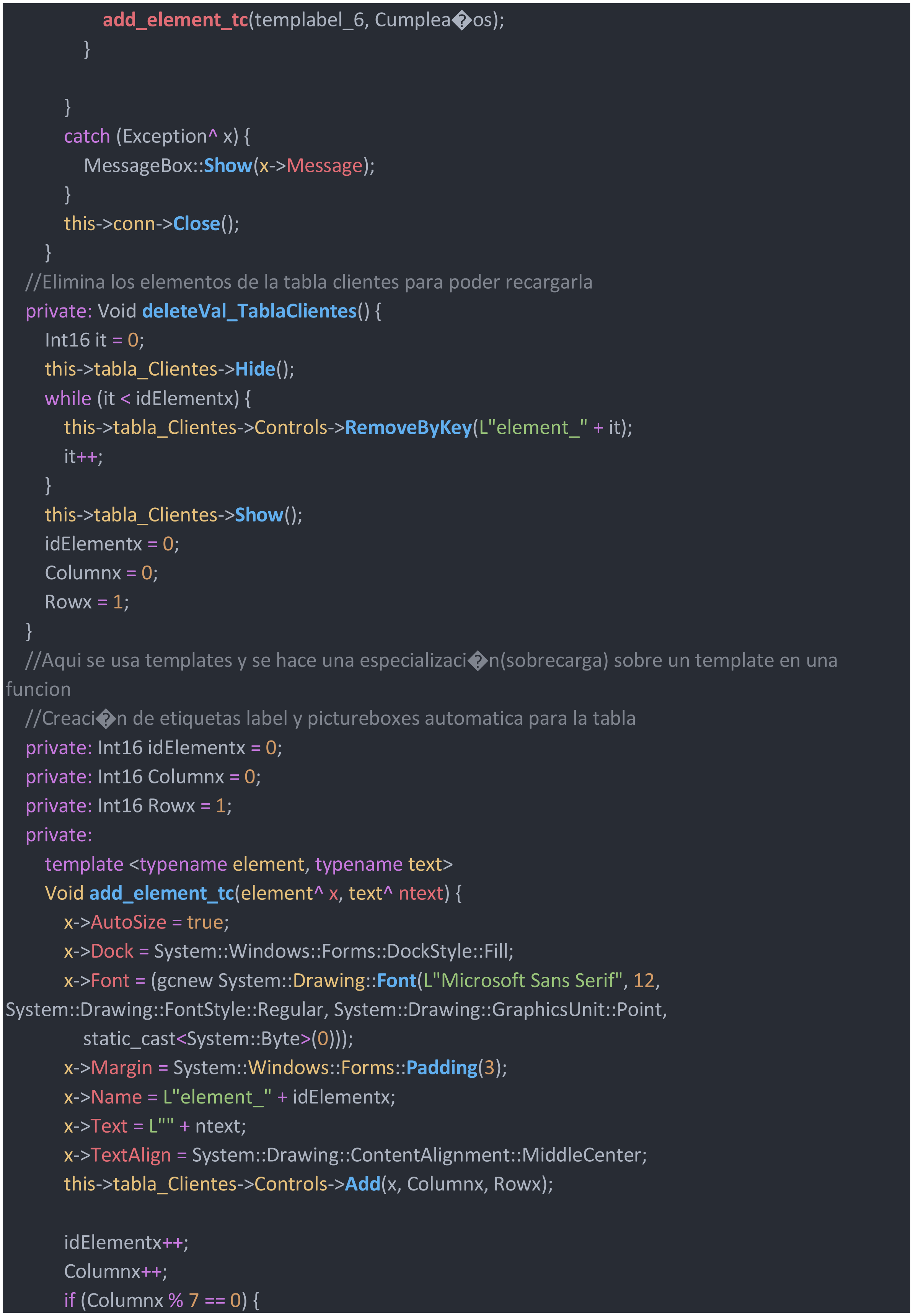
```
        this->txt_foto2->Dock = System::Windows::Forms::DockStyle::Fill;
        this->txt_foto2->Font = (gcnew System::Drawing::Font(L"Microsoft Sans Serif", 12,
System::Drawing::FontStyle::Bold, System::Drawing::GraphicsUnit::Point,
          static_cast<System::Byte>(0)));
        this->txt_foto2->Location = System::Drawing::Point(459, 3);
        this->txt_foto2->Margin = System::Windows::Forms::Padding(3);
        this->txt_foto2->Name = L"txt_foto2";
        this->txt_foto2->Size = System::Drawing::Size(63, 388);
        this->txt_foto2->TabIndex = 5;
        this->txt_foto2->Text = L"Foto";
        this->txt_foto2->TextAlign = System::Drawing::ContentAlignment::MiddleCenter;
        //
        // txt_vp
        //
        this->txt_vp->AutoSize = true;
        this->txt_vp->Dock = System::Windows::Forms::DockStyle::Fill;
        this->txt_vp->Font = (gcnew System::Drawing::Font(L"Microsoft Sans Serif", 12,
System::Drawing::FontStyle::Bold, System::Drawing::GraphicsUnit::Point,
          static_cast<System::Byte>(0)));
        this->txt_vp->Location = System::Drawing::Point(528, 3);
        this->txt_vp->Margin = System::Windows::Forms::Padding(3);
        this->txt_vp->Name = L"txt_vp";
        this->txt_vp->Size = System::Drawing::Size(120, 388);
        this->txt_vp->TabIndex = 6;
        this->txt_vp->Text = L"Vacuna Parvo";
        this->txt_vp->TextAlign = System::Drawing::ContentAlignment::MiddleCenter;
        //
        // txt_vd
        //
        this->txt_vd->AutoSize = true;
        this->txt_vd->Dock = System::Windows::Forms::DockStyle::Fill;
        this->txt_vd->Font = (gcnew System::Drawing::Font(L"Microsoft Sans Serif", 12,
System::Drawing::FontStyle::Bold));
        this->txt_vd->Location = System::Drawing::Point(654, 3);
        this->txt_vd->Margin = System::Windows::Forms::Padding(3);
        this->txt_vd->Name = L"txt_vd";
        this->txt_vd->Size = System::Drawing::Size(157, 388);
        this->txt_vd->TabIndex = 7;
        this->txt_vd->Text = L"Vacuna Distemper";
        this->txt_vd->TextAlign = System::Drawing::ContentAlignment::MiddleCenter;
        //
        // txt_vr
        //
        this->txt_vr->AutoSize = true;
        this->txt_vr->Dock = System::Windows::Forms::DockStyle::Fill;
```

```
	this->txt_vr->Font = (gcnew System::Drawing::Font(L"Microsoft Sans Serif", 12, 
System::Drawing::FontStyle::Bold));
	this->txt_vr->Location = System::Drawing::Point(817, 3);
	this->txt_vr->Margin = System::Windows::Forms::Padding(3);
	this->txt_vr->Name = L"txt_vr";
	this->txt_vr->Size = System::Drawing::Size(122, 388);
	this->txt_vr->TabIndex = 8;
	this->txt_vr->Text = L"Vacuna Rabia";
	this->txt_vr->TextAlign = System::Drawing::ContentAlignment::MiddleCenter;
	// 
	// txt_vh
	// 
	this->txt_vh->AutoSize = true;
	this->txt_vh->Dock = System::Windows::Forms::DockStyle::Fill;
	this->txt_vh->Font = (gcnew System::Drawing::Font(L"Microsoft Sans Serif", 12, 
System::Drawing::FontStyle::Bold));
	this->txt_vh->Location = System::Drawing::Point(945, 3);
	this->txt_vh->Margin = System::Windows::Forms::Padding(3);
	this->txt_vh->Name = L"txt_vh";
	this->txt_vh->Size = System::Drawing::Size(147, 388);
	this->txt_vh->TabIndex = 9;
	this->txt_vh->Text = L"Vacuna Hepatitis";
	this->txt_vh->TextAlign = System::Drawing::ContentAlignment::MiddleCenter;
	// 
	// txt_obs
	// 
	this->txt_obs->AutoSize = true;
	this->txt_obs->Dock = System::Windows::Forms::DockStyle::Fill;
	this->txt_obs->Font = (gcnew System::Drawing::Font(L"Microsoft Sans Serif", 12, 
System::Drawing::FontStyle::Bold));
	this->txt_obs->Location = System::Drawing::Point(1098, 3);
	this->txt_obs->Margin = System::Windows::Forms::Padding(3);
	this->txt_obs->Name = L"txt_obs";
	this->txt_obs->Size = System::Drawing::Size(443, 388);
	this->txt_obs->TabIndex = 10;
	this->txt_obs->Text = L"Observaciones";
	this->txt_obs->TextAlign = System::Drawing::ContentAlignment::MiddleCenter;
	// 
	// txt_idcliente
	// 
	this->txt_idcliente->AutoSize = true;
	this->txt_idcliente->Dock = System::Windows::Forms::DockStyle::Fill;
	this->txt_idcliente->Font = (gcnew System::Drawing::Font(L"Microsoft Sans Serif", 12, 
System::Drawing::FontStyle::Bold));
	this->txt_idcliente->Location = System::Drawing::Point(1547, 3);
```

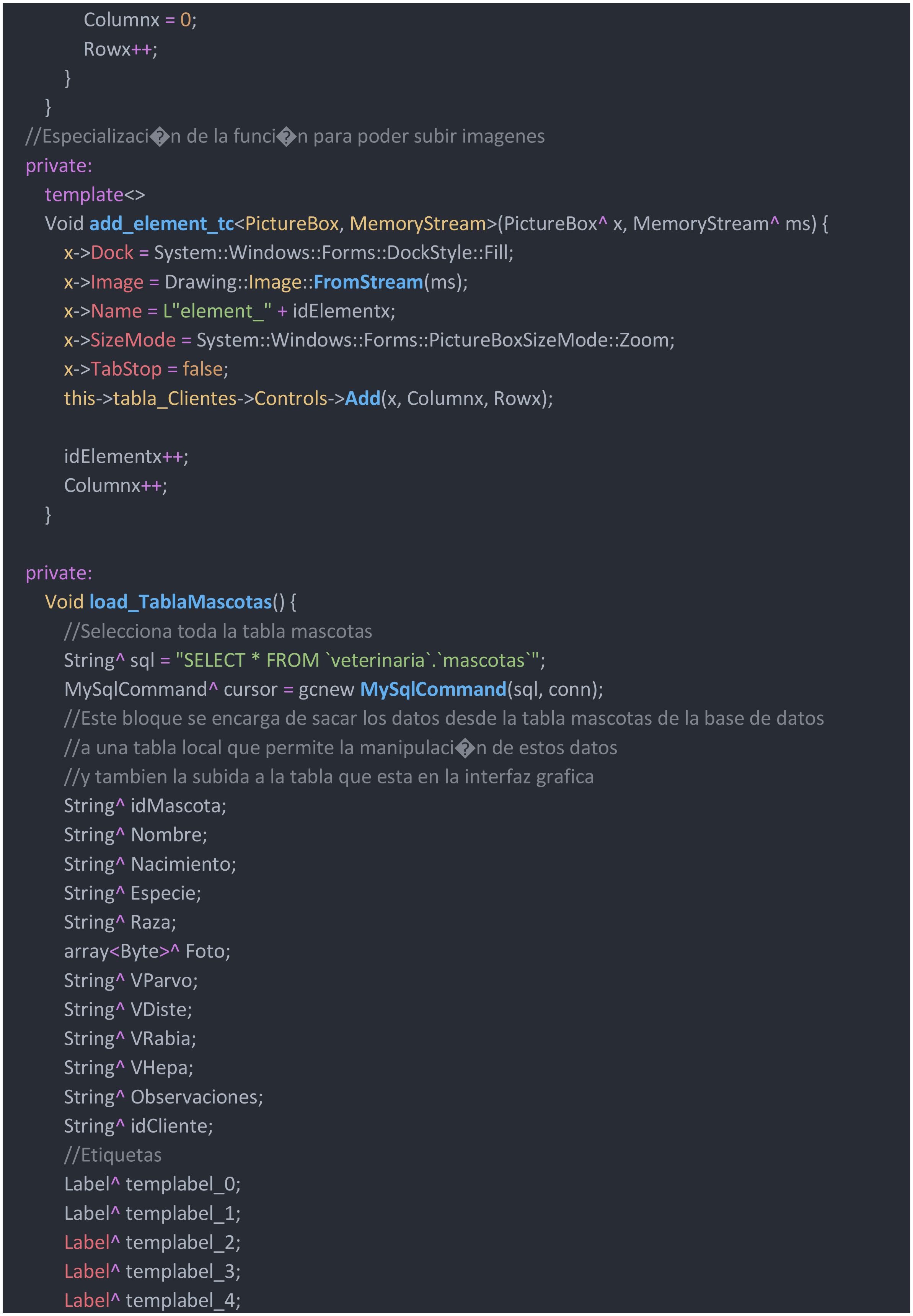
```
      this->txt_idcliente->Margin = System::Windows::Forms::Padding(3);
      this->txt_idcliente->Name = L"txt_idcliente";
      this->txt_idcliente->Size = System::Drawing::Size(89, 388);
      this->txt_idcliente->TabIndex = 11;
      this->txt_idcliente->Text = L"ID Cliente";
      this->txt_idcliente->TextAlign = System::Drawing::ContentAlignment::MiddleCenter;
      //
      // Cliente
      //
      this->AutoScaleDimensions = System::Drawing::SizeF(6, 13);
      this->AutoScaleMode = System::Windows::Forms::AutoScaleMode::Font;
      this->ClientSize = System::Drawing::Size(1250, 550);
      this->Controls->Add(this->mult_cli_mas);
      this->Controls->Add(this->n_masco);
      this->Controls->Add(this->label3);
      this->Controls->Add(this->n_clie);
      this->FormBorderStyle = System::Windows::Forms::FormBorderStyle::None;
      this->Margin = System::Windows::Forms::Padding(2);
      this->Name = L"Cliente";
      this->Text = L"d";
      this->mult_cli_mas->ResumeLayout(false);
      this->tabPage1->ResumeLayout(false);
      this->tabla_Clientes->ResumeLayout(false);
      this->tabla_Clientes->PerformLayout();
      this->tabPage2->ResumeLayout(false);
      this->tabla_Mascotas->ResumeLayout(false);
      this->tabla_Mascotas->PerformLayout();
      this->ResumeLayout(false);

    }
#pragma endregion
  private:
    Void load_TablaClientes() {
      //Selecciona toda la tabla clientes
      String^ sql = "SELECT * FROM `veterinaria`.`clientes`";
      MySqlCommand^ cursor = gcnew MySqlCommand(sql, conn);
      //Este bloque se encarga de sacar los datos desde la tabla clientes de la base de datos
      //a una tabla local que permite la manipulaci�n de estos datos
      //y tambien la subida a la tabla que esta en la interfaz grafica
      String^ idCliente;
      String^ Nombres;
      String^ Apellidos;
      array<Byte>^ Foto;
      String^ Correo;
      String^ Telefono;
```



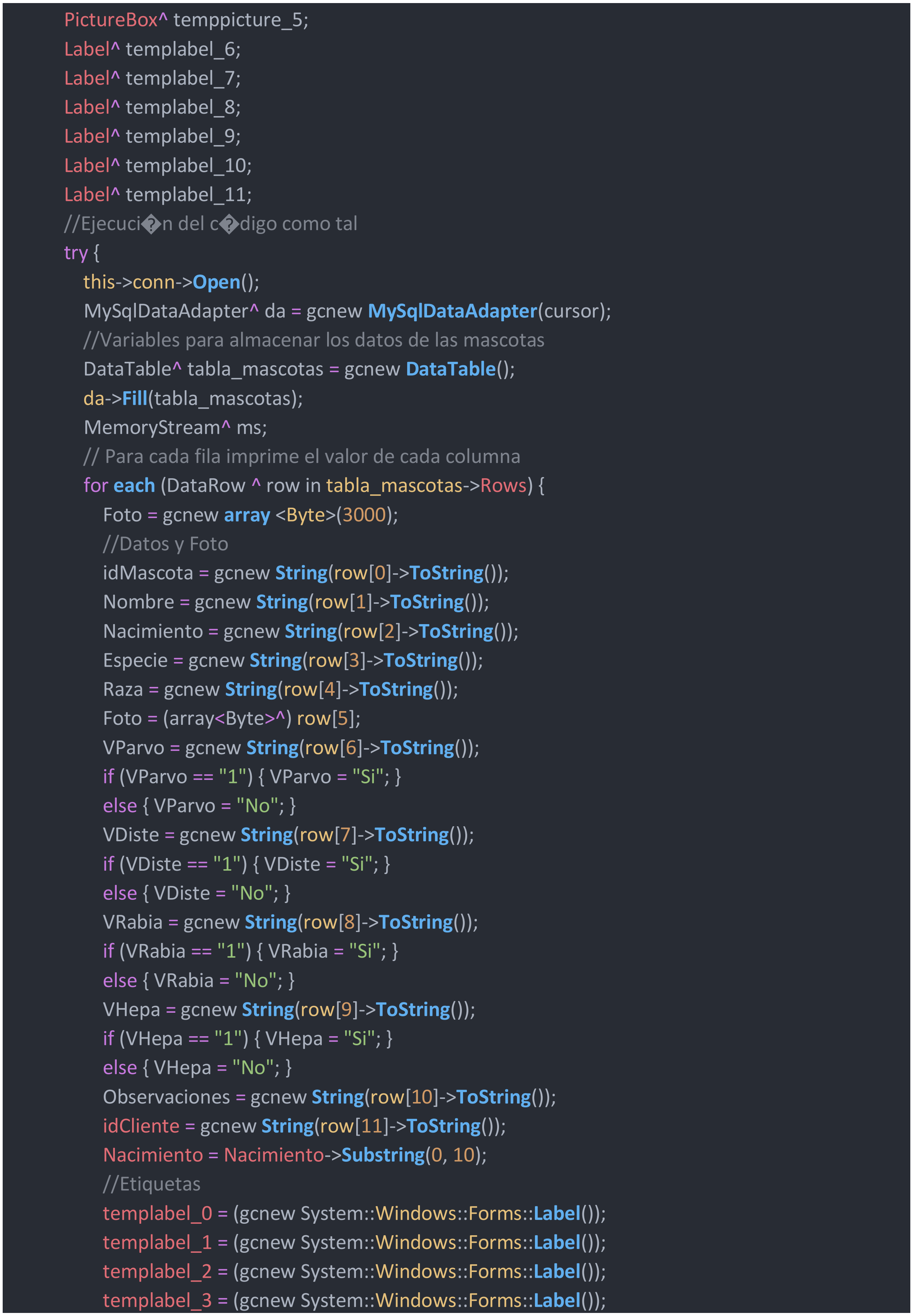
```
        String^ Cumplea�os;
        //Etiquetas
        Label^ templabel_0;
        Label^ templabel_1;
        Label^ templabel_2;
        PictureBox^ temppicture_3;
        Label^ templabel_4;
        Label^ templabel_5;
        Label^ templabel_6;
        //Ejecuci�n del c�digo como tal
        try {
          this->conn->Open();
          MySqlDataAdapter^ da = gcnew MySqlDataAdapter(cursor);
          //Variables para almacenar los datos de los clientes
          DataTable^ tabla_clientes = gcnew DataTable();
          da->Fill(tabla_clientes);
          MemoryStream^ ms;
          // Para cada fila imprime el valor de cada columna
          for each (DataRow ^ row in tabla_clientes->Rows) {
            Foto = gcnew array <Byte>(3000);
            //Datos y Foto
            idCliente = gcnew String(row[0]->ToString());
            Nombres = gcnew String(row[1]->ToString());
            Apellidos = gcnew String(row[2]->ToString());
            Foto = (array<Byte>^) row[3];
            Correo = gcnew String(row[4]->ToString());
            Telefono = gcnew String(row[5]->ToString());
            Cumplea�os = gcnew String(row[6]->ToString());
            Cumplea�os = Cumplea�os->Substring(0, 10);
            //Etiquetas
            templabel_0 = (gcnew System::Windows::Forms::Label());
            templabel_1 = (gcnew System::Windows::Forms::Label());
            templabel_2 = (gcnew System::Windows::Forms::Label());
            temppicture_3 = (gcnew System::Windows::Forms::PictureBox());
            templabel_4 = (gcnew System::Windows::Forms::Label());
            templabel_5 = (gcnew System::Windows::Forms::Label());
            templabel_6 = (gcnew System::Windows::Forms::Label());
            //Subida de los datos a la interfaz grafica
            ms = gcnew MemoryStream(Foto);
            add_element_tc(templabel_0, idCliente);
            add_element_tc(templabel_1, Nombres);
            add_element_tc(templabel_2, Apellidos);
            add_element_tc(temppicture_3, ms);
            add_element_tc(templabel_4, Correo);
            add_element_tc(templabel_5, Telefono);
```



