

Creación de la Interfaz para la Práctica de Laboratorio

Para Trabajar con la Cola

Interfaz del trabajo práctico de Clínica Veterinaria:

Clínica Veterinaria

COLA

Fecha: 18/ 5/2025

Cola de Ingreso

Atencion en Consulta

Información del Dueño

Nombre

Información de Contacto

Hora de Llegada: 06:42:07

Numero de Mascotas: 1

Agregar

Eliminar

Limpiar

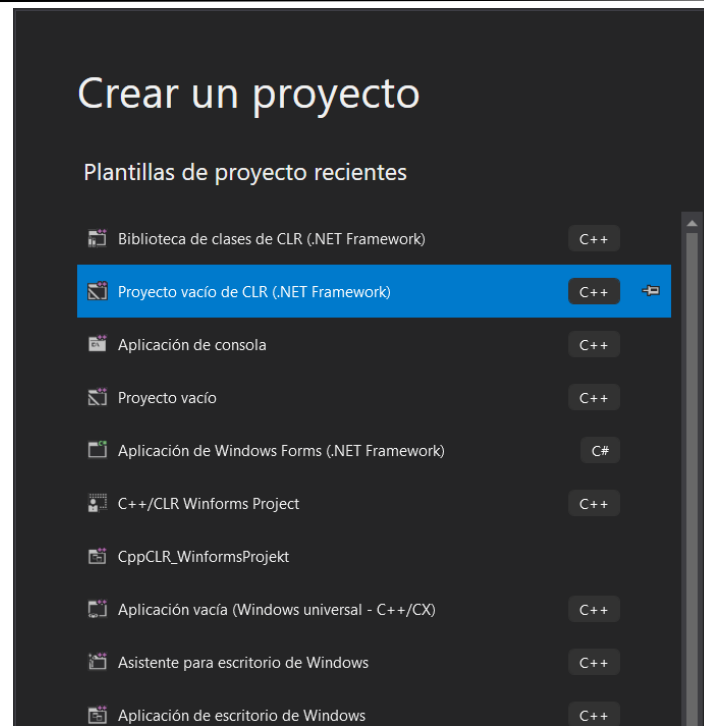
Ocultar la cola

COLA DE ATENCION

Dueño	Info Contacto	Hora Llegada	Cant Mascotas
-------	---------------	--------------	---------------

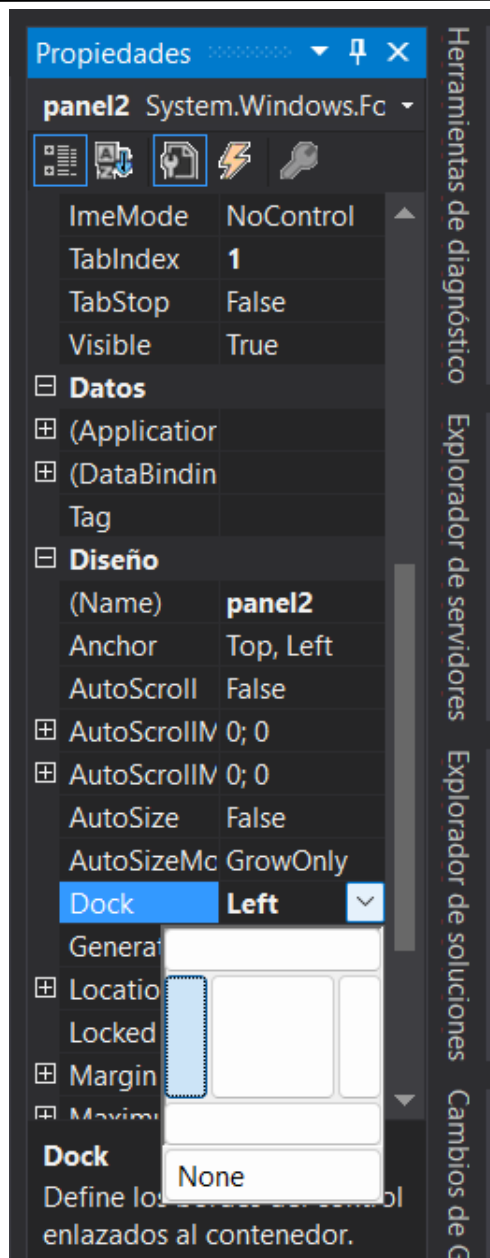
Instrucciones:

- 1.- Cree un nuevo proyecto, Proyecto vacío de CLR. (Como ya se le ha instruido en PV con VS 01), siga todos los pasos.