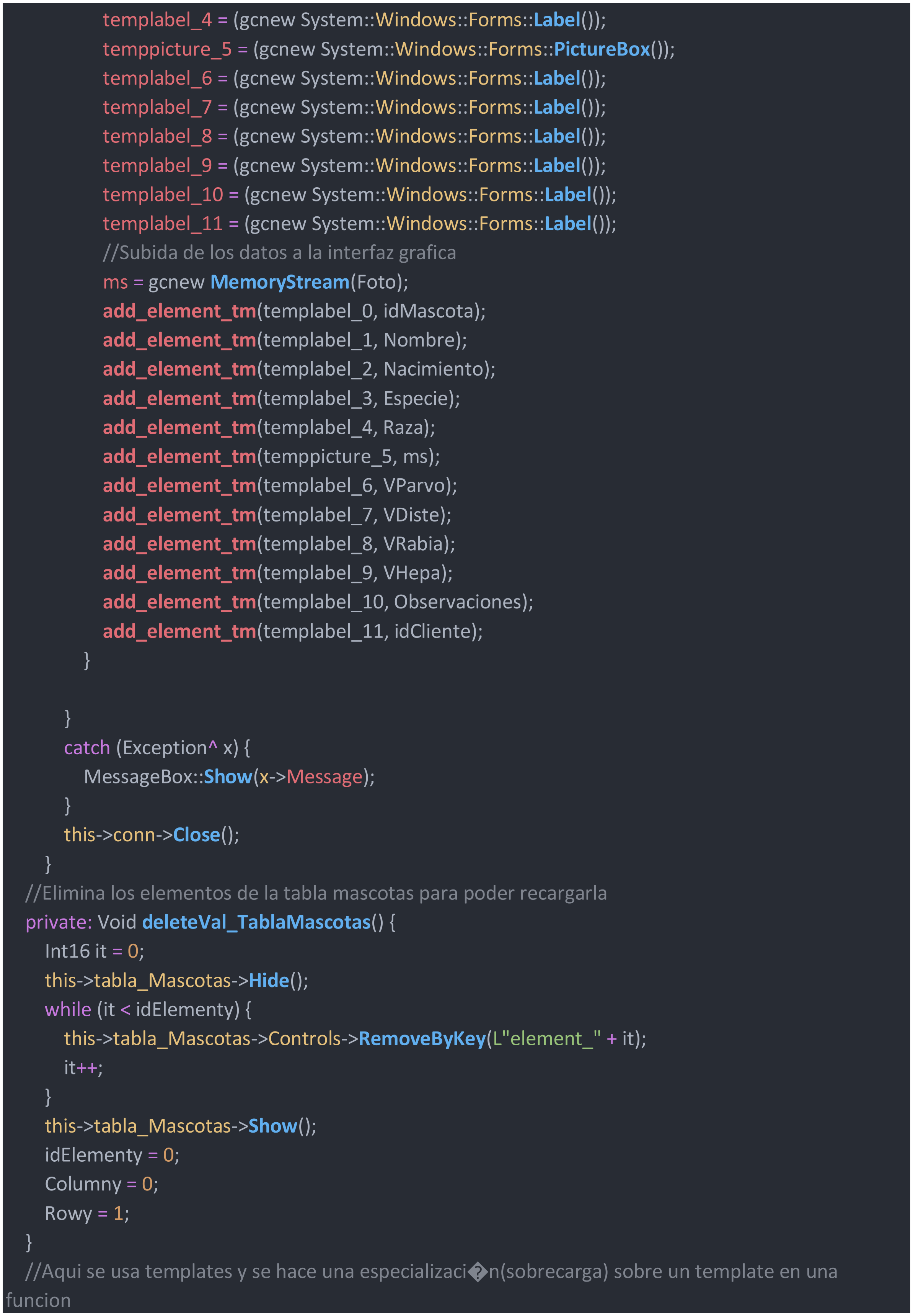
```
                    add_element_tc(templabel_6, Cumplea�os);
                }

            }
            catch (Exception^ x) {
                MessageBox::Show(x->Message);
            }
            this->conn->Close();
        }
    //Elimina los elementos de la tabla clientes para poder recargarla
    private: Void deleteVal_TablaClientes() {
        Int16 it = 0;
        this->tabla_Clientes->Hide();
        while (it < idElementx) {
            this->tabla_Clientes->Controls->RemoveByKey(L"element_" + it);
            it++;
        }
        this->tabla_Clientes->Show();
        idElementx = 0;
        Columnx = 0;
        Rowx = 1;
    }
    //Aqui se usa templates y se hace una especializaci�n(sobrecarga) sobre un template en una
funcion
    //Creaci�n de etiquetas label y pictureboxes automatica para la tabla
    private: Int16 idElementx = 0;
    private: Int16 Columnx = 0;
    private: Int16 Rowx = 1;
    private:
        template <typename element, typename text>
        Void add_element_tc(element^ x, text^ ntext) {
            x->AutoSize = true;
            x->Dock = System::Windows::Forms::DockStyle::Fill;
            x->Font = (gcnew System::Drawing::Font(L"Microsoft Sans Serif", 12,
System::Drawing::FontStyle::Regular, System::Drawing::GraphicsUnit::Point,
                static_cast<System::Byte>(0)));
            x->Margin = System::Windows::Forms::Padding(3);
            x->Name = L"element_" + idElementx;
            x->Text = L"" + ntext;
            x->TextAlign = System::Drawing::ContentAlignment::MiddleCenter;
            this->tabla_Clientes->Controls->Add(x, Columnx, Rowx);

            idElementx++;
            Columnx++;
            if (Columnx % 7 == 0) {
```



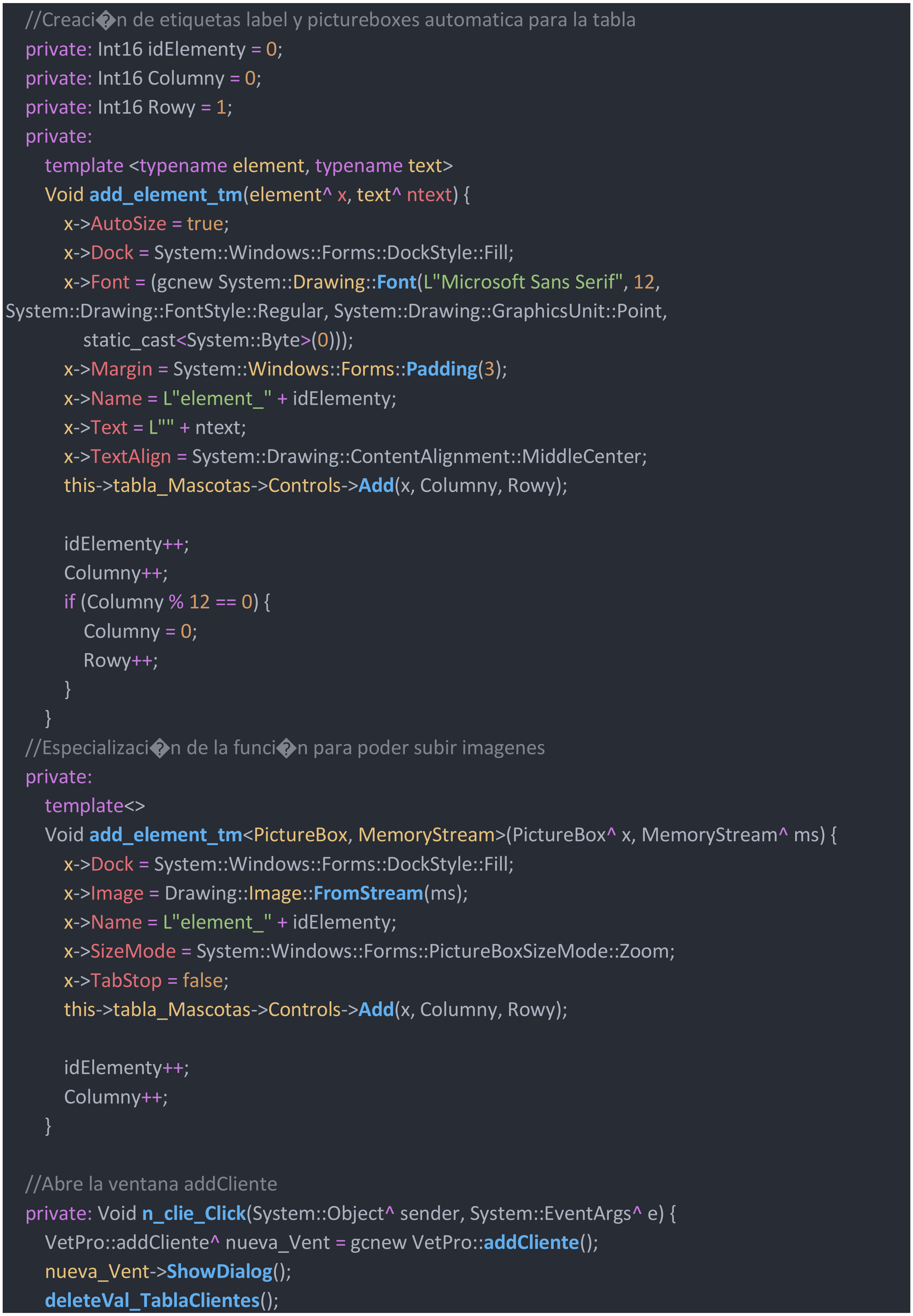
```
            Columnx = 0;
            Rowx++;
        }
    }
//Especializaci�n de la funci�n para poder subir imagenes
private:
    template<>
    Void add_element_tc<PictureBox, MemoryStream>(PictureBox^ x, MemoryStream^ ms) {
        x->Dock = System::Windows::Forms::DockStyle::Fill;
        x->Image = Drawing::Image::FromStream(ms);
        x->Name = L"element_" + idElementx;
        x->SizeMode = System::Windows::Forms::PictureBoxSizeMode::Zoom;
        x->TabStop = false;
        this->tabla_Clientes->Controls->Add(x, Columnx, Rowx);

        idElementx++;
        Columnx++;
    }

private:
    Void load_TablaMascotas() {
        //Selecciona toda la tabla mascotas
        String^ sql = "SELECT * FROM `veterinaria`.`mascotas`";
        MySqlCommand^ cursor = gcnew MySqlCommand(sql, conn);
        //Este bloque se encarga de sacar los datos desde la tabla mascotas de la base de datos
        //a una tabla local que permite la manipulaci�n de estos datos
        //y tambien la subida a la tabla que esta en la interfaz grafica
        String^ idMascota;
        String^ Nombre;
        String^ Nacimiento;
        String^ Especie;
        String^ Raza;
        array<Byte>^ Foto;
        String^ VParvo;
        String^ VDiste;
        String^ VRabia;
        String^ VHepa;
        String^ Observaciones;
        String^ idCliente;
        //Etiquetas
        Label^ templabel_0;
        Label^ templabel_1;
        Label^ templabel_2;
        Label^ templabel_3;
        Label^ templabel_4;
```



```
PictureBox^ temppicture_5;
Label^ templabel_6;
Label^ templabel_7;
Label^ templabel_8;
Label^ templabel_9;
Label^ templabel_10;
Label^ templabel_11;
//Ejecuci�n del c�digo como tal
try {
    this->conn->Open();
    MySqlDataAdapter^ da = gcnew MySqlDataAdapter(cursor);
    //Variables para almacenar los datos de las mascotas
    DataTable^ tabla_mascotas = gcnew DataTable();
    da->Fill(tabla_mascotas);
    MemoryStream^ ms;
    // Para cada fila imprime el valor de cada columna
    for each (DataRow ^ row in tabla_mascotas->Rows) {
        Foto = gcnew array <Byte>(3000);
        //Datos y Foto
        idMascota = gcnew String(row[0]->ToString());
        Nombre = gcnew String(row[1]->ToString());
        Nacimiento = gcnew String(row[2]->ToString());
        Especie = gcnew String(row[3]->ToString());
        Raza = gcnew String(row[4]->ToString());
        Foto = (array<Byte>^) row[5];
        VParvo = gcnew String(row[6]->ToString());
        if (VParvo == "1") { VParvo = "Si"; }
        else { VParvo = "No"; }
        VDiste = gcnew String(row[7]->ToString());
        if (VDiste == "1") { VDiste = "Si"; }
        else { VDiste = "No"; }
        VRabia = gcnew String(row[8]->ToString());
        if (VRabia == "1") { VRabia = "Si"; }
        else { VRabia = "No"; }
        VHepa = gcnew String(row[9]->ToString());
        if (VHepa == "1") { VHepa = "Si"; }
        else { VHepa = "No"; }
        Observaciones = gcnew String(row[10]->ToString());
        idCliente = gcnew String(row[11]->ToString());
        Nacimiento = Nacimiento->Substring(0, 10);
        //Etiquetas
        templabel_0 = (gcnew System::Windows::Forms::Label());
        templabel_1 = (gcnew System::Windows::Forms::Label());
        templabel_2 = (gcnew System::Windows::Forms::Label());
        templabel_3 = (gcnew System::Windows::Forms::Label());
```



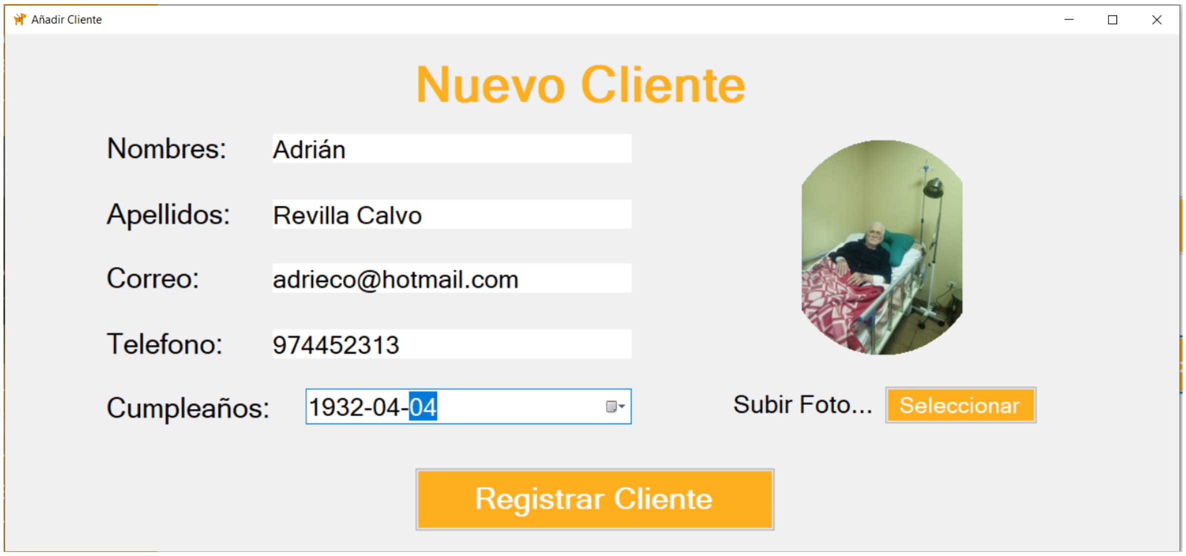
```
          templabel_4 = (gcnew System::Windows::Forms::Label());
          temppicture_5 = (gcnew System::Windows::Forms::PictureBox());
          templabel_6 = (gcnew System::Windows::Forms::Label());
          templabel_7 = (gcnew System::Windows::Forms::Label());
          templabel_8 = (gcnew System::Windows::Forms::Label());
          templabel_9 = (gcnew System::Windows::Forms::Label());
          templabel_10 = (gcnew System::Windows::Forms::Label());
          templabel_11 = (gcnew System::Windows::Forms::Label());
          //Subida de los datos a la interfaz grafica
          ms = gcnew MemoryStream(Foto);
          add_element_tm(templabel_0, idMascota);
          add_element_tm(templabel_1, Nombre);
          add_element_tm(templabel_2, Nacimiento);
          add_element_tm(templabel_3, Especie);
          add_element_tm(templabel_4, Raza);
          add_element_tm(temppicture_5, ms);
          add_element_tm(templabel_6, VParvo);
          add_element_tm(templabel_7, VDiste);
          add_element_tm(templabel_8, VRabia);
          add_element_tm(templabel_9, VHepa);
          add_element_tm(templabel_10, Observaciones);
          add_element_tm(templabel_11, idCliente);
        }

      }
      catch (Exception^ x) {
        MessageBox::Show(x->Message);
      }
      this->conn->Close();
    }
  //Elimina los elementos de la tabla mascotas para poder recargarla
  private: Void deleteVal_TablaMascotas() {
    Int16 it = 0;
    this->tabla_Mascotas->Hide();
    while (it < idElementy) {
      this->tabla_Mascotas->Controls->RemoveByKey(L"element_" + it);
      it++;
    }
    this->tabla_Mascotas->Show();
    idElementy = 0;
    Columny = 0;
    Rowy = 1;
  }
  //Aqui se usa templates y se hace una especializaci�n(sobrecarga) sobre un template en una
funcion
```



```
//Creaci�n de etiquetas label y pictureboxes automatica para la tabla
private: Int16 idElementy = 0;
private: Int16 Columny = 0;
private: Int16 Rowy = 1;
private:
    template <typename element, typename text>
    Void add_element_tm(element^ x, text^ ntext) {
        x->AutoSize = true;
        x->Dock = System::Windows::Forms::DockStyle::Fill;
        x->Font = (gcnew System::Drawing::Font(L"Microsoft Sans Serif", 12,
System::Drawing::FontStyle::Regular, System::Drawing::GraphicsUnit::Point,
            static_cast<System::Byte>(0)));
        x->Margin = System::Windows::Forms::Padding(3);
        x->Name = L"element_" + idElementy;
        x->Text = L"" + ntext;
        x->TextAlign = System::Drawing::ContentAlignment::MiddleCenter;
        this->tabla_Mascotas->Controls->Add(x, Columny, Rowy);

        idElementy++;
        Columny++;
        if (Columny % 12 == 0) {
            Columny = 0;
            Rowy++;
        }
    }
//Especializaci�n de la funci�n para poder subir imagenes
private:
    template<>
    Void add_element_tm<PictureBox, MemoryStream>(PictureBox^ x, MemoryStream^ ms) {
        x->Dock = System::Windows::Forms::DockStyle::Fill;
        x->Image = Drawing::Image::FromStream(ms);
        x->Name = L"element_" + idElementy;
        x->SizeMode = System::Windows::Forms::PictureBoxSizeMode::Zoom;
        x->TabStop = false;
        this->tabla_Mascotas->Controls->Add(x, Columny, Rowy);

        idElementy++;
        Columny++;
    }

//Abre la ventana addCliente
private: Void n_clie_Click(System::Object^ sender, System::EventArgs^ e) {
    VetPro::addCliente^ nueva_Vent = gcnew VetPro::addCliente();
    nueva_Vent->ShowDialog();
    deleteVal_TablaClientes();
```

```
        load_TablaClientes();
    }
    //Abre la ventana addMascota
    private: Void n_masco_Click(System::Object^ sender, System::EventArgs^ e) {
        VetPro::addMascota^ nueva_Vent = gcnew VetPro::addMascota();
        nueva_Vent->ShowDialog();
        deleteVal_TablaMascotas();
        load_TablaMascotas();
    }
};
}
```

Add_Cliente



Nuevo Cliente

Nombres: Adrián

Apellidos: Revilla Calvo

Correo: adrieco@hotmail.com

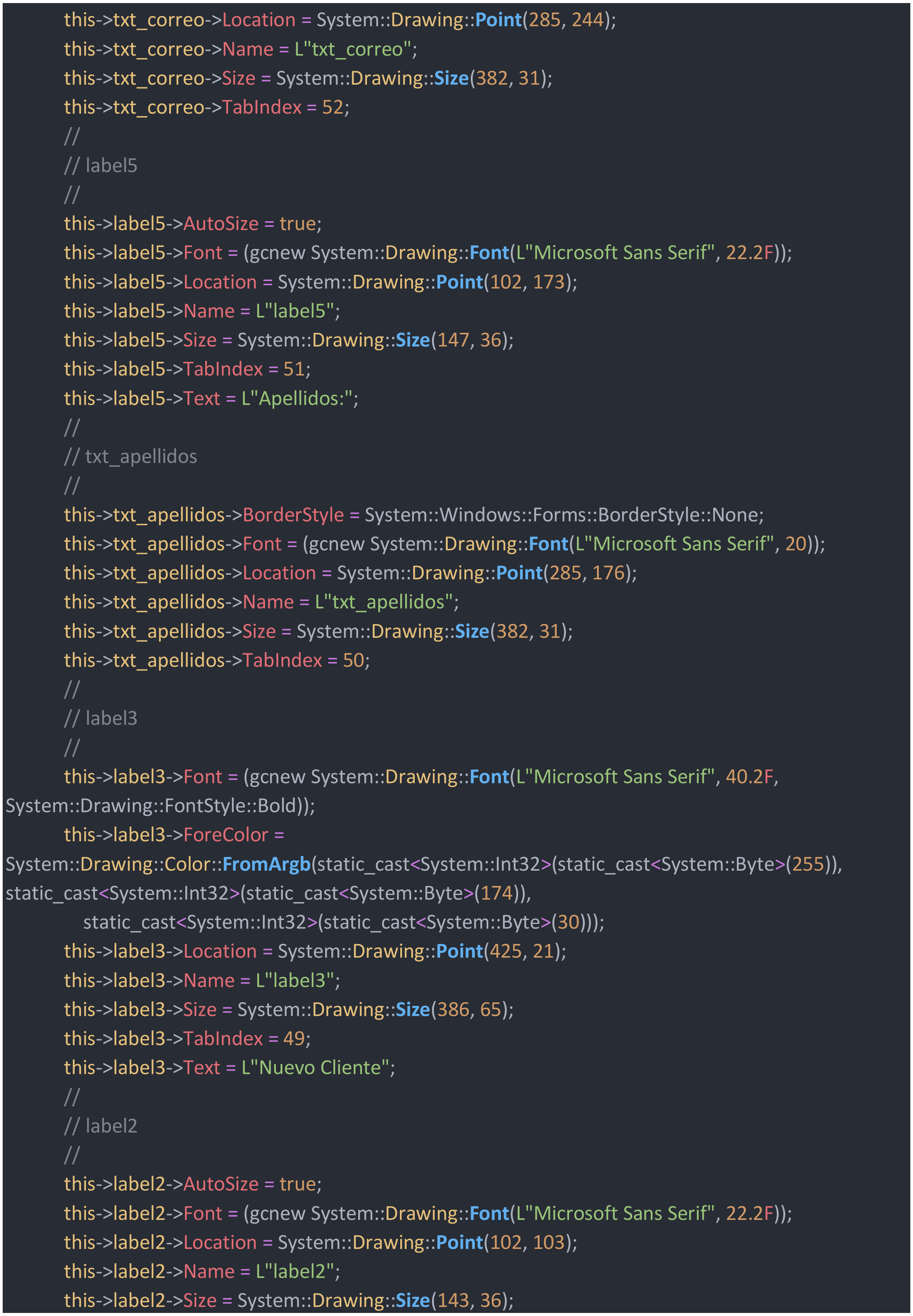
Telefono: 974452313

Cumpleaños: 1932-04-04

Similar a la ventana de registro de Veterinarios, este form permite subir información a la base datos, manejando también información como cumpleaños, teléfonos o correos

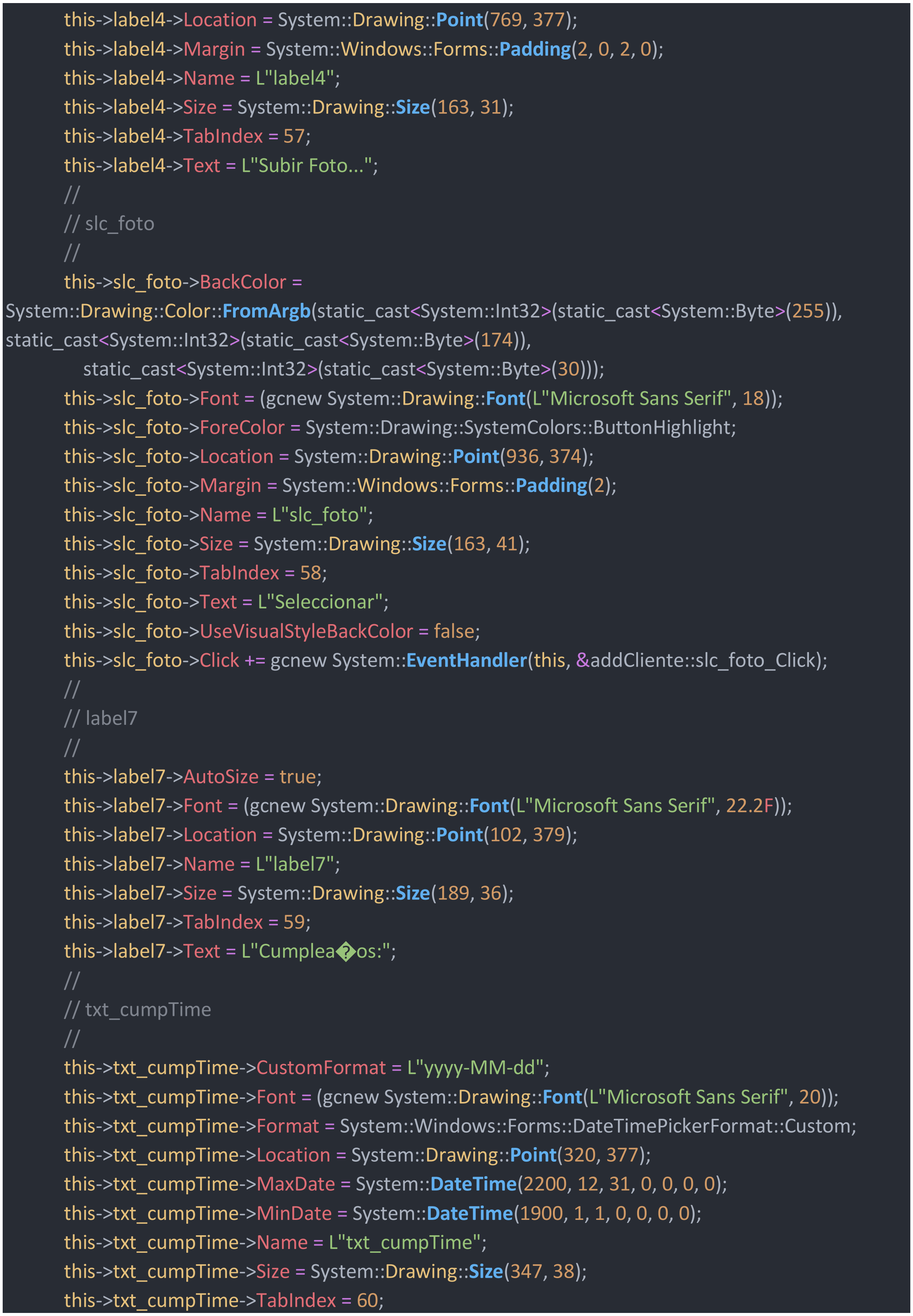
```
#pragma once
#include "CircularDisplay_forPictureBox.h"

namespace VetPro {

  using namespace System;
  using namespace System::ComponentModel;
  using namespace System::Collections;
  using namespace System::Windows::Forms;
  using namespace System::Data;
  using namespace System::Drawing;
  using namespace System::Drawing::Drawing2D;
  using namespace System::Text;
  using namespace System::IO;;
  using namespace System::Threading;
  using namespace System::Threading::Tasks;
  using namespace MySql::Data::MySqlClient;

  public ref class addCliente : public System::Windows::Forms::Form {
  public:
    addCliente(void) {
      InitializeComponent();

      //Cambia la forma de la imagen del cliente de un cuadrado a un elipse
      CircularDisplay_forPictureBox cd;
      cd(this->client_Picture);
    }

  protected:
    ~addCliente() {
      if (components) {
        delete components;
      }
    }

  private: String^ connectionString = "datasource=localhost; username=root; password=1234;
database=veterinaria";
  private: MySqlConnection^ conn = gcnew MySqlConnection(connectionString);
  private: System::Windows::Forms::Label^ label6;
  private: System::Windows::Forms::TextBox^ txt_telefono;
  private: System::Windows::Forms::Label^ label1;
  private: System::Windows::Forms::TextBox^ txt_correo;
  private: System::Windows::Forms::Label^ label5;
  private: System::Windows::Forms::TextBox^ txt_apellidos;
  private: System::Windows::Forms::Label^ label3;
  private: System::Windows::Forms::Label^ label2;
  private: System::Windows::Forms::TextBox^ txt_nombres;
  private: System::Windows::Forms::PictureBox^ client_Picture;
  private: System::Windows::Forms::Button^ button2;
  private: System::Windows::Forms::Label^ label4;
  private: System::Windows::Forms::Button^ slc_foto;
  private: System::Windows::Forms::Label^ label7;
  private: System::Windows::Forms::DateTimePicker^ txt_cumpTime;

  private: System::ComponentModel::Container ^components;

#pragma region Windows Form Designer generated code
    /// <summary>
    /// M�todo necesario para admitir el Dise�ador. No se puede modificar
    /// el contenido de este m�todo con el editor de c�digo.
    /// </summary>
    void InitializeComponent(void) {
      System::ComponentModel::ComponentResourceManager^ resources = (gcnew
System::ComponentModel::ComponentResourceManager(addCliente::typeid));
      this->label6 = (gcnew System::Windows::Forms::Label());
      this->txt_telefono = (gcnew System::Windows::Forms::TextBox());
      this->label1 = (gcnew System::Windows::Forms::Label());
      this->txt_correo = (gcnew System::Windows::Forms::TextBox());
      this->label5 = (gcnew System::Windows::Forms::Label());
      this->txt_apellidos = (gcnew System::Windows::Forms::TextBox());
      this->label3 = (gcnew System::Windows::Forms::Label());
```

```
	this->label2 = (gcnew System::Windows::Forms::Label());
	this->txt_nombres = (gcnew System::Windows::Forms::TextBox());
	this->client_Picture = (gcnew System::Windows::Forms::PictureBox());
	this->button2 = (gcnew System::Windows::Forms::Button());
	this->label4 = (gcnew System::Windows::Forms::Label());
	this->slc_foto = (gcnew System::Windows::Forms::Button());
	this->label7 = (gcnew System::Windows::Forms::Label());
	this->txt_cumpTime = (gcnew System::Windows::Forms::DateTimePicker());
	(cli::safe_cast<System::ComponentModel::ISupportInitialize^>(this->client_Picture))-
>BeginInit();
	this->SuspendLayout();
	//
	// label6
	//
	this->label6->AutoSize = true;
	this->label6->Font = (gcnew System::Drawing::Font(L"Microsoft Sans Serif", 22.2F));
	this->label6->Location = System::Drawing::Point(102, 311);
	this->label6->Name = L"label6";
	this->label6->Size = System::Drawing::Size(139, 36);
	this->label6->TabIndex = 55;
	this->label6->Text = L"Telefono:";
	//
	// txt_telefono
	//
	this->txt_telefono->BorderStyle = System::Windows::Forms::BorderStyle::None;
	this->txt_telefono->Font = (gcnew System::Drawing::Font(L"Microsoft Sans Serif", 20));
	this->txt_telefono->Location = System::Drawing::Point(285, 314);
	this->txt_telefono->Name = L"txt_telefono";
	this->txt_telefono->Size = System::Drawing::Size(382, 31);
	this->txt_telefono->TabIndex = 54;
	//
	// label1
	//
	this->label1->AutoSize = true;
	this->label1->Font = (gcnew System::Drawing::Font(L"Microsoft Sans Serif", 22.2F));
	this->label1->Location = System::Drawing::Point(102, 241);
	this->label1->Name = L"label1";
	this->label1->Size = System::Drawing::Size(114, 36);
	this->label1->TabIndex = 53;
	this->label1->Text = L"Correo:";
	//
	// txt_correo
	//
	this->txt_correo->BorderStyle = System::Windows::Forms::BorderStyle::None;
	this->txt_correo->Font = (gcnew System::Drawing::Font(L"Microsoft Sans Serif", 20));
```



```
			this->txt_correo->Location = System::Drawing::Point(285, 244);
			this->txt_correo->Name = L"txt_correo";
			this->txt_correo->Size = System::Drawing::Size(382, 31);
			this->txt_correo->TabIndex = 52;
			// 
			// label5
			// 
			this->label5->AutoSize = true;
			this->label5->Font = (gcnew System::Drawing::Font(L"Microsoft Sans Serif", 22.2F));
			this->label5->Location = System::Drawing::Point(102, 173);
			this->label5->Name = L"label5";
			this->label5->Size = System::Drawing::Size(147, 36);
			this->label5->TabIndex = 51;
			this->label5->Text = L"Apellidos:";
			// 
			// txt_apellidos
			// 
			this->txt_apellidos->BorderStyle = System::Windows::Forms::BorderStyle::None;
			this->txt_apellidos->Font = (gcnew System::Drawing::Font(L"Microsoft Sans Serif", 20));
			this->txt_apellidos->Location = System::Drawing::Point(285, 176);
			this->txt_apellidos->Name = L"txt_apellidos";
			this->txt_apellidos->Size = System::Drawing::Size(382, 31);
			this->txt_apellidos->TabIndex = 50;
			// 
			// label3
			// 
			this->label3->Font = (gcnew System::Drawing::Font(L"Microsoft Sans Serif", 40.2F, 
System::Drawing::FontStyle::Bold));
			this->label3->ForeColor = 
System::Drawing::Color::FromArgb(static_cast<System::Int32>(static_cast<System::Byte>(255)), 
static_cast<System::Int32>(static_cast<System::Byte>(174)),
				static_cast<System::Int32>(static_cast<System::Byte>(30)));
			this->label3->Location = System::Drawing::Point(425, 21);
			this->label3->Name = L"label3";
			this->label3->Size = System::Drawing::Size(386, 65);
			this->label3->TabIndex = 49;
			this->label3->Text = L"Nuevo Cliente";
			// 
			// label2
			// 
			this->label2->AutoSize = true;
			this->label2->Font = (gcnew System::Drawing::Font(L"Microsoft Sans Serif", 22.2F));
			this->label2->Location = System::Drawing::Point(102, 103);
			this->label2->Name = L"label2";
			this->label2->Size = System::Drawing::Size(143, 36);
```

```
	this->label2->TabIndex = 48;
	this->label2->Text = L"Nombres:";
	//
	// txt_nombres
	//
	this->txt_nombres->BorderStyle = System::Windows::Forms::BorderStyle::None;
	this->txt_nombres->Font = (gcnew System::Drawing::Font(L"Microsoft Sans Serif", 20));
	this->txt_nombres->Location = System::Drawing::Point(285, 106);
	this->txt_nombres->Name = L"txt_nombres";
	this->txt_nombres->Size = System::Drawing::Size(382, 31);
	this->txt_nombres->TabIndex = 46;
	//
	// client_Picture
	//
	this->client_Picture->Image = (cli::safe_cast<System::Drawing::Image^>(resources-
>GetObject(L"client_Picture.Image")));
	this->client_Picture->Location = System::Drawing::Point(828, 112);
	this->client_Picture->Name = L"client_Picture";
	this->client_Picture->Size = System::Drawing::Size(212, 229);
	this->client_Picture->SizeMode = System::Windows::Forms::PictureBoxSizeMode::Zoom;
	this->client_Picture->TabIndex = 45;
	this->client_Picture->TabStop = false;
	//
	// button2
	//
	this->button2->BackColor =
System::Drawing::Color::FromArgb(static_cast<System::Int32>(static_cast<System::Byte>(255)),
static_cast<System::Int32>(static_cast<System::Byte>(174)),
	  static_cast<System::Int32>(static_cast<System::Byte>(30)));
	this->button2->Font = (gcnew System::Drawing::Font(L"Microsoft Sans Serif", 24,
System::Drawing::FontStyle::Bold, System::Drawing::GraphicsUnit::Point,
	  static_cast<System::Byte>(129)));
	this->button2->ForeColor = System::Drawing::SystemColors::ButtonHighlight;
	this->button2->Location = System::Drawing::Point(436, 461);
	this->button2->Name = L"button2";
	this->button2->Size = System::Drawing::Size(384, 68);
	this->button2->TabIndex = 56;
	this->button2->Text = L"Registrar Cliente";
	this->button2->UseVisualStyleBackColor = false;
	this->button2->Click += gcnew System::EventHandler(this, &addCliente::button2_Click);
	//
	// label4
	//
	this->label4->AutoSize = true;
	this->label4->Font = (gcnew System::Drawing::Font(L"Microsoft Sans Serif", 20));
```