2- En formulario principal, Cambie el nombre del formulario, en la propiedad, Text del Formulario, ajustar el tamaño del formulario según lo necesario.

3.- Inserte 3 controles panel: Uno con la propiedad **Dock** a la izquierda, otro con la propiedad Dock al tope y otro con la propiedad Dock al centro

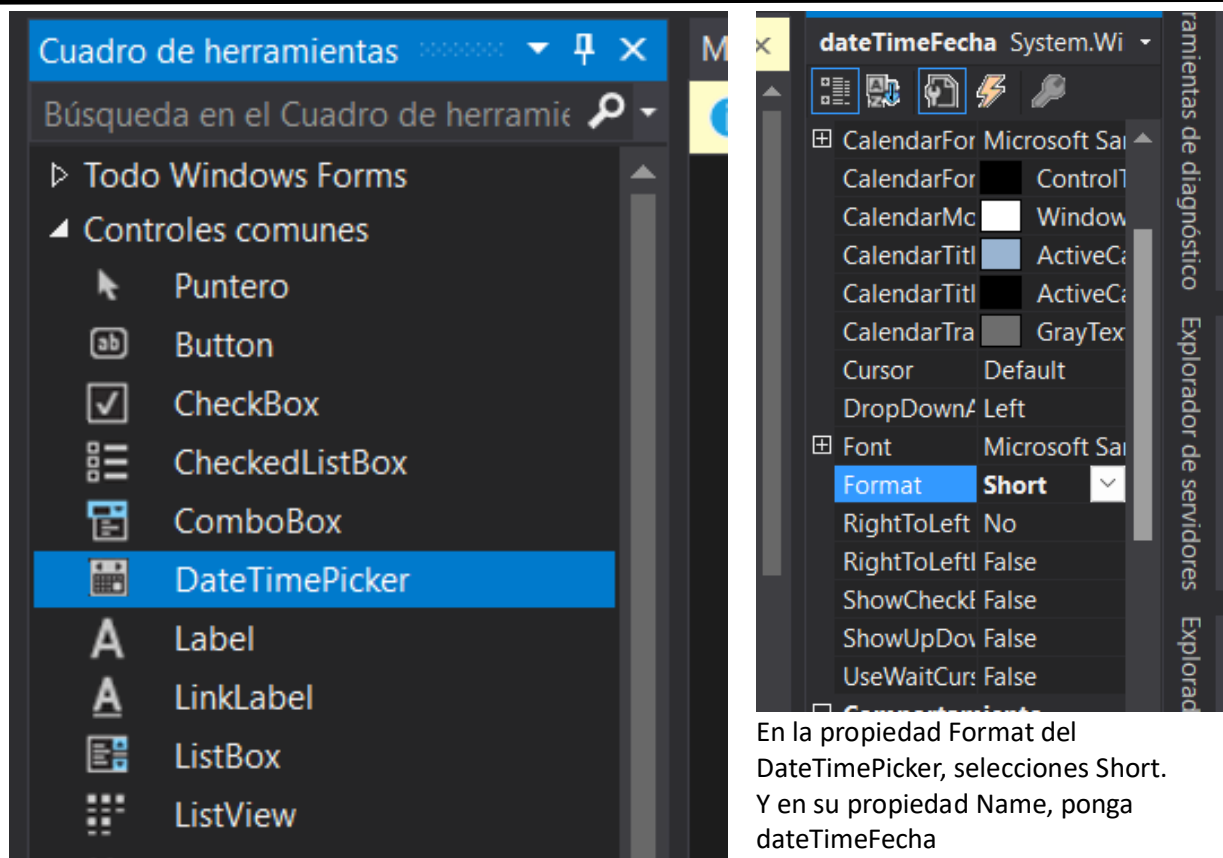


4.- En el panel de la izquierda inserte 2 controles button, uno se llamara Cola de Ingreso y otro Atención en consulta (ya sea veterinaria o consultorio), o Atención en Tienda (florería).