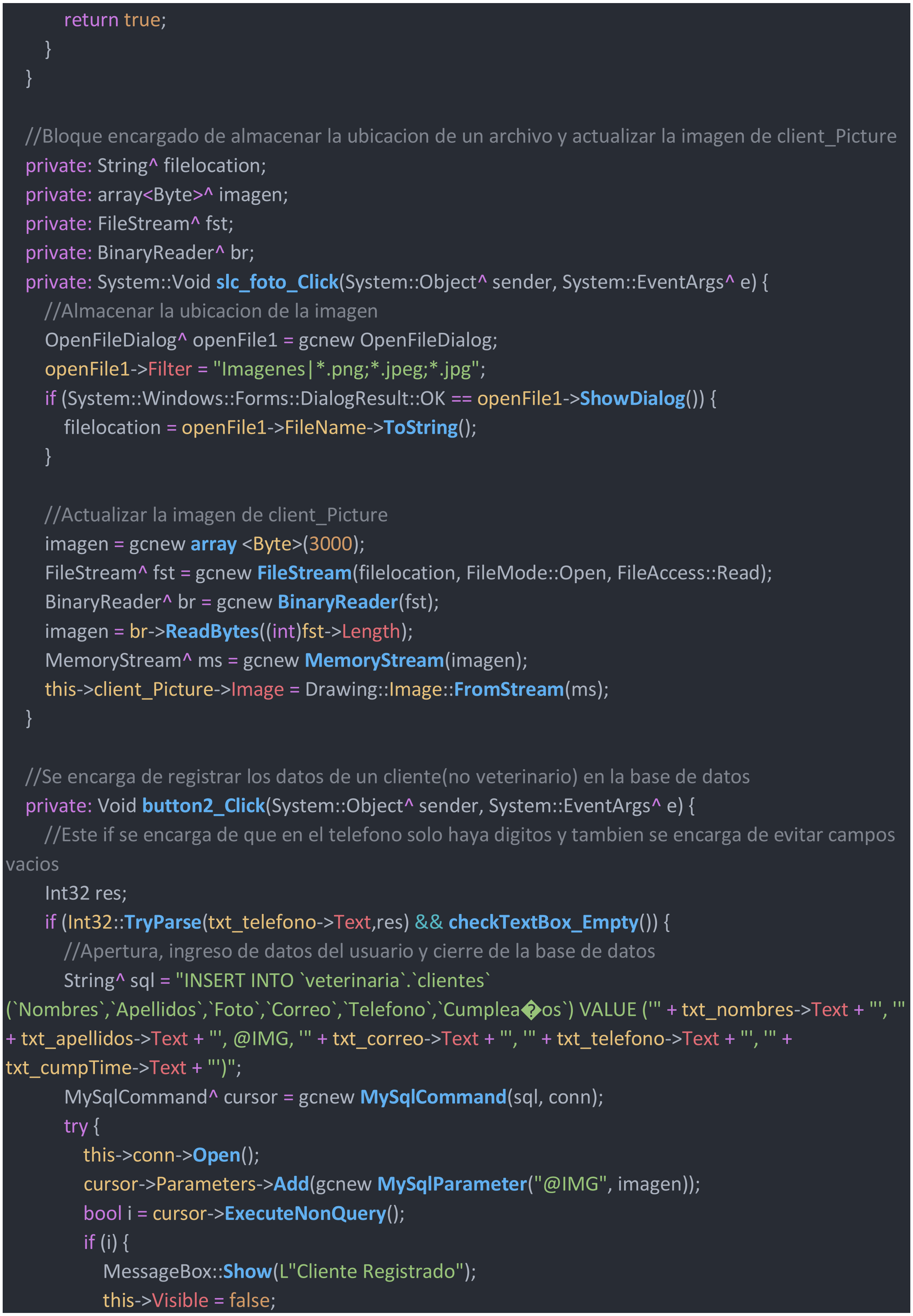


```
    this->label4->Location = System::Drawing::Point(769, 377);
    this->label4->Margin = System::Windows::Forms::Padding(2, 0, 2, 0);
    this->label4->Name = L"label4";
    this->label4->Size = System::Drawing::Size(163, 31);
    this->label4->TabIndex = 57;
    this->label4->Text = L"Subir Foto...";
    //
    // slc_foto
    //
    this->slc_foto->BackColor =
System::Drawing::Color::FromArgb(static_cast<System::Int32>(static_cast<System::Byte>(255)),
static_cast<System::Int32>(static_cast<System::Byte>(174)),
      static_cast<System::Int32>(static_cast<System::Byte>(30)));
    this->slc_foto->Font = (gcnew System::Drawing::Font(L"Microsoft Sans Serif", 18));
    this->slc_foto->ForeColor = System::Drawing::SystemColors::ButtonHighlight;
    this->slc_foto->Location = System::Drawing::Point(936, 374);
    this->slc_foto->Margin = System::Windows::Forms::Padding(2);
    this->slc_foto->Name = L"slc_foto";
    this->slc_foto->Size = System::Drawing::Size(163, 41);
    this->slc_foto->TabIndex = 58;
    this->slc_foto->Text = L"Seleccionar";
    this->slc_foto->UseVisualStyleBackColor = false;
    this->slc_foto->Click += gcnew System::EventHandler(this, &addCliente::slc_foto_Click);
    //
    // label7
    //
    this->label7->AutoSize = true;
    this->label7->Font = (gcnew System::Drawing::Font(L"Microsoft Sans Serif", 22.2F));
    this->label7->Location = System::Drawing::Point(102, 379);
    this->label7->Name = L"label7";
    this->label7->Size = System::Drawing::Size(189, 36);
    this->label7->TabIndex = 59;
    this->label7->Text = L"Cumplea�os:";
    //
    // txt_cumpTime
    //
    this->txt_cumpTime->CustomFormat = L"yyyy-MM-dd";
    this->txt_cumpTime->Font = (gcnew System::Drawing::Font(L"Microsoft Sans Serif", 20));
    this->txt_cumpTime->Format = System::Windows::Forms::DateTimePickerFormat::Custom;
    this->txt_cumpTime->Location = System::Drawing::Point(320, 377);
    this->txt_cumpTime->MaxDate = System::DateTime(2200, 12, 31, 0, 0, 0, 0);
    this->txt_cumpTime->MinDate = System::DateTime(1900, 1, 1, 0, 0, 0, 0);
    this->txt_cumpTime->Name = L"txt_cumpTime";
    this->txt_cumpTime->Size = System::Drawing::Size(347, 38);
    this->txt_cumpTime->TabIndex = 60;
```

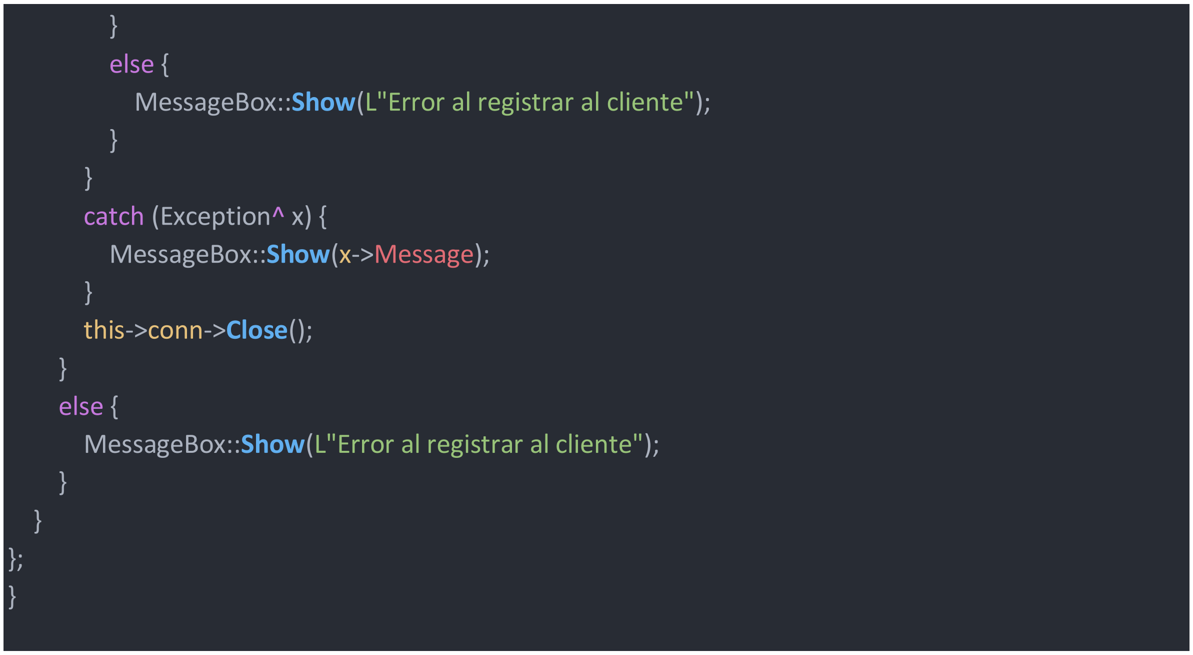
```
			this->txt_cumpTime->Value = System::DateTime(2022, 8, 16, 0, 0, 0, 0);
			//
			// addCliente
			//
			this->AutoScaleDimensions = System::Drawing::SizeF(6, 13);
			this->AutoScaleMode = System::Windows::Forms::AutoScaleMode::Font;
			this->ClientSize = System::Drawing::Size(1250, 550);
			this->Controls->Add(this->txt_cumpTime);
			this->Controls->Add(this->label7);
			this->Controls->Add(this->slc_foto);
			this->Controls->Add(this->label4);
			this->Controls->Add(this->button2);
			this->Controls->Add(this->label6);
			this->Controls->Add(this->txt_telefono);
			this->Controls->Add(this->label1);
			this->Controls->Add(this->txt_correo);
			this->Controls->Add(this->label5);
			this->Controls->Add(this->txt_apellidos);
			this->Controls->Add(this->label3);
			this->Controls->Add(this->label2);
			this->Controls->Add(this->txt_nombres);
			this->Controls->Add(this->client_Picture);
			this->FormBorderStyle = System::Windows::Forms::FormBorderStyle::FixedSingle;
			this->Icon = (cli::safe_cast<System::Drawing::Icon^>(resources->GetObject(L"$this.Icon")));
			this->Margin = System::Windows::Forms::Padding(2);
			this->Name = L"addCliente";
			this->Text = L"A�adir Cliente";
			(cli::safe_cast<System::ComponentModel::ISupportInitialize^>(this->client_Picture))->EndInit();
			this->ResumeLayout(false);
			this->PerformLayout();

		}

#pragma endregion
```

Esta función comprueba que ninguno de los text boz del form se encuentren vacíos, el valor de retorno es un booleano

```
	private: Boolean checkTextBox_Empty() {
		if (String::IsNullOrEmpty(txt_nombres->Text) || System::String::IsNullOrEmpty(txt_apellidos->Text)
			|| String::IsNullOrEmpty(txt_correo->Text) || System::String::IsNullOrEmpty(txt_telefono->Text)
			|| String::IsNullOrEmpty(txt_cumpTime->Text)) {
			return false;
		}
		else {
```

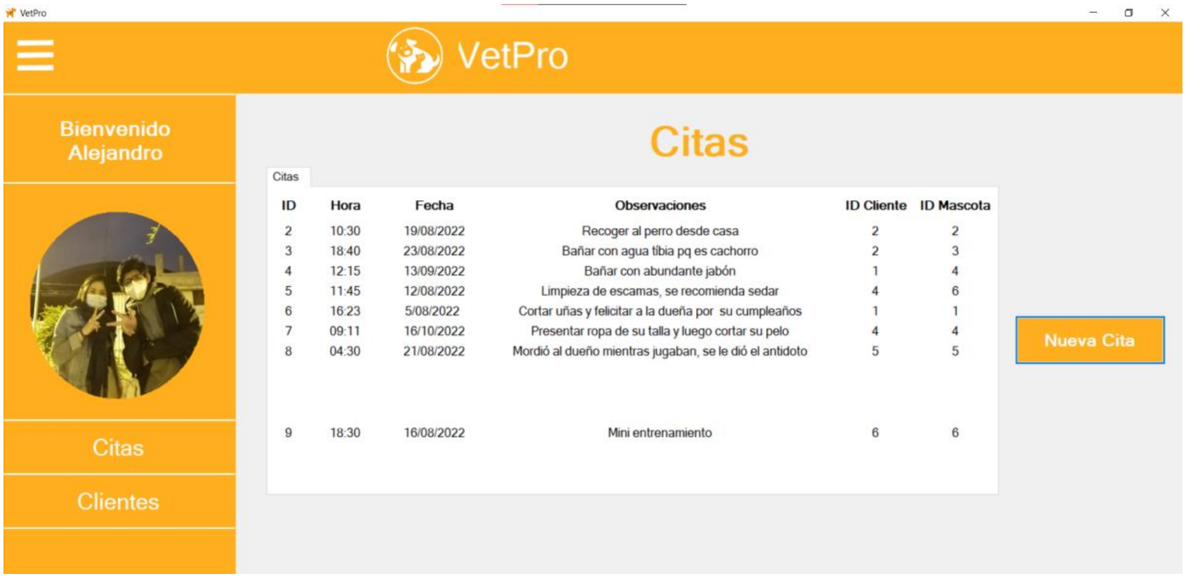


```
            return true;
        }
    }

    //Bloque encargado de almacenar la ubicacion de un archivo y actualizar la imagen de client_Picture
    private: String^ filelocation;
    private: array<Byte>^ imagen;
    private: FileStream^ fst;
    private: BinaryReader^ br;
    private: System::Void slc_foto_Click(System::Object^ sender, System::EventArgs^ e) {
        //Almacenar la ubicacion de la imagen
        OpenFileDialog^ openFile1 = gcnew OpenFileDialog;
        openFile1->Filter = "Imagenes|*.png;*.jpeg;*.jpg";
        if (System::Windows::Forms::DialogResult::OK == openFile1->ShowDialog()) {
            filelocation = openFile1->FileName->ToString();
        }

        //Actualizar la imagen de client_Picture
        imagen = gcnew array <Byte>(3000);
        FileStream^ fst = gcnew FileStream(filelocation, FileMode::Open, FileAccess::Read);
        BinaryReader^ br = gcnew BinaryReader(fst);
        imagen = br->ReadBytes((int)fst->Length);
        MemoryStream^ ms = gcnew MemoryStream(imagen);
        this->client_Picture->Image = Drawing::Image::FromStream(ms);
    }

    //Se encarga de registrar los datos de un cliente(no veterinario) en la base de datos
    private: Void button2_Click(System::Object^ sender, System::EventArgs^ e) {
        //Este if se encarga de que en el telefono solo haya digitos y tambien se encarga de evitar campos
vacios
        Int32 res;
        if (Int32::TryParse(txt_telefono->Text,res) && checkTextBox_Empty()) {
            //Apertura, ingreso de datos del usuario y cierre de la base de datos
            String^ sql = "INSERT INTO `veterinaria`.`clientes`
(`Nombres`,`Apellidos`,`Foto`,`Correo`,`Telefono`,`Cumplea�os`) VALUE ('" + txt_nombres->Text + "', '"
+ txt_apellidos->Text + "', @IMG, '" + txt_correo->Text + "', '" + txt_telefono->Text + "', '" +
txt_cumpTime->Text + "')";
            MySqlCommand^ cursor = gcnew MySqlCommand(sql, conn);
            try {
                this->conn->Open();
                cursor->Parameters->Add(gcnew MySqlParameter("@IMG", imagen));
                bool i = cursor->ExecuteNonQuery();
                if (i) {
                    MessageBox::Show(L"Cliente Registrado");
                    this->Visible = false;
```



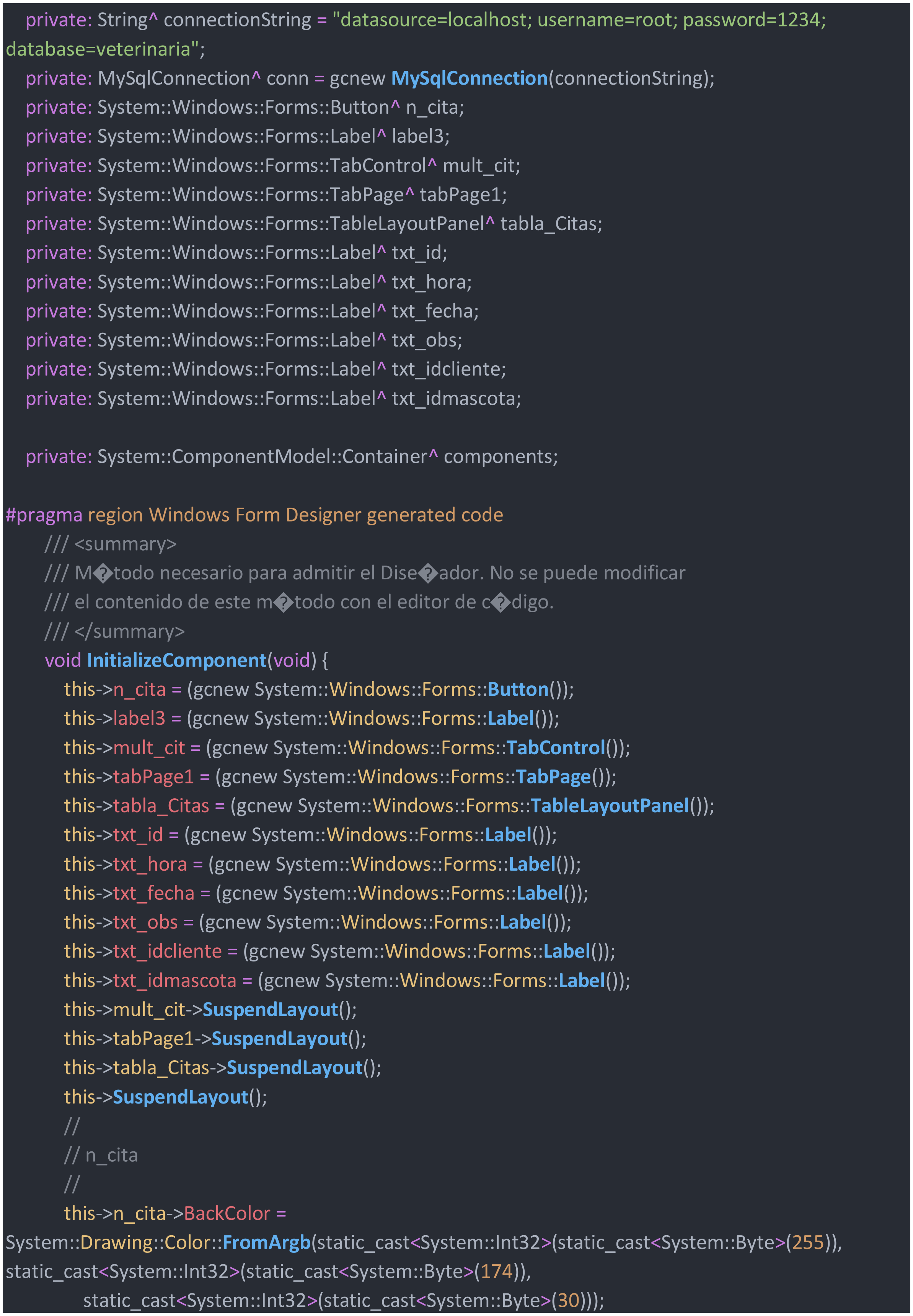
```
                }
                else {
                    MessageBox::Show(L"Error al registrar al cliente");
                }
            }
            catch (Exception^ x) {
                MessageBox::Show(x->Message);
            }
            this->conn->Close();
        }
        else {
            MessageBox::Show(L"Error al registrar al cliente");
        }
    }
};
}
```

Citas



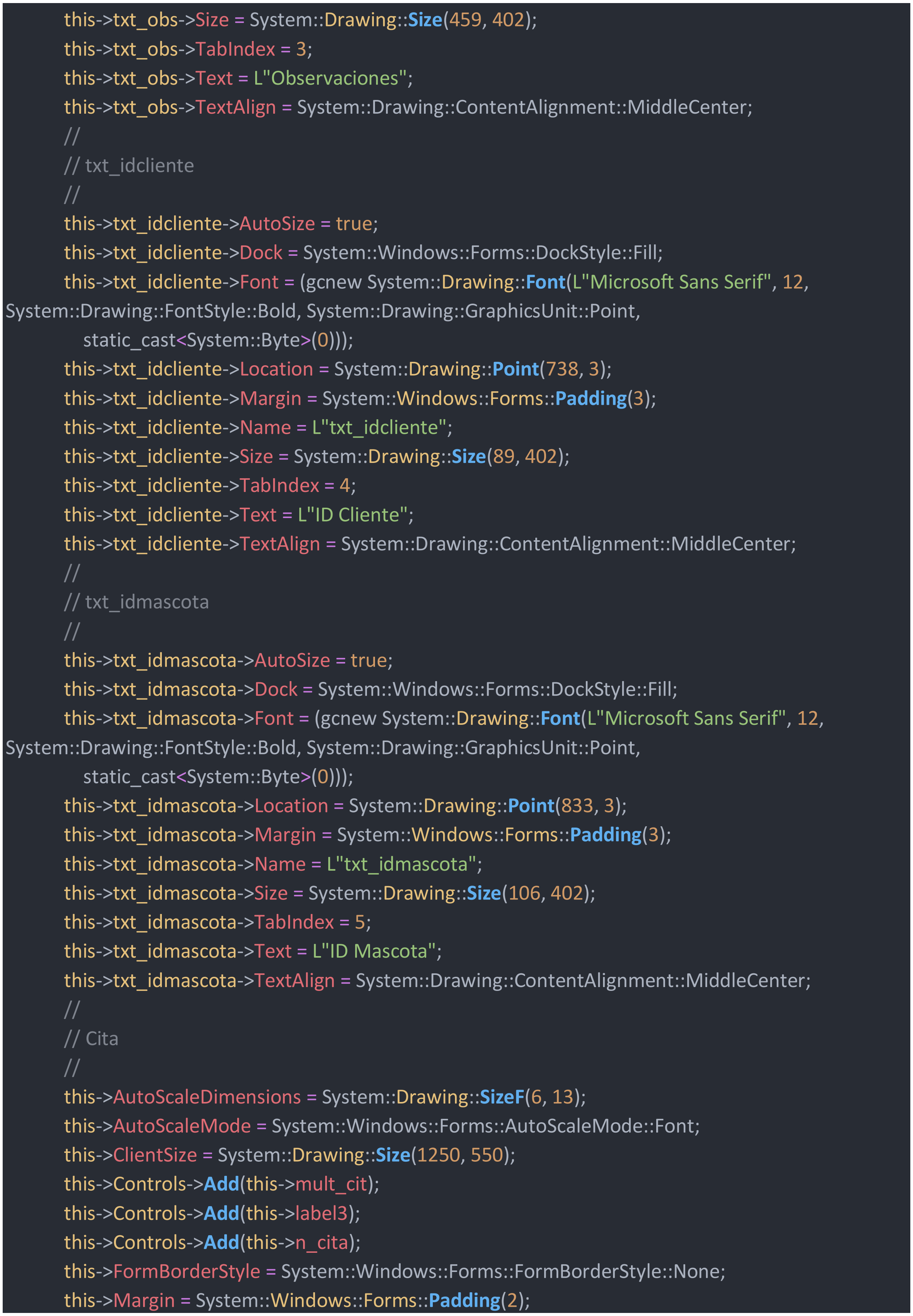
ID	Hora	Fecha	Observaciones	ID Cliente	ID Mascota
2	10:30	19/08/2022	Recoger al perro desde casa	2	2
3	18:40	23/08/2022	Bañar con agua tíbia pq es cachorro	2	3
4	12:15	13/09/2022	Bañar con abundante jabón	1	4
5	11:45	12/08/2022	Limpieza de escamas, se recomienda sedar	4	6
6	16:23	5/08/2022	Cortar uñas y felicitar a la dueña por su cumpleaños	1	1
7	09:11	16/10/2022	Presentar ropa de su talla y luego cortar su pelo	4	4
8	04:30	21/08/2022	Mordió al dueño mientras jugaban, se le dió el antidoto	5	5
9	18:30	16/08/2022	Mini entrenamiento	6	6

```
#pragma once
#pragma once
#include "addCitas.h"

namespace VetPro {

    using namespace System;
    using namespace System::ComponentModel;
    using namespace System::Collections;
    using namespace System::Windows::Forms;
    using namespace System::Data;
    using namespace System::Drawing;
    using namespace MySql::Data::MySqlClient;

    public ref class Cita : public System::Windows::Forms::Form {
    public:
        Cita(void) {
            InitializeComponent();
            load_TablaCitas();
        }

    protected:
        ~Cita() {
            if (components) {
                delete components;
            }
        }
```

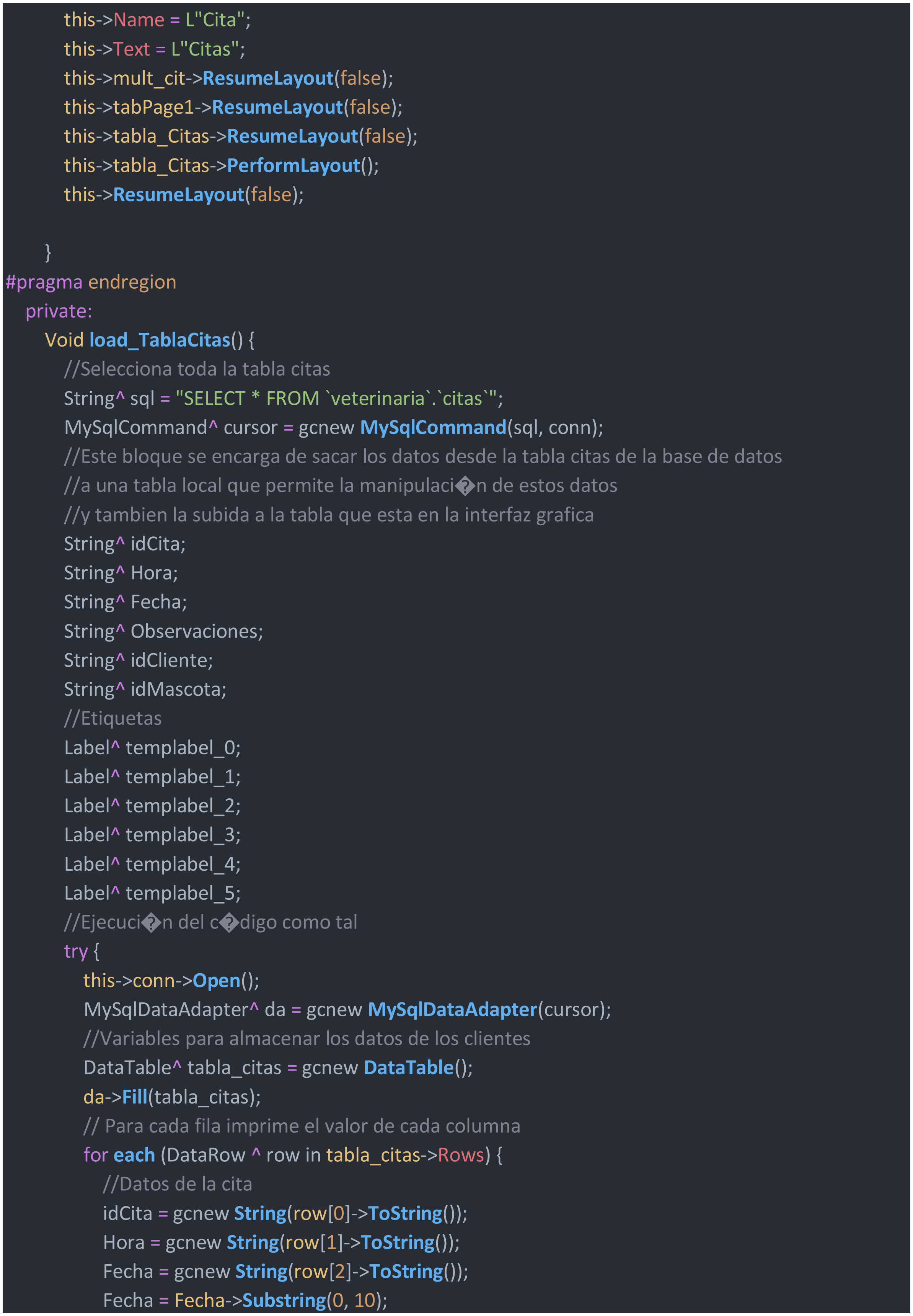


```
   private: String^ connectionString = "datasource=localhost; username=root; password=1234;
database=veterinaria";
   private: MySqlConnection^ conn = gcnew MySqlConnection(connectionString);
   private: System::Windows::Forms::Button^ n_cita;
   private: System::Windows::Forms::Label^ label3;
   private: System::Windows::Forms::TabControl^ mult_cit;
   private: System::Windows::Forms::TabPage^ tabPage1;
   private: System::Windows::Forms::TableLayoutPanel^ tabla_Citas;
   private: System::Windows::Forms::Label^ txt_id;
   private: System::Windows::Forms::Label^ txt_hora;
   private: System::Windows::Forms::Label^ txt_fecha;
   private: System::Windows::Forms::Label^ txt_obs;
   private: System::Windows::Forms::Label^ txt_idcliente;
   private: System::Windows::Forms::Label^ txt_idmascota;

   private: System::ComponentModel::Container^ components;

#pragma region Windows Form Designer generated code
      /// <summary>
      /// M�todo necesario para admitir el Dise�ador. No se puede modificar
      /// el contenido de este m�todo con el editor de c�digo.
      /// </summary>
      void InitializeComponent(void) {
         this->n_cita = (gcnew System::Windows::Forms::Button());
         this->label3 = (gcnew System::Windows::Forms::Label());
         this->mult_cit = (gcnew System::Windows::Forms::TabControl());
         this->tabPage1 = (gcnew System::Windows::Forms::TabPage());
         this->tabla_Citas = (gcnew System::Windows::Forms::TableLayoutPanel());
         this->txt_id = (gcnew System::Windows::Forms::Label());
         this->txt_hora = (gcnew System::Windows::Forms::Label());
         this->txt_fecha = (gcnew System::Windows::Forms::Label());
         this->txt_obs = (gcnew System::Windows::Forms::Label());
         this->txt_idcliente = (gcnew System::Windows::Forms::Label());
         this->txt_idmascota = (gcnew System::Windows::Forms::Label());
         this->mult_cit->SuspendLayout();
         this->tabPage1->SuspendLayout();
         this->tabla_Citas->SuspendLayout();
         this->SuspendLayout();
         //
         // n_cita
         //
         this->n_cita->BackColor =
System::Drawing::Color::FromArgb(static_cast<System::Int32>(static_cast<System::Byte>(255)),
static_cast<System::Int32>(static_cast<System::Byte>(174)),
            static_cast<System::Int32>(static_cast<System::Byte>(30)));
```

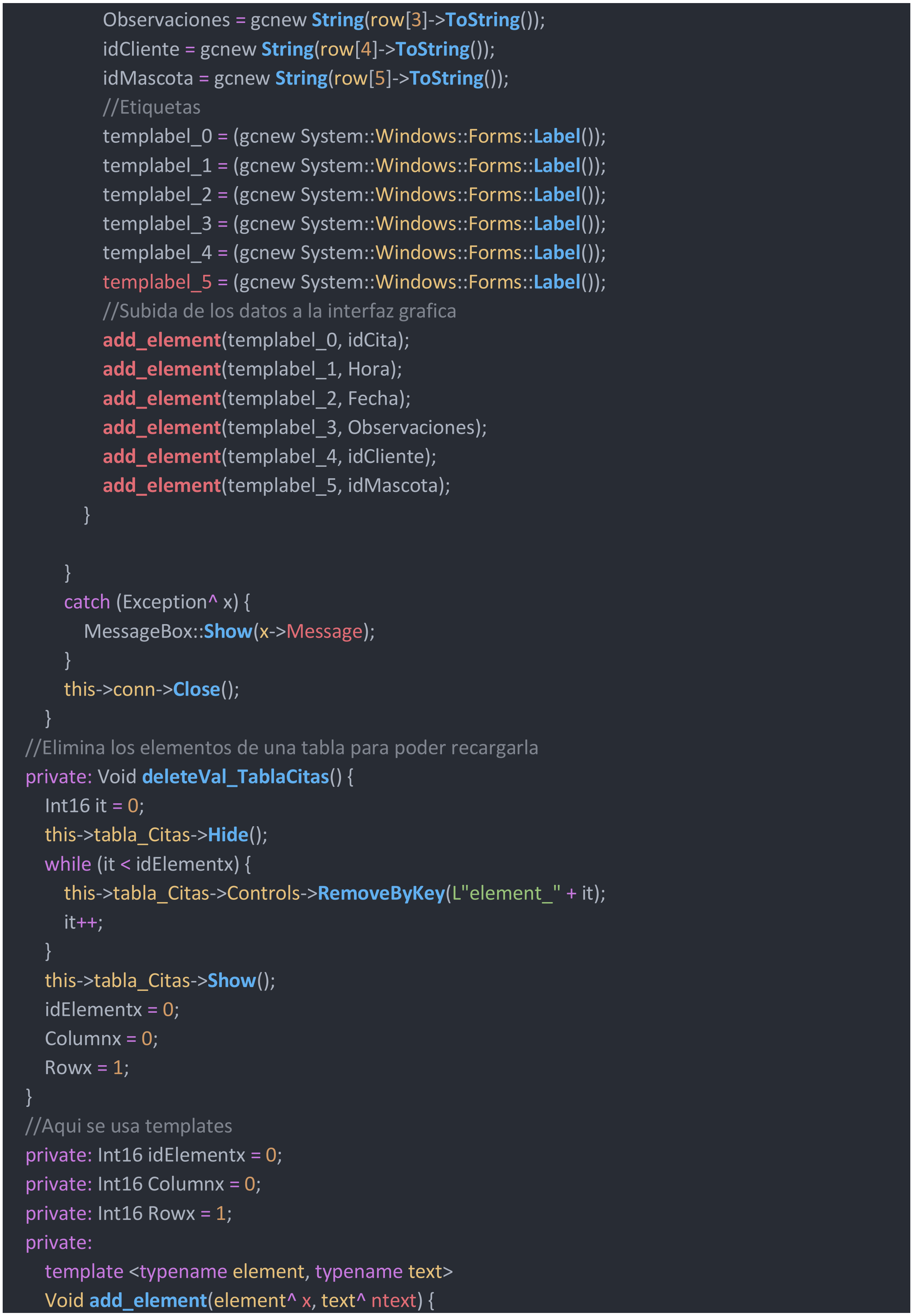
```
        this->n_cita->Font = (gcnew System::Drawing::Font(L"Microsoft Sans Serif", 16,
System::Drawing::FontStyle::Bold));
        this->n_cita->ForeColor = System::Drawing::SystemColors::ButtonHighlight;
        this->n_cita->Location = System::Drawing::Point(1013, 288);
        this->n_cita->Margin = System::Windows::Forms::Padding(2);
        this->n_cita->Name = L"n_cita";
        this->n_cita->Size = System::Drawing::Size(196, 64);
        this->n_cita->TabIndex = 0;
        this->n_cita->Text = L"Nueva Cita";
        this->n_cita->UseVisualStyleBackColor = false;
        this->n_cita->Click += gcnew System::EventHandler(this, &Cita::n_cita_Click);
        //
        // label3
        //
        this->label3->Font = (gcnew System::Drawing::Font(L"Microsoft Sans Serif", 40.2F,
System::Drawing::FontStyle::Bold, System::Drawing::GraphicsUnit::Point,
          static_cast<System::Byte>(0)));
        this->label3->ForeColor =
System::Drawing::Color::FromArgb(static_cast<System::Int32>(static_cast<System::Byte>(255)),
static_cast<System::Int32>(static_cast<System::Byte>(174)),
          static_cast<System::Int32>(static_cast<System::Byte>(30)));
        this->label3->Location = System::Drawing::Point(391, 19);
        this->label3->Name = L"label3";
        this->label3->Size = System::Drawing::Size(429, 84);
        this->label3->TabIndex = 36;
        this->label3->Text = L"Citas";
        this->label3->TextAlign = System::Drawing::ContentAlignment::MiddleCenter;
        //
        // mult_cit
        //
        this->mult_cit->Controls->Add(this->tabPage1);
        this->mult_cit->Font = (gcnew System::Drawing::Font(L"Microsoft Sans Serif", 11.25F,
System::Drawing::FontStyle::Regular, System::Drawing::GraphicsUnit::Point,
          static_cast<System::Byte>(0)));
        this->mult_cit->Location = System::Drawing::Point(40, 95);
        this->mult_cit->Name = L"mult_cit";
        this->mult_cit->SelectedIndex = 0;
        this->mult_cit->Size = System::Drawing::Size(954, 427);
        this->mult_cit->TabIndex = 42;
        //
        // tabPage1
        //
        this->tabPage1->Controls->Add(this->tabla_Citas);
        this->tabPage1->Location = System::Drawing::Point(4, 27);
        this->tabPage1->Name = L"tabPage1";
        this->tabPage1->Padding = System::Windows::Forms::Padding(3);
        this->tabPage1->Size = System::Drawing::Size(946, 396);
        this->tabPage1->TabIndex = 0;
        this->tabPage1->Text = L"Citas";
        this->tabPage1->UseVisualStyleBackColor = true;
        //
        // tabla_Citas
        //
        this->tabla_Citas->ColumnCount = 6;
        this->tabla_Citas->ColumnStyles->Add((gcnew
System::Windows::Forms::ColumnStyle(System::Windows::Forms::SizeType::Absolute, 39)));
        this->tabla_Citas->ColumnStyles->Add((gcnew
System::Windows::Forms::ColumnStyle(System::Windows::Forms::SizeType::Percent, 11.97398F)));
        this->tabla_Citas->ColumnStyles->Add((gcnew
System::Windows::Forms::ColumnStyle(System::Windows::Forms::SizeType::Percent, 13.63627F)));
        this->tabla_Citas->ColumnStyles->Add((gcnew
System::Windows::Forms::ColumnStyle(System::Windows::Forms::SizeType::Percent, 51.55176F)));
        this->tabla_Citas->ColumnStyles->Add((gcnew
System::Windows::Forms::ColumnStyle(System::Windows::Forms::SizeType::Percent, 10.53208F)));
        this->tabla_Citas->ColumnStyles->Add((gcnew
System::Windows::Forms::ColumnStyle(System::Windows::Forms::SizeType::Percent, 12.30591F)));
        this->tabla_Citas->Controls->Add(this->txt_id, 0, 0);
        this->tabla_Citas->Controls->Add(this->txt_hora, 1, 0);
        this->tabla_Citas->Controls->Add(this->txt_fecha, 2, 0);
        this->tabla_Citas->Controls->Add(this->txt_obs, 3, 0);
        this->tabla_Citas->Controls->Add(this->txt_idcliente, 4, 0);
        this->tabla_Citas->Controls->Add(this->txt_idmascota, 5, 0);
        this->tabla_Citas->Location = System::Drawing::Point(6, 3);
        this->tabla_Citas->Name = L"tabla_Citas";
        this->tabla_Citas->RowCount = 1;
        this->tabla_Citas->RowStyles->Add((gcnew
System::Windows::Forms::RowStyle(System::Windows::Forms::SizeType::Absolute, 40)));
        this->tabla_Citas->Size = System::Drawing::Size(942, 408);
        this->tabla_Citas->TabIndex = 42;
        //
        // txt_id
        //
        this->txt_id->AutoSize = true;
        this->txt_id->Dock = System::Windows::Forms::DockStyle::Fill;
        this->txt_id->Font = (gcnew System::Drawing::Font(L"Microsoft Sans Serif", 12,
System::Drawing::FontStyle::Bold, System::Drawing::GraphicsUnit::Point,
            static_cast<System::Byte>(0)));
        this->txt_id->Location = System::Drawing::Point(3, 3);
        this->txt_id->Margin = System::Windows::Forms::Padding(3);
        this->txt_id->Name = L"txt_id";
        this->txt_id->Size = System::Drawing::Size(33, 402);
        this->txt_id->TabIndex = 0;
        this->txt_id->Text = L"ID";
        this->txt_id->TextAlign = System::Drawing::ContentAlignment::MiddleCenter;
        //
        // txt_hora
        //
        this->txt_hora->AutoSize = true;
        this->txt_hora->Dock = System::Windows::Forms::DockStyle::Fill;
        this->txt_hora->Font = (gcnew System::Drawing::Font(L"Microsoft Sans Serif", 12,
System::Drawing::FontStyle::Bold, System::Drawing::GraphicsUnit::Point,
            static_cast<System::Byte>(0)));
        this->txt_hora->Location = System::Drawing::Point(42, 3);
        this->txt_hora->Margin = System::Windows::Forms::Padding(3);
        this->txt_hora->Name = L"txt_hora";
        this->txt_hora->Size = System::Drawing::Size(102, 402);
        this->txt_hora->TabIndex = 1;
        this->txt_hora->Text = L"Hora";
        this->txt_hora->TextAlign = System::Drawing::ContentAlignment::MiddleCenter;
        //
        // txt_fecha
        //
        this->txt_fecha->AutoSize = true;
        this->txt_fecha->Dock = System::Windows::Forms::DockStyle::Fill;
        this->txt_fecha->Font = (gcnew System::Drawing::Font(L"Microsoft Sans Serif", 12,
System::Drawing::FontStyle::Bold, System::Drawing::GraphicsUnit::Point,
            static_cast<System::Byte>(0)));
        this->txt_fecha->Location = System::Drawing::Point(150, 3);
        this->txt_fecha->Margin = System::Windows::Forms::Padding(3);
        this->txt_fecha->Name = L"txt_fecha";
        this->txt_fecha->Size = System::Drawing::Size(117, 402);
        this->txt_fecha->TabIndex = 2;
        this->txt_fecha->Text = L"Fecha";
        this->txt_fecha->TextAlign = System::Drawing::ContentAlignment::MiddleCenter;
        //
        // txt_obs
        //
        this->txt_obs->AutoSize = true;
        this->txt_obs->Dock = System::Windows::Forms::DockStyle::Fill;
        this->txt_obs->Font = (gcnew System::Drawing::Font(L"Microsoft Sans Serif", 12,
System::Drawing::FontStyle::Bold, System::Drawing::GraphicsUnit::Point,
            static_cast<System::Byte>(0)));
        this->txt_obs->Location = System::Drawing::Point(273, 3);
        this->txt_obs->Margin = System::Windows::Forms::Padding(3);
        this->txt_obs->Name = L"txt_obs";
```



```
	this->txt_obs->Size = System::Drawing::Size(459, 402);
	this->txt_obs->TabIndex = 3;
	this->txt_obs->Text = L"Observaciones";
	this->txt_obs->TextAlign = System::Drawing::ContentAlignment::MiddleCenter;
	//
	// txt_idcliente
	//
	this->txt_idcliente->AutoSize = true;
	this->txt_idcliente->Dock = System::Windows::Forms::DockStyle::Fill;
	this->txt_idcliente->Font = (gcnew System::Drawing::Font(L"Microsoft Sans Serif", 12,
System::Drawing::FontStyle::Bold, System::Drawing::GraphicsUnit::Point,
		static_cast<System::Byte>(0)));
	this->txt_idcliente->Location = System::Drawing::Point(738, 3);
	this->txt_idcliente->Margin = System::Windows::Forms::Padding(3);
	this->txt_idcliente->Name = L"txt_idcliente";
	this->txt_idcliente->Size = System::Drawing::Size(89, 402);
	this->txt_idcliente->TabIndex = 4;
	this->txt_idcliente->Text = L"ID Cliente";
	this->txt_idcliente->TextAlign = System::Drawing::ContentAlignment::MiddleCenter;
	//
	// txt_idmascota
	//
	this->txt_idmascota->AutoSize = true;
	this->txt_idmascota->Dock = System::Windows::Forms::DockStyle::Fill;
	this->txt_idmascota->Font = (gcnew System::Drawing::Font(L"Microsoft Sans Serif", 12,
System::Drawing::FontStyle::Bold, System::Drawing::GraphicsUnit::Point,
		static_cast<System::Byte>(0)));
	this->txt_idmascota->Location = System::Drawing::Point(833, 3);
	this->txt_idmascota->Margin = System::Windows::Forms::Padding(3);
	this->txt_idmascota->Name = L"txt_idmascota";
	this->txt_idmascota->Size = System::Drawing::Size(106, 402);
	this->txt_idmascota->TabIndex = 5;
	this->txt_idmascota->Text = L"ID Mascota";
	this->txt_idmascota->TextAlign = System::Drawing::ContentAlignment::MiddleCenter;
	//
	// Cita
	//
	this->AutoScaleDimensions = System::Drawing::SizeF(6, 13);
	this->AutoScaleMode = System::Windows::Forms::AutoScaleMode::Font;
	this->ClientSize = System::Drawing::Size(1250, 550);
	this->Controls->Add(this->mult_cit);
	this->Controls->Add(this->label3);
	this->Controls->Add(this->n_cita);
	this->FormBorderStyle = System::Windows::Forms::FormBorderStyle::None;
	this->Margin = System::Windows::Forms::Padding(2);
```