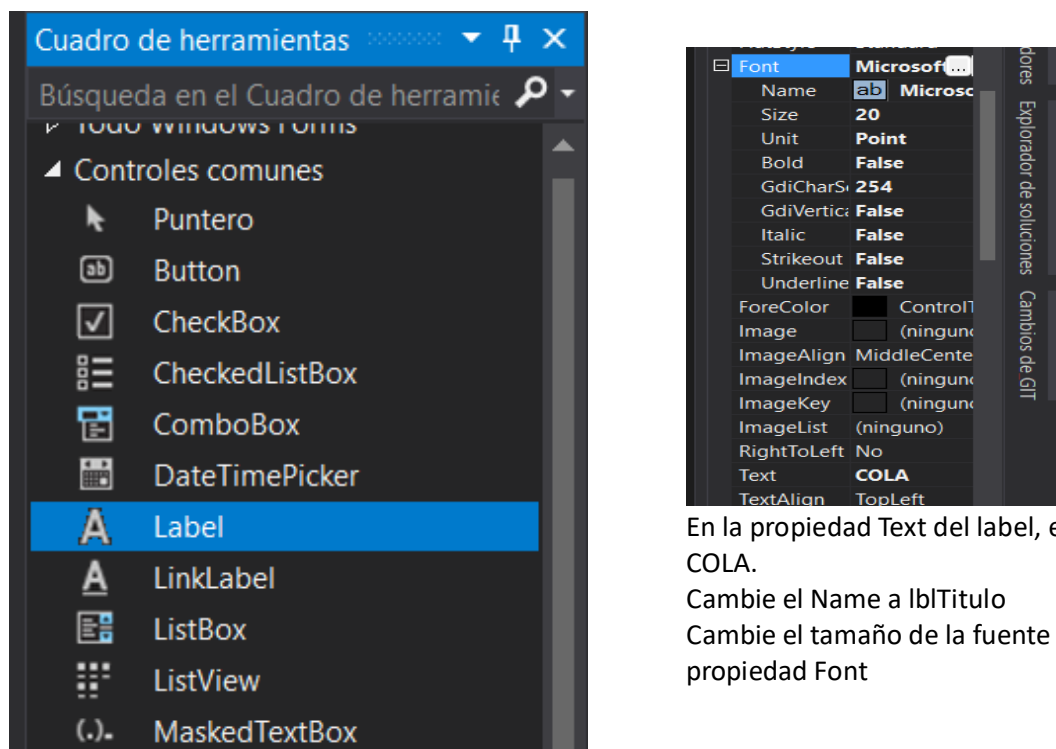
Al hacer Click en cola de ingreso, se trabajara con la cola.

Al hacer click en el segundo botón se trabajara con la lista.

5.- En el panel superior, inserte un control DateTimePicker y un control control label que tendrá el Nombre de lo que estemos haciendo.



En la propiedad Format del DateTimePicker, selecciones Short. Y en su propiedad Name, ponga dateTimeFecha



En la propiedad Text del label, escriba COLA.
Cambie el Name a lblTitulo
Cambie el tamaño de la fuente en la propiedad Font

6.- En el panel central inserte otro panel, pongale en la propiedad name panelCola

En este panel, insertara los controles necesarios para trabajar con la cola.

Clínica Veterinaria

COLA Fecha 18/ 5/2025

Cola de Ingreso

Atencion en Consulta

Información del Dueño

Nombre

Información de Contacto

Hora de Llegada 06:40:13

Numero de Mascotas 1

Agregar

Eliminar

Limpiar

Ocultar la cola

En el caso de la cola de la clínica veterinaria, pondrá nombre del dueño, Información de contacto, hora de llegada, numero de mascotas

7.- En el panel central, a la derecha inserte un label, ponga en la propiedad Text, COLA de ATENCION, cambien el tamaño de la fuente.