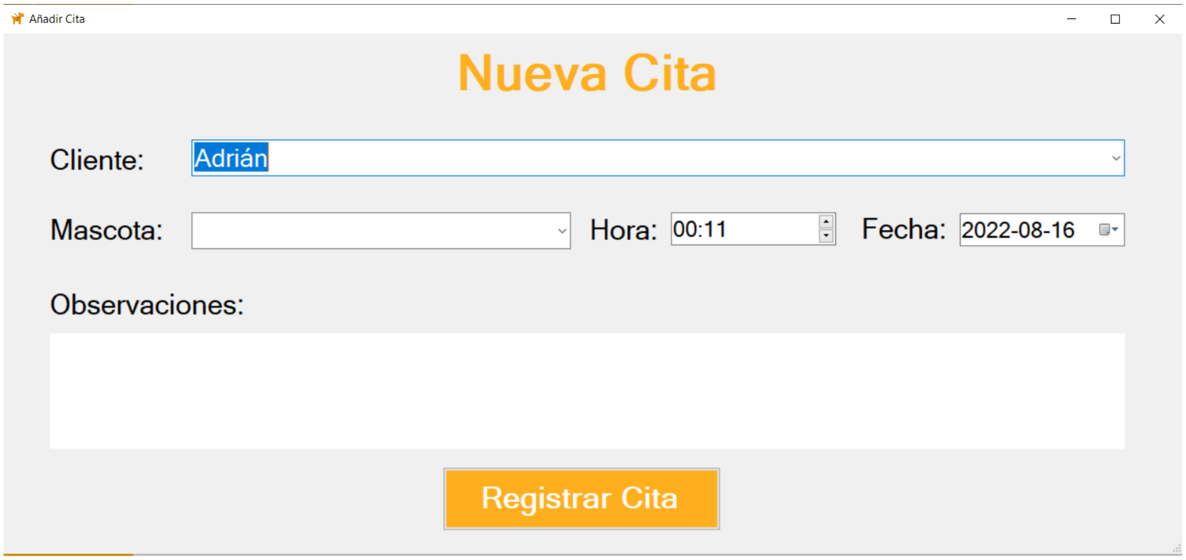


```
            this->Name = L"Cita";
            this->Text = L"Citas";
            this->mult_cit->ResumeLayout(false);
            this->tabPage1->ResumeLayout(false);
            this->tabla_Citas->ResumeLayout(false);
            this->tabla_Citas->PerformLayout();
            this->ResumeLayout(false);

        }
#pragma endregion
    private:
        Void load_TablaCitas() {
            //Selecciona toda la tabla citas
            String^ sql = "SELECT * FROM `veterinaria`.`citas`";
            MySqlCommand^ cursor = gcnew MySqlCommand(sql, conn);
            //Este bloque se encarga de sacar los datos desde la tabla citas de la base de datos
            //a una tabla local que permite la manipulaci�n de estos datos
            //y tambien la subida a la tabla que esta en la interfaz grafica
            String^ idCita;
            String^ Hora;
            String^ Fecha;
            String^ Observaciones;
            String^ idCliente;
            String^ idMascota;
            //Etiquetas
            Label^ templabel_0;
            Label^ templabel_1;
            Label^ templabel_2;
            Label^ templabel_3;
            Label^ templabel_4;
            Label^ templabel_5;
            //Ejecuci�n del c�digo como tal
            try {
                this->conn->Open();
                MySqlDataAdapter^ da = gcnew MySqlDataAdapter(cursor);
                //Variables para almacenar los datos de los clientes
                DataTable^ tabla_citas = gcnew DataTable();
                da->Fill(tabla_citas);
                // Para cada fila imprime el valor de cada columna
                for each (DataRow ^ row in tabla_citas->Rows) {
                    //Datos de la cita
                    idCita = gcnew String(row[0]->ToString());
                    Hora = gcnew String(row[1]->ToString());
                    Fecha = gcnew String(row[2]->ToString());
                    Fecha = Fecha->Substring(0, 10);
```



```
                Observaciones = gcnew String(row[3]->ToString());
                idCliente = gcnew String(row[4]->ToString());
                idMascota = gcnew String(row[5]->ToString());
                //Etiquetas
                templabel_0 = (gcnew System::Windows::Forms::Label());
                templabel_1 = (gcnew System::Windows::Forms::Label());
                templabel_2 = (gcnew System::Windows::Forms::Label());
                templabel_3 = (gcnew System::Windows::Forms::Label());
                templabel_4 = (gcnew System::Windows::Forms::Label());
                templabel_5 = (gcnew System::Windows::Forms::Label());
                //Subida de los datos a la interfaz grafica
                add_element(templabel_0, idCita);
                add_element(templabel_1, Hora);
                add_element(templabel_2, Fecha);
                add_element(templabel_3, Observaciones);
                add_element(templabel_4, idCliente);
                add_element(templabel_5, idMascota);
            }

        }
        catch (Exception^ x) {
            MessageBox::Show(x->Message);
        }
        this->conn->Close();
    }
//Elimina los elementos de una tabla para poder recargarla
private: Void deleteVal_TablaCitas() {
    Int16 it = 0;
    this->tabla_Citas->Hide();
    while (it < idElementx) {
        this->tabla_Citas->Controls->RemoveByKey(L"element_" + it);
        it++;
    }
    this->tabla_Citas->Show();
    idElementx = 0;
    Columnx = 0;
    Rowx = 1;
}
//Aqui se usa templates
private: Int16 idElementx = 0;
private: Int16 Columnx = 0;
private: Int16 Rowx = 1;
private:
    template <typename element, typename text>
    Void add_element(element^ x, text^ ntext) {
            x->AutoSize = true;
            x->Dock = System::Windows::Forms::DockStyle::Fill;
            x->Font = (gcnew System::Drawing::Font(L"Microsoft Sans Serif", 12,
System::Drawing::FontStyle::Regular, System::Drawing::GraphicsUnit::Point,
                static_cast<System::Byte>(0)));
            x->Margin = System::Windows::Forms::Padding(3);
            x->Name = L"sub_element_" + idElementx;
            x->Text = L"" + ntext;
            x->TextAlign = System::Drawing::ContentAlignment::MiddleCenter;
            this->tabla_Citas->Controls->Add(x, Columnx, Rowx);

            idElementx++;
            Columnx++;
            if (Columnx % 6 == 0) {
                Columnx = 0;
                Rowx++;
            }
        }
    //Abre la ventana addCitas
    private: Void n_cita_Click(System::Object^ sender, System::EventArgs^ e) {
        VetPro::addCitas^ nueva_Vent = gcnew VetPro::addCitas();
        nueva_Vent->ShowDialog();
        deleteVal_TablaCitas();
        load_TablaCitas();
    }
    };
}
```

Add Citas



Nueva Cita

Cliente: Adrián

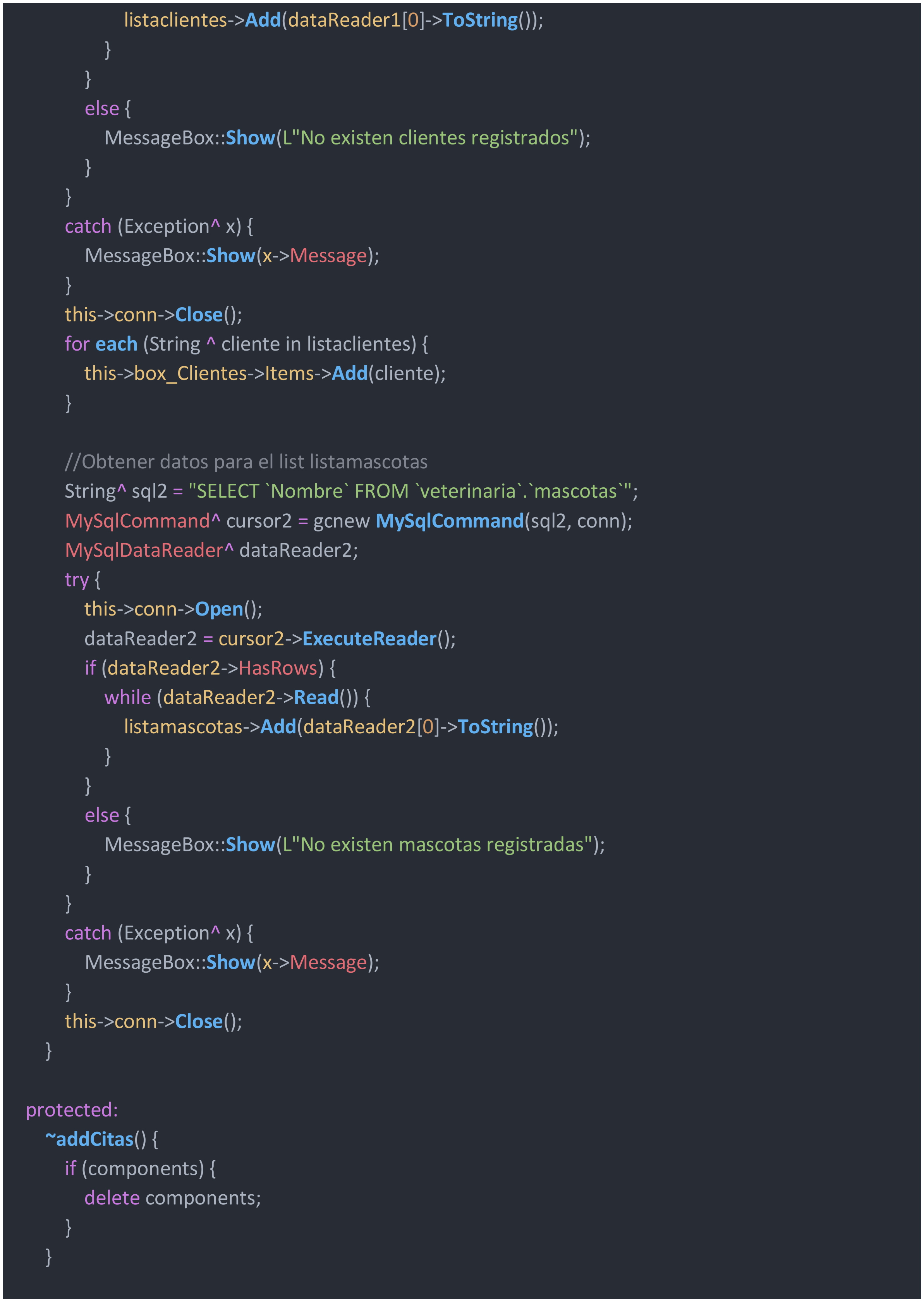
Mascota:

Hora: 00:11

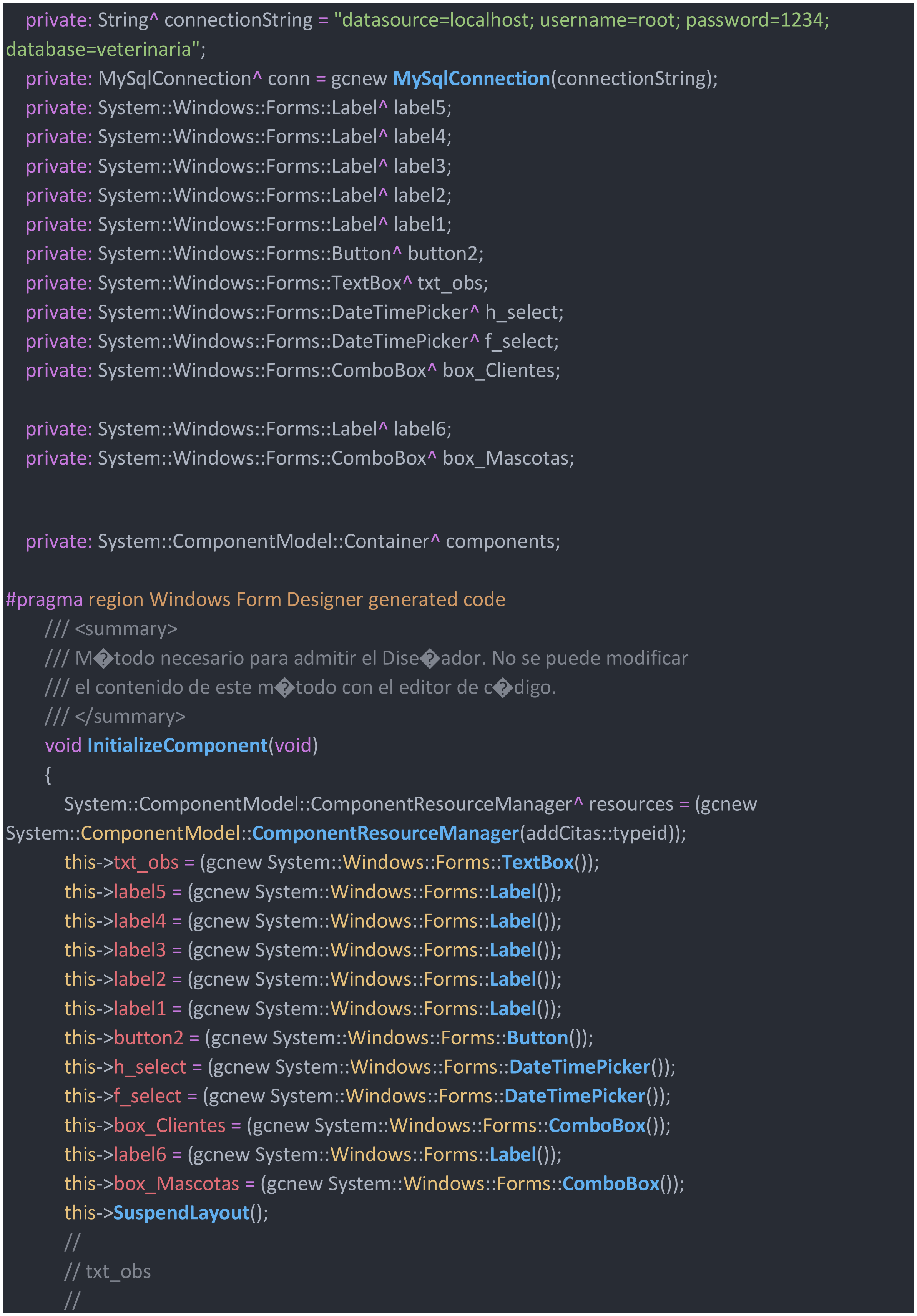
Fecha: 2022-08-16

Observaciones:

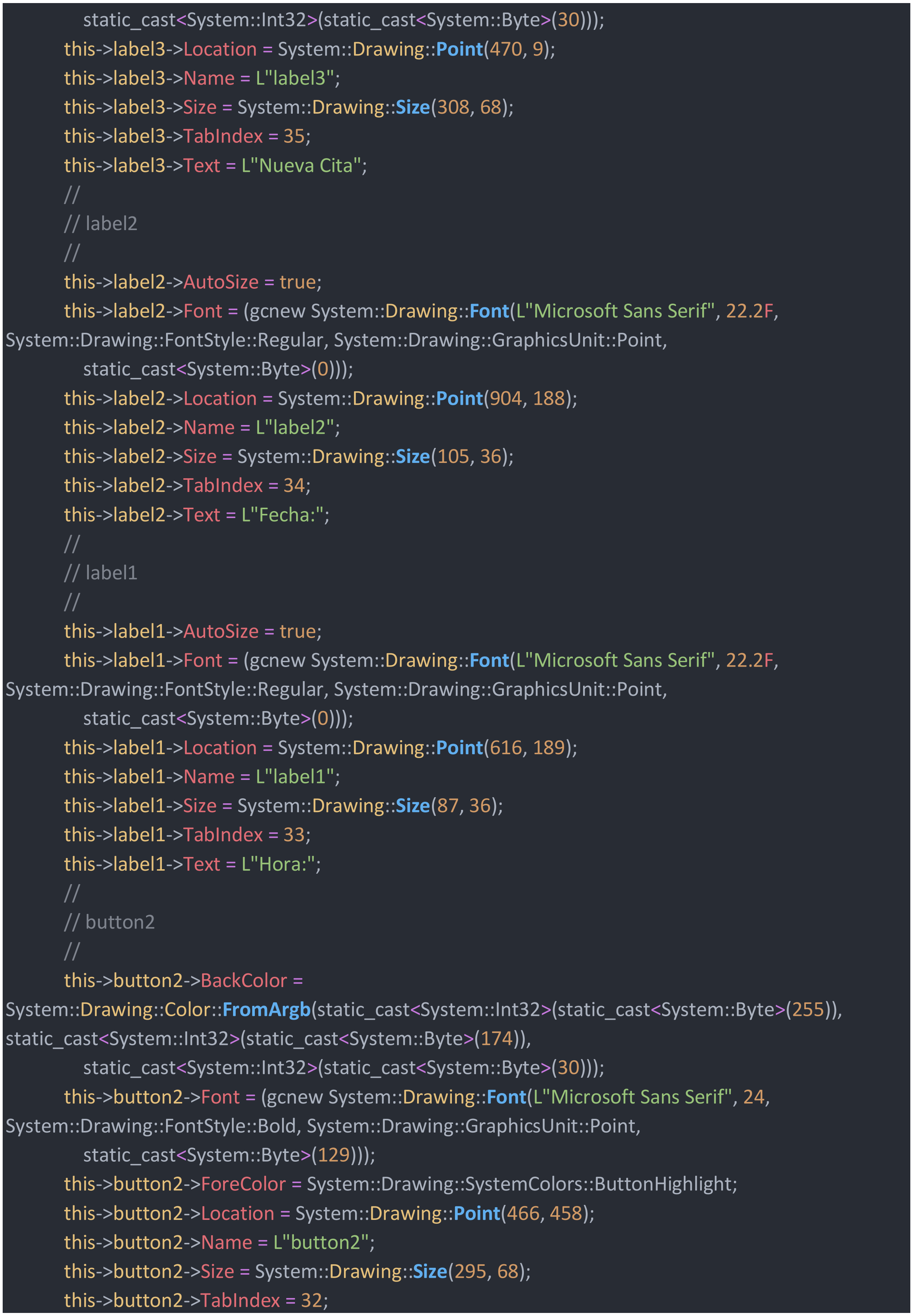
```
#pragma once
namespace VetPro {

  using namespace System;
  using namespace System::ComponentModel;
  using namespace System::Collections;
  using namespace System::Collections::Generic;
  using namespace System::Windows::Forms;
  using namespace System::Data;
  using namespace System::Drawing;
  using namespace MySql::Data::MySqlClient;

  /// <summary>
  /// Resumen de MyForm
  /// </summary>
  public ref class addCitas : public System::Windows::Forms::Form {
  public:
    addCitas(void) {
      InitializeComponent();
      //Insertar los nombres de los clientes en una list para agregarlos al ComboBox box_Clientes
      String^ sql1 = "SELECT `Nombres` FROM `veterinaria`.`clientes`";
      MySqlCommand^ cursor1 = gcnew MySqlCommand(sql1, conn);
      MySqlDataReader^ dataReader1;
      try {
        this->conn->Open();
        dataReader1 = cursor1->ExecuteReader();
        if (dataReader1->HasRows) {
          while (dataReader1->Read()) {
```



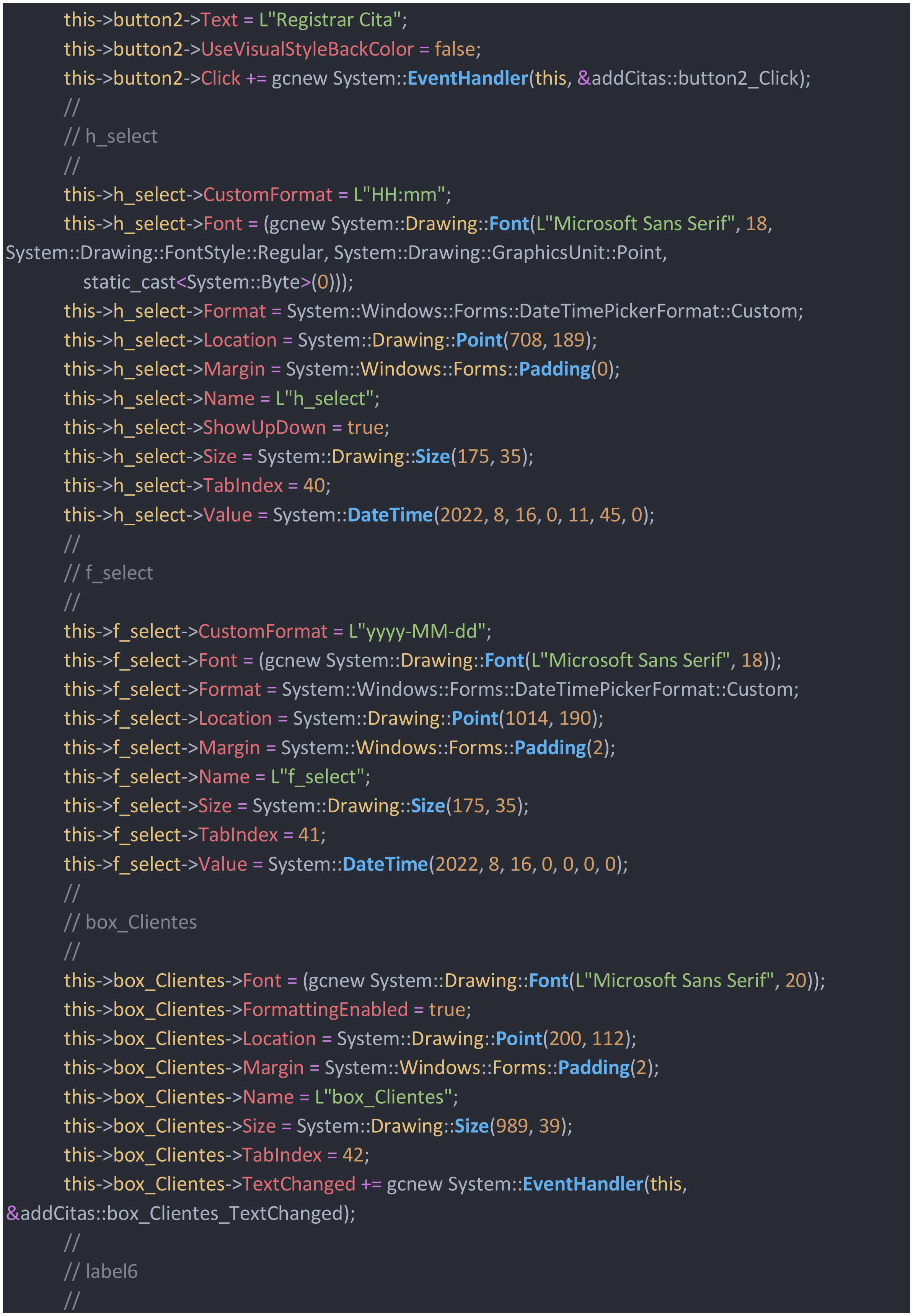
```
                    listaclientes->Add(dataReader1[0]->ToString());
                }
            }
            else {
                MessageBox::Show(L"No existen clientes registrados");
            }
        }
        catch (Exception^ x) {
            MessageBox::Show(x->Message);
        }
        this->conn->Close();
        for each (String ^ cliente in listaclientes) {
            this->box_Clientes->Items->Add(cliente);
        }

        //Obtener datos para el list listamascotas
        String^ sql2 = "SELECT `Nombre` FROM `veterinaria`.`mascotas`";
        MySqlCommand^ cursor2 = gcnew MySqlCommand(sql2, conn);
        MySqlDataReader^ dataReader2;
        try {
            this->conn->Open();
            dataReader2 = cursor2->ExecuteReader();
            if (dataReader2->HasRows) {
                while (dataReader2->Read()) {
                    listamascotas->Add(dataReader2[0]->ToString());
                }
            }
            else {
                MessageBox::Show(L"No existen mascotas registradas");
            }
        }
        catch (Exception^ x) {
            MessageBox::Show(x->Message);
        }
        this->conn->Close();
    }

protected:
    ~addCitas() {
        if (components) {
            delete components;
        }
    }
```



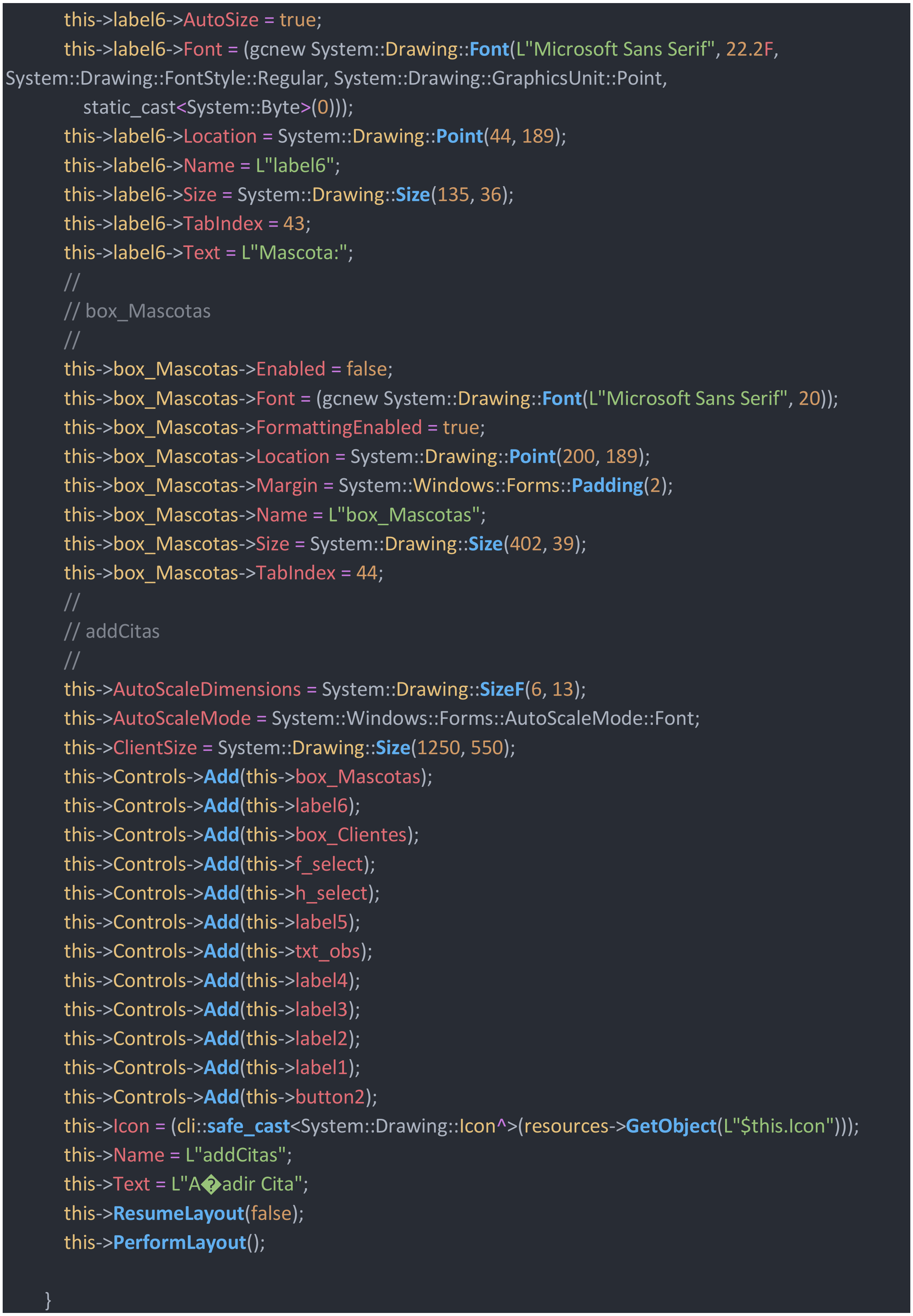
```
    private: String^ connectionString = "datasource=localhost; username=root; password=1234;
database=veterinaria";
    private: MySqlConnection^ conn = gcnew MySqlConnection(connectionString);
    private: System::Windows::Forms::Label^ label5;
    private: System::Windows::Forms::Label^ label4;
    private: System::Windows::Forms::Label^ label3;
    private: System::Windows::Forms::Label^ label2;
    private: System::Windows::Forms::Label^ label1;
    private: System::Windows::Forms::Button^ button2;
    private: System::Windows::Forms::TextBox^ txt_obs;
    private: System::Windows::Forms::DateTimePicker^ h_select;
    private: System::Windows::Forms::DateTimePicker^ f_select;
    private: System::Windows::Forms::ComboBox^ box_Clientes;

    private: System::Windows::Forms::Label^ label6;
    private: System::Windows::Forms::ComboBox^ box_Mascotas;


    private: System::ComponentModel::Container^ components;

#pragma region Windows Form Designer generated code
        /// <summary>
        /// M�todo necesario para admitir el Dise�ador. No se puede modificar
        /// el contenido de este m�todo con el editor de c�digo.
        /// </summary>
        void InitializeComponent(void)
        {
            System::ComponentModel::ComponentResourceManager^ resources = (gcnew
System::ComponentModel::ComponentResourceManager(addCitas::typeid));
            this->txt_obs = (gcnew System::Windows::Forms::TextBox());
            this->label5 = (gcnew System::Windows::Forms::Label());
            this->label4 = (gcnew System::Windows::Forms::Label());
            this->label3 = (gcnew System::Windows::Forms::Label());
            this->label2 = (gcnew System::Windows::Forms::Label());
            this->label1 = (gcnew System::Windows::Forms::Label());
            this->button2 = (gcnew System::Windows::Forms::Button());
            this->h_select = (gcnew System::Windows::Forms::DateTimePicker());
            this->f_select = (gcnew System::Windows::Forms::DateTimePicker());
            this->box_Clientes = (gcnew System::Windows::Forms::ComboBox());
            this->label6 = (gcnew System::Windows::Forms::Label());
            this->box_Mascotas = (gcnew System::Windows::Forms::ComboBox());
            this->SuspendLayout();
            //
            // txt_obs
            //
            this->txt_obs->Anchor =
static_cast<System::Windows::Forms::AnchorStyles>((((System::Windows::Forms::AnchorStyles::Top |
System::Windows::Forms::AnchorStyles::Bottom)
            | System::Windows::Forms::AnchorStyles::Left)
            | System::Windows::Forms::AnchorStyles::Right));
            this->txt_obs->BorderStyle = System::Windows::Forms::BorderStyle::None;
            this->txt_obs->Font = (gcnew System::Drawing::Font(L"Microsoft Sans Serif", 20));
            this->txt_obs->Location = System::Drawing::Point(50, 317);
            this->txt_obs->Multiline = true;
            this->txt_obs->Name = L"txt_obs";
            this->txt_obs->Size = System::Drawing::Size(1139, 122);
            this->txt_obs->TabIndex = 38;
            //
            // label5
            //
            this->label5->AutoSize = true;
            this->label5->Font = (gcnew System::Drawing::Font(L"Microsoft Sans Serif", 22.2F,
System::Drawing::FontStyle::Regular, System::Drawing::GraphicsUnit::Point,
          static_cast<System::Byte>(0)));
            this->label5->Location = System::Drawing::Point(44, 268);
            this->label5->Name = L"label5";
            this->label5->Size = System::Drawing::Size(221, 36);
            this->label5->TabIndex = 39;
            this->label5->Text = L"Observaciones:";
            //
            // label4
            //
            this->label4->AutoSize = true;
            this->label4->Font = (gcnew System::Drawing::Font(L"Microsoft Sans Serif", 22.2F,
System::Drawing::FontStyle::Regular, System::Drawing::GraphicsUnit::Point,
          static_cast<System::Byte>(0)));
            this->label4->Location = System::Drawing::Point(44, 115);
            this->label4->Name = L"label4";
            this->label4->Size = System::Drawing::Size(115, 36);
            this->label4->TabIndex = 37;
            this->label4->Text = L"Cliente:";
            //
            // label3
            //
            this->label3->Font = (gcnew System::Drawing::Font(L"Microsoft Sans Serif", 40.2F,
System::Drawing::FontStyle::Bold, System::Drawing::GraphicsUnit::Point,
          static_cast<System::Byte>(0)));
            this->label3->ForeColor =
System::Drawing::Color::FromArgb(static_cast<System::Int32>(static_cast<System::Byte>(255)),
static_cast<System::Int32>(static_cast<System::Byte>(174)),
```