Clínica Veterinaria

COLA Fecha 18/ 5/2025

Cola de Ingreso

Atencion en Consulta

Información del Dueño

Nombre

Información de Contacto

Hora de Llegada 06:40:13

Numero de Mascotas 1

Agregar

Eliminar

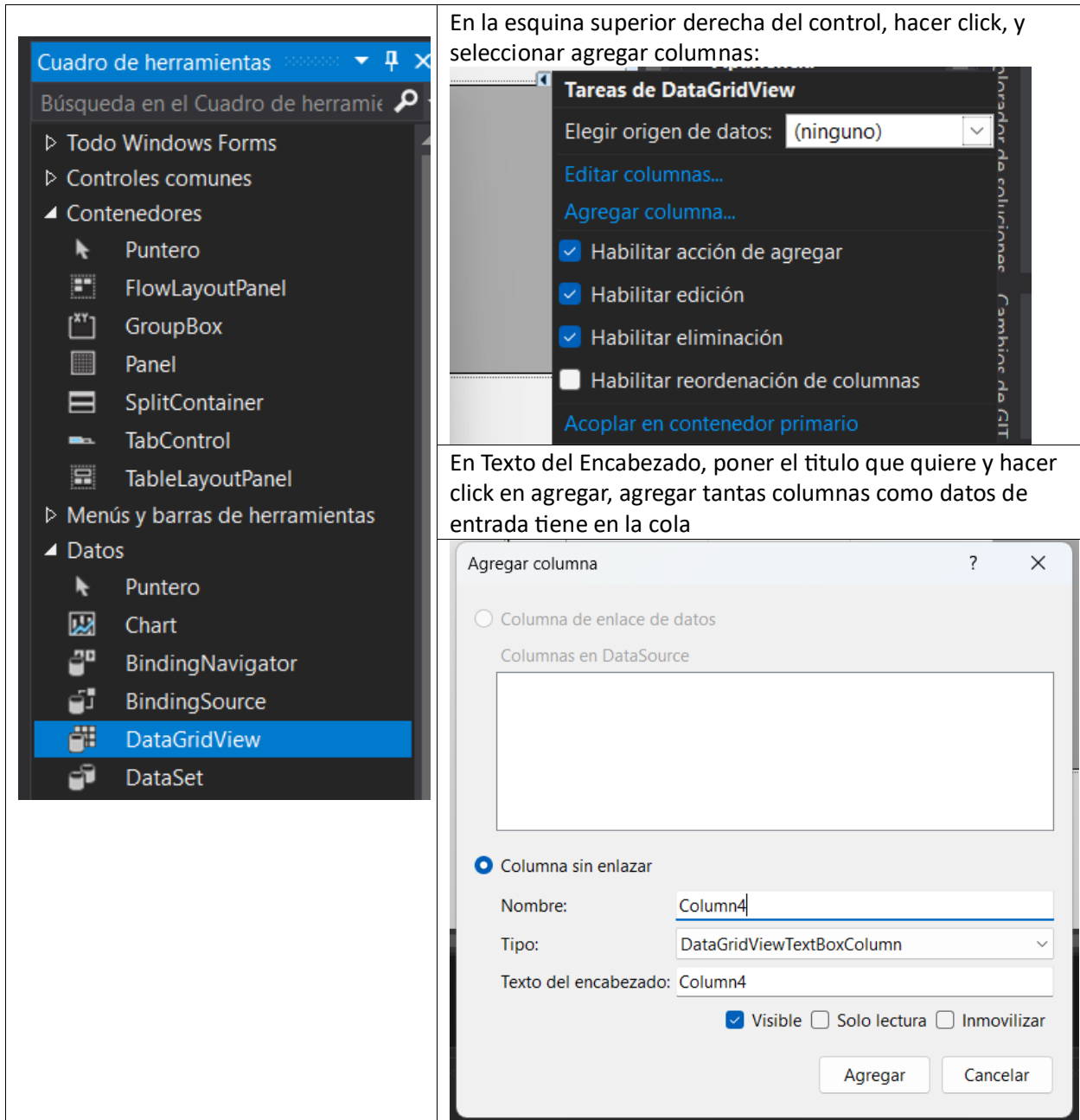
Limpiar

Ocultar la cola

COLA DE ATENCION

Para la Clínica de Veterinaria

7.- En el panel central, a la derecha, debajo del label COLA de ATENCION, insertar un DataGridView



En la esquina superior derecha del control, hacer click, y seleccionar agregar columnas:

En Texto del Encabezado, poner el titulo que quiere y hacer click en agregar, agregar tantas columnas como datos de entrada tiene en la cola

Agregar columna

☐ Columna de enlace de datos

Columnas en DataSource

☒ Columna sin enlazar

Nombre: Column4

Tipo: DataGridViewTextBoxColumn

Texto del encabezado: Column4

☒ Visible ☐ Solo lectura ☐ Inmovilizar

Agregar Cancelar

Quedando la pantalla de esta forma:

Para la Clinica Veterinaria:

COLA Fecha: 18/ 5/2025

Cola de Ingreso

Atencion en Consulta

Información del Dueño

Nombre:

Información de Contacto:

Hora de Llegada: 06:40:13

Numero de Mascotas: 1

Agregar

Eliminar

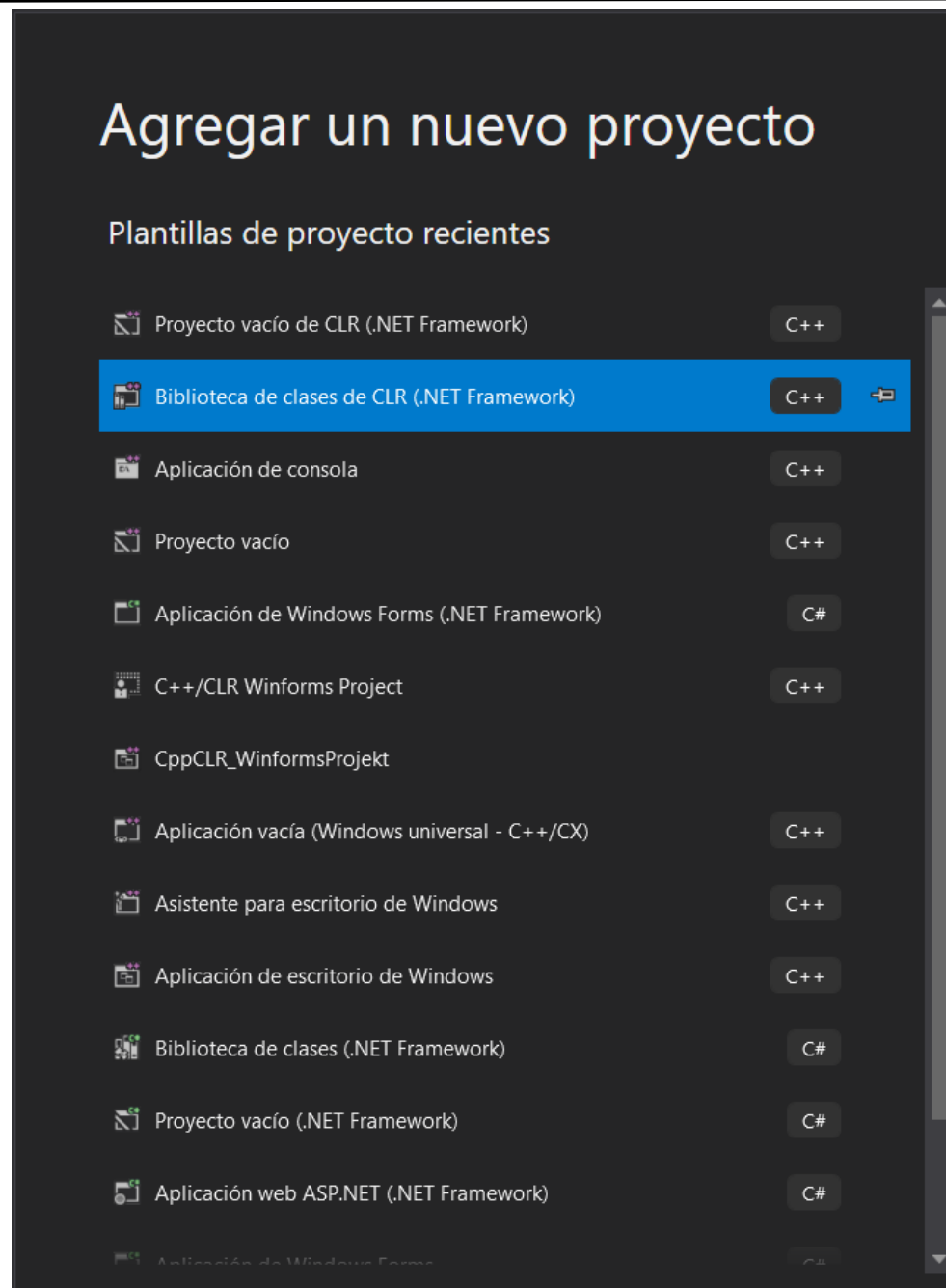
Limpiar

Ocultar la cola

COLA DE ATENCION

Dueño	Info Contacto	Hora Llegada	Cant Mascotas
*			

Ya tiene los controles necesarios, para trabajar con la cola, ahora deberá crear las clase para trabajar con los datos de la cola y la cola como tal, para ello se crea un nuevo proyecto, biblioteca de clases (explicado como crear esto, en el archivo PV con VS 03 Creación de las clases)



1.- Creará las clases necesarias:

Deberá adicionar al inicio de la clase:


```