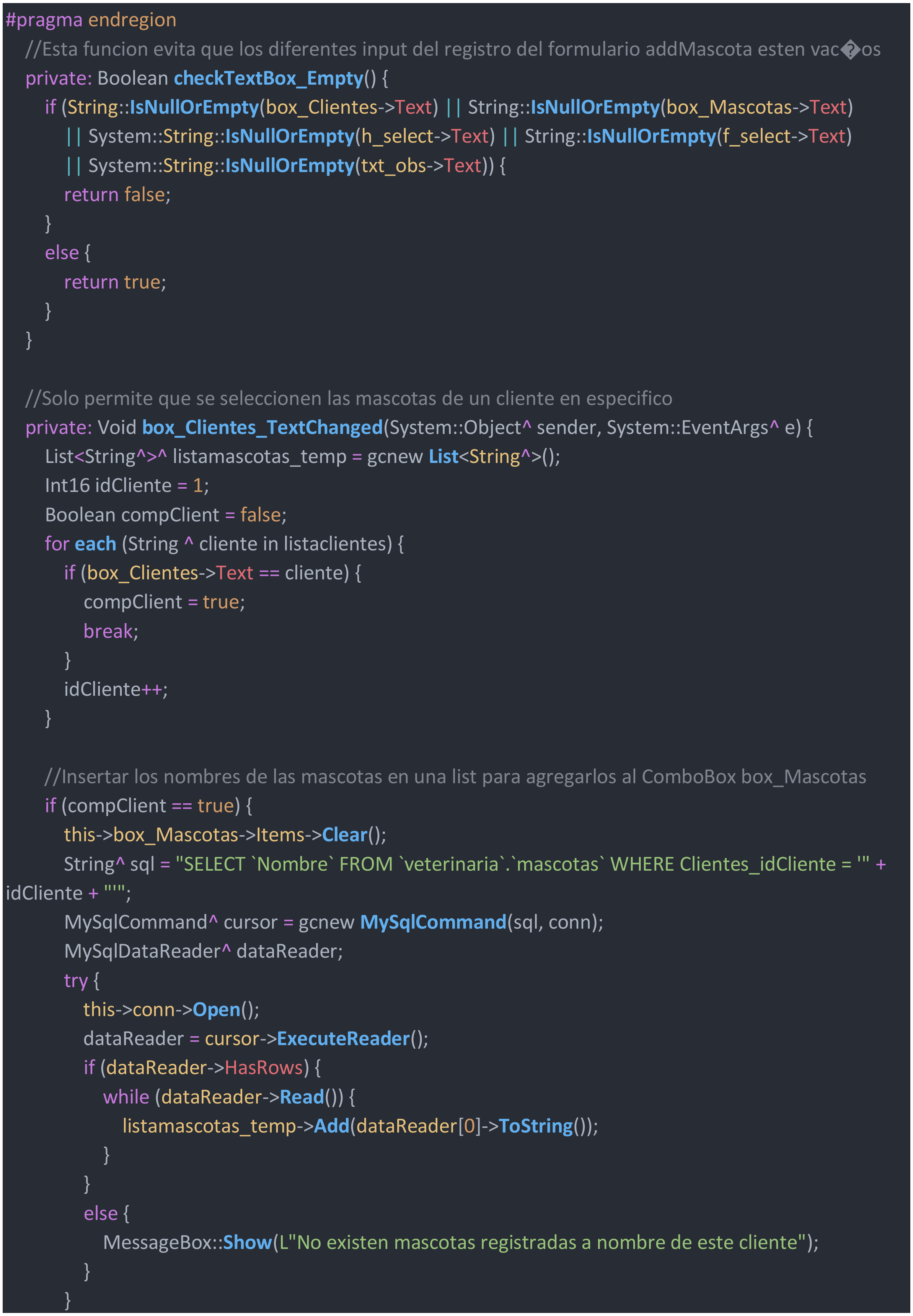
```
            static_cast<System::Int32>(static_cast<System::Byte>(30)));
        this->label3->Location = System::Drawing::Point(470, 9);
        this->label3->Name = L"label3";
        this->label3->Size = System::Drawing::Size(308, 68);
        this->label3->TabIndex = 35;
        this->label3->Text = L"Nueva Cita";
        //
        // label2
        //
        this->label2->AutoSize = true;
        this->label2->Font = (gcnew System::Drawing::Font(L"Microsoft Sans Serif", 22.2F,
System::Drawing::FontStyle::Regular, System::Drawing::GraphicsUnit::Point,
            static_cast<System::Byte>(0)));
        this->label2->Location = System::Drawing::Point(904, 188);
        this->label2->Name = L"label2";
        this->label2->Size = System::Drawing::Size(105, 36);
        this->label2->TabIndex = 34;
        this->label2->Text = L"Fecha:";
        //
        // label1
        //
        this->label1->AutoSize = true;
        this->label1->Font = (gcnew System::Drawing::Font(L"Microsoft Sans Serif", 22.2F,
System::Drawing::FontStyle::Regular, System::Drawing::GraphicsUnit::Point,
            static_cast<System::Byte>(0)));
        this->label1->Location = System::Drawing::Point(616, 189);
        this->label1->Name = L"label1";
        this->label1->Size = System::Drawing::Size(87, 36);
        this->label1->TabIndex = 33;
        this->label1->Text = L"Hora:";
        //
        // button2
        //
        this->button2->BackColor =
System::Drawing::Color::FromArgb(static_cast<System::Int32>(static_cast<System::Byte>(255)),
static_cast<System::Int32>(static_cast<System::Byte>(174)),
            static_cast<System::Int32>(static_cast<System::Byte>(30)));
        this->button2->Font = (gcnew System::Drawing::Font(L"Microsoft Sans Serif", 24,
System::Drawing::FontStyle::Bold, System::Drawing::GraphicsUnit::Point,
            static_cast<System::Byte>(129)));
        this->button2->ForeColor = System::Drawing::SystemColors::ButtonHighlight;
        this->button2->Location = System::Drawing::Point(466, 458);
        this->button2->Name = L"button2";
        this->button2->Size = System::Drawing::Size(295, 68);
        this->button2->TabIndex = 32;
```



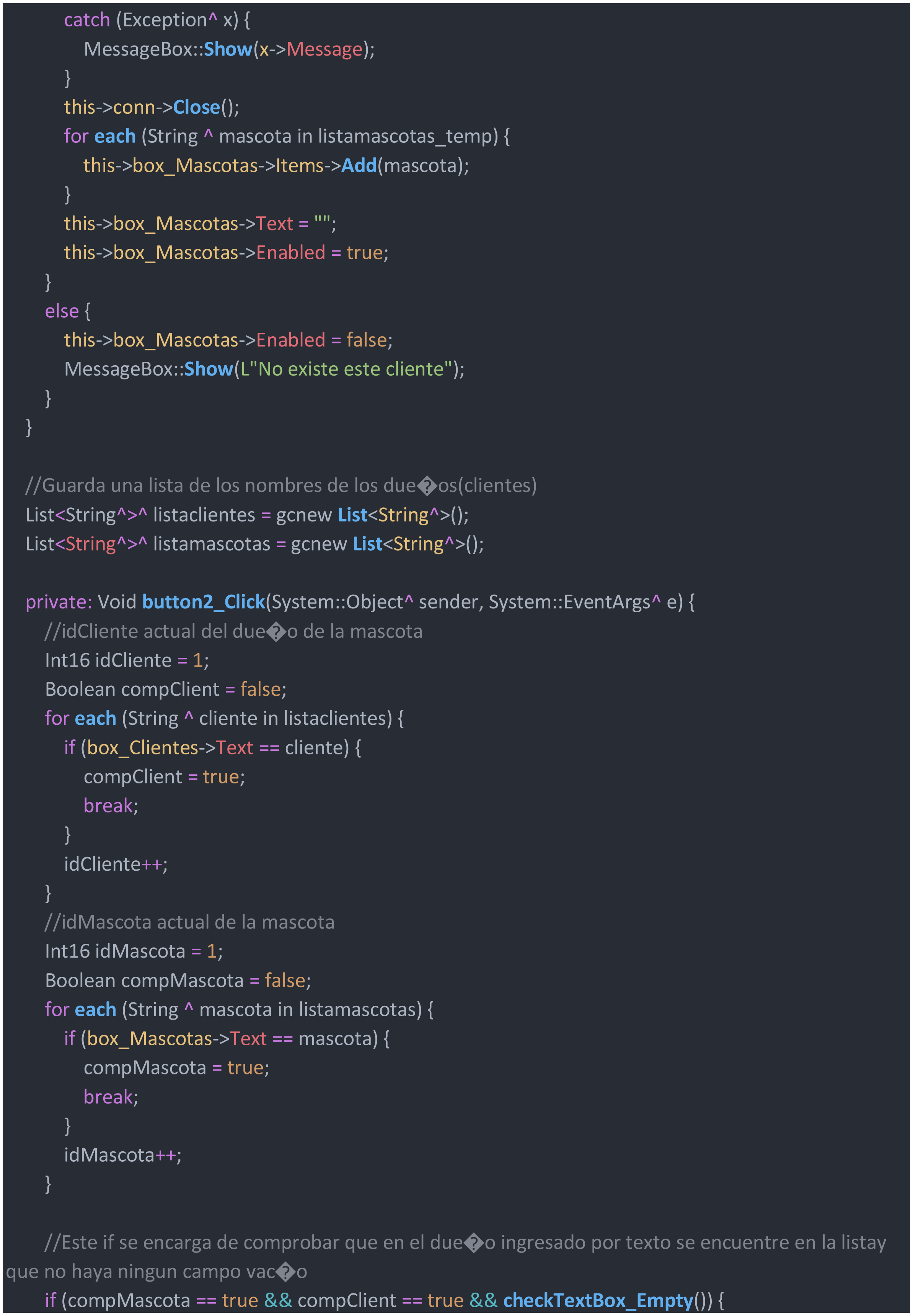
```
        this->button2->Text = L"Registrar Cita";
        this->button2->UseVisualStyleBackColor = false;
        this->button2->Click += gcnew System::EventHandler(this, &addCitas::button2_Click);
        //
        // h_select
        //
        this->h_select->CustomFormat = L"HH:mm";
        this->h_select->Font = (gcnew System::Drawing::Font(L"Microsoft Sans Serif", 18,
System::Drawing::FontStyle::Regular, System::Drawing::GraphicsUnit::Point,
          static_cast<System::Byte>(0)));
        this->h_select->Format = System::Windows::Forms::DateTimePickerFormat::Custom;
        this->h_select->Location = System::Drawing::Point(708, 189);
        this->h_select->Margin = System::Windows::Forms::Padding(0);
        this->h_select->Name = L"h_select";
        this->h_select->ShowUpDown = true;
        this->h_select->Size = System::Drawing::Size(175, 35);
        this->h_select->TabIndex = 40;
        this->h_select->Value = System::DateTime(2022, 8, 16, 0, 11, 45, 0);
        //
        // f_select
        //
        this->f_select->CustomFormat = L"yyyy-MM-dd";
        this->f_select->Font = (gcnew System::Drawing::Font(L"Microsoft Sans Serif", 18));
        this->f_select->Format = System::Windows::Forms::DateTimePickerFormat::Custom;
        this->f_select->Location = System::Drawing::Point(1014, 190);
        this->f_select->Margin = System::Windows::Forms::Padding(2);
        this->f_select->Name = L"f_select";
        this->f_select->Size = System::Drawing::Size(175, 35);
        this->f_select->TabIndex = 41;
        this->f_select->Value = System::DateTime(2022, 8, 16, 0, 0, 0, 0);
        //
        // box_Clientes
        //
        this->box_Clientes->Font = (gcnew System::Drawing::Font(L"Microsoft Sans Serif", 20));
        this->box_Clientes->FormattingEnabled = true;
        this->box_Clientes->Location = System::Drawing::Point(200, 112);
        this->box_Clientes->Margin = System::Windows::Forms::Padding(2);
        this->box_Clientes->Name = L"box_Clientes";
        this->box_Clientes->Size = System::Drawing::Size(989, 39);
        this->box_Clientes->TabIndex = 42;
        this->box_Clientes->TextChanged += gcnew System::EventHandler(this,
&addCitas::box_Clientes_TextChanged);
        //
        // label6
        //
```



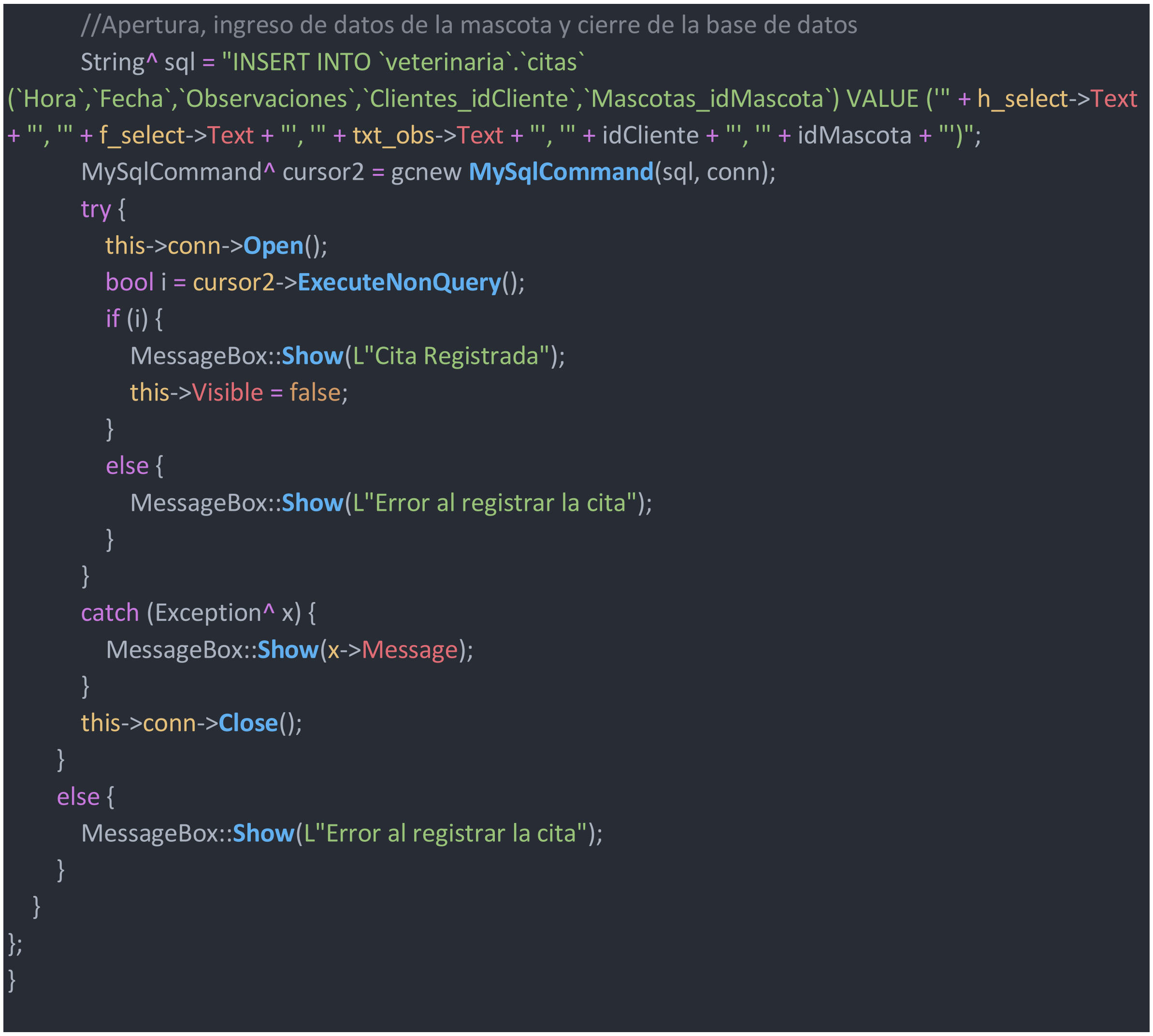
```
        this->label6->AutoSize = true;
        this->label6->Font = (gcnew System::Drawing::Font(L"Microsoft Sans Serif", 22.2F,
System::Drawing::FontStyle::Regular, System::Drawing::GraphicsUnit::Point,
            static_cast<System::Byte>(0)));
        this->label6->Location = System::Drawing::Point(44, 189);
        this->label6->Name = L"label6";
        this->label6->Size = System::Drawing::Size(135, 36);
        this->label6->TabIndex = 43;
        this->label6->Text = L"Mascota:";
        //
        // box_Mascotas
        //
        this->box_Mascotas->Enabled = false;
        this->box_Mascotas->Font = (gcnew System::Drawing::Font(L"Microsoft Sans Serif", 20));
        this->box_Mascotas->FormattingEnabled = true;
        this->box_Mascotas->Location = System::Drawing::Point(200, 189);
        this->box_Mascotas->Margin = System::Windows::Forms::Padding(2);
        this->box_Mascotas->Name = L"box_Mascotas";
        this->box_Mascotas->Size = System::Drawing::Size(402, 39);
        this->box_Mascotas->TabIndex = 44;
        //
        // addCitas
        //
        this->AutoScaleDimensions = System::Drawing::SizeF(6, 13);
        this->AutoScaleMode = System::Windows::Forms::AutoScaleMode::Font;
        this->ClientSize = System::Drawing::Size(1250, 550);
        this->Controls->Add(this->box_Mascotas);
        this->Controls->Add(this->label6);
        this->Controls->Add(this->box_Clientes);
        this->Controls->Add(this->f_select);
        this->Controls->Add(this->h_select);
        this->Controls->Add(this->label5);
        this->Controls->Add(this->txt_obs);
        this->Controls->Add(this->label4);
        this->Controls->Add(this->label3);
        this->Controls->Add(this->label2);
        this->Controls->Add(this->label1);
        this->Controls->Add(this->button2);
        this->Icon = (cli::safe_cast<System::Drawing::Icon^>(resources->GetObject(L"$this.Icon")));
        this->Name = L"addCitas";
        this->Text = L"A�adir Cita";
        this->ResumeLayout(false);
        this->PerformLayout();

      }
```



```
#pragma endregion
    //Esta funcion evita que los diferentes input del registro del formulario addMascota esten vac�os
    private: Boolean checkTextBox_Empty() {
        if (String::IsNullOrEmpty(box_Clientes->Text) || String::IsNullOrEmpty(box_Mascotas->Text)
            || System::String::IsNullOrEmpty(h_select->Text) || String::IsNullOrEmpty(f_select->Text)
            || System::String::IsNullOrEmpty(txt_obs->Text)) {
            return false;
        }
        else {
            return true;
        }
    }

    //Solo permite que se seleccionen las mascotas de un cliente en especifico
    private: Void box_Clientes_TextChanged(System::Object^ sender, System::EventArgs^ e) {
        List<String^>^ listamascotas_temp = gcnew List<String^>();
        Int16 idCliente = 1;
        Boolean compClient = false;
        for each (String ^ cliente in listaclientes) {
            if (box_Clientes->Text == cliente) {
                compClient = true;
                break;
            }
            idCliente++;
        }

        //Insertar los nombres de las mascotas en una list para agregarlos al ComboBox box_Mascotas
        if (compClient == true) {
            this->box_Mascotas->Items->Clear();
            String^ sql = "SELECT `Nombre` FROM `veterinaria`.`mascotas` WHERE Clientes_idCliente = '" +
idCliente + "'";
            MySqlCommand^ cursor = gcnew MySqlCommand(sql, conn);
            MySqlDataReader^ dataReader;
            try {
                this->conn->Open();
                dataReader = cursor->ExecuteReader();
                if (dataReader->HasRows) {
                    while (dataReader->Read()) {
                        listamascotas_temp->Add(dataReader[0]->ToString());
                    }
                }
                else {
                    MessageBox::Show(L"No existen mascotas registradas a nombre de este cliente");
                }
            }
```



```
        catch (Exception^ x) {
            MessageBox::Show(x->Message);
        }
        this->conn->Close();
        for each (String ^ mascota in listamascotas_temp) {
            this->box_Mascotas->Items->Add(mascota);
        }
        this->box_Mascotas->Text = "";
        this->box_Mascotas->Enabled = true;
    }
    else {
        this->box_Mascotas->Enabled = false;
        MessageBox::Show(L"No existe este cliente");
    }
}

//Guarda una lista de los nombres de los due�os(clientes)
List<String^>^ listaclientes = gcnew List<String^>();
List<String^>^ listamascotas = gcnew List<String^>();

private: Void button2_Click(System::Object^ sender, System::EventArgs^ e) {
    //idCliente actual del due�o de la mascota
    Int16 idCliente = 1;
    Boolean compClient = false;
    for each (String ^ cliente in listaclientes) {
        if (box_Clientes->Text == cliente) {
            compClient = true;
            break;
        }
        idCliente++;
    }
    //idMascota actual de la mascota
    Int16 idMascota = 1;
    Boolean compMascota = false;
    for each (String ^ mascota in listamascotas) {
        if (box_Mascotas->Text == mascota) {
            compMascota = true;
            break;
        }
        idMascota++;
    }

    //Este if se encarga de comprobar que en el due�o ingresado por texto se encuentre en la listay que no haya ningun campo vac�o
    if (compMascota == true && compClient == true && checkTextBox_Empty()) {
```



```
            //Apertura, ingreso de datos de la mascota y cierre de la base de datos
            String^ sql = "INSERT INTO `veterinaria`.`citas` (`Hora`,`Fecha`,`Observaciones`,`Clientes_idCliente`,`Mascotas_idMascota`) VALUE ('" + h_select->Text + "', '" + f_select->Text + "', '" + txt_obs->Text + "', '" + idCliente + "', '" + idMascota + "')";
            MySqlCommand^ cursor2 = gcnew MySqlCommand(sql, conn);
            try {
                this->conn->Open();
                bool i = cursor2->ExecuteNonQuery();
                if (i) {
                    MessageBox::Show(L"Cita Registrada");
                    this->Visible = false;
                }
                else {
                    MessageBox::Show(L"Error al registrar la cita");
                }
            }
            catch (Exception^ x) {
                MessageBox::Show(x->Message);
            }
            this->conn->Close();
        }
        else {
            MessageBox::Show(L"Error al registrar la cita");
        }
    }
};
}
```