using namespace std;
using namespace System;
using namespace System::Collections::Generic;
```

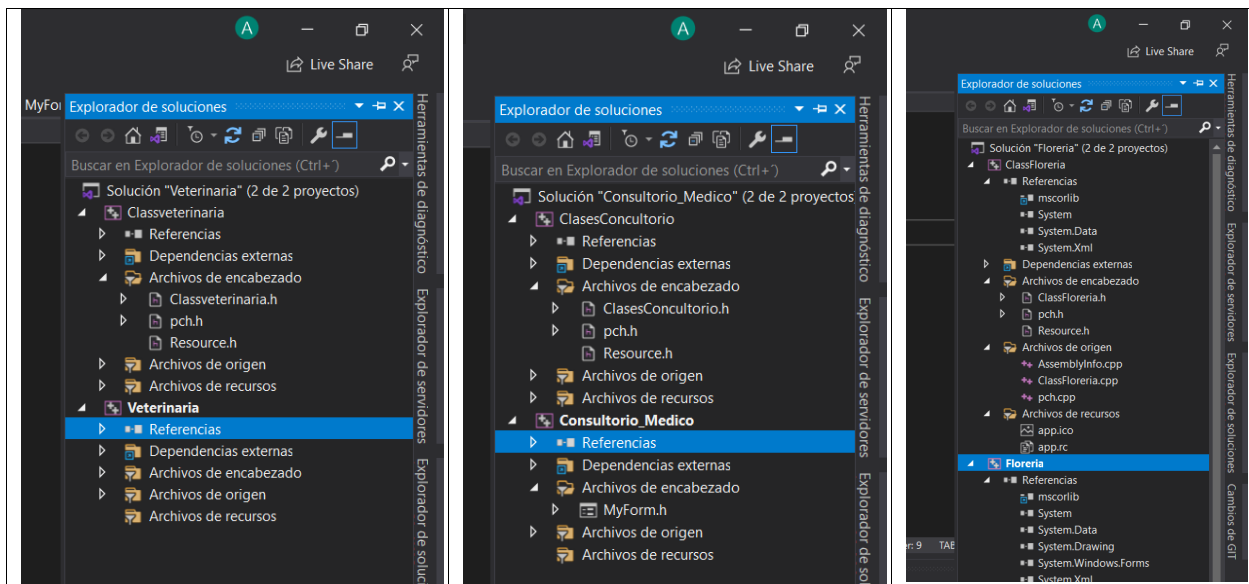
Para el Caso clinica Veterinaria creara la clase datos y cola_veterinaria

```
public ref class datos{
private:
String^ Dueño;
String^ Contacto;
String^ Hora_llegada;
int cantmascotas;
public:
datos(String^ Dueño, String^ Contacto, String^ Hora_llegada,int cantmascotas) {
this->Dueño = Dueño;
this->Contacto = Contacto;
this->Hora_llegada = Hora_llegada;
this->cantmascotas= cantmascotas;
}
String^ getDueño() { return Dueño; }
String^ getContacto() { return Contacto; }
String^ getHora_llegada() { return Hora_llegada; }
int getcantmascotas() { return cantmascotas; }
void setDueño(String^ _Dueño) { Dueño = _Dueño; }
void setContacto(String^ _Contacto) { Contacto = _Contacto; }
void setHora_llegada(String^ _Hora_llegada) { Hora_llegada = _Hora_llegada; }
void setcantmascotas(int _cantmascotas) { cantmascotas = _cantmascotas; }
};
public ref class cola_veterinaria
```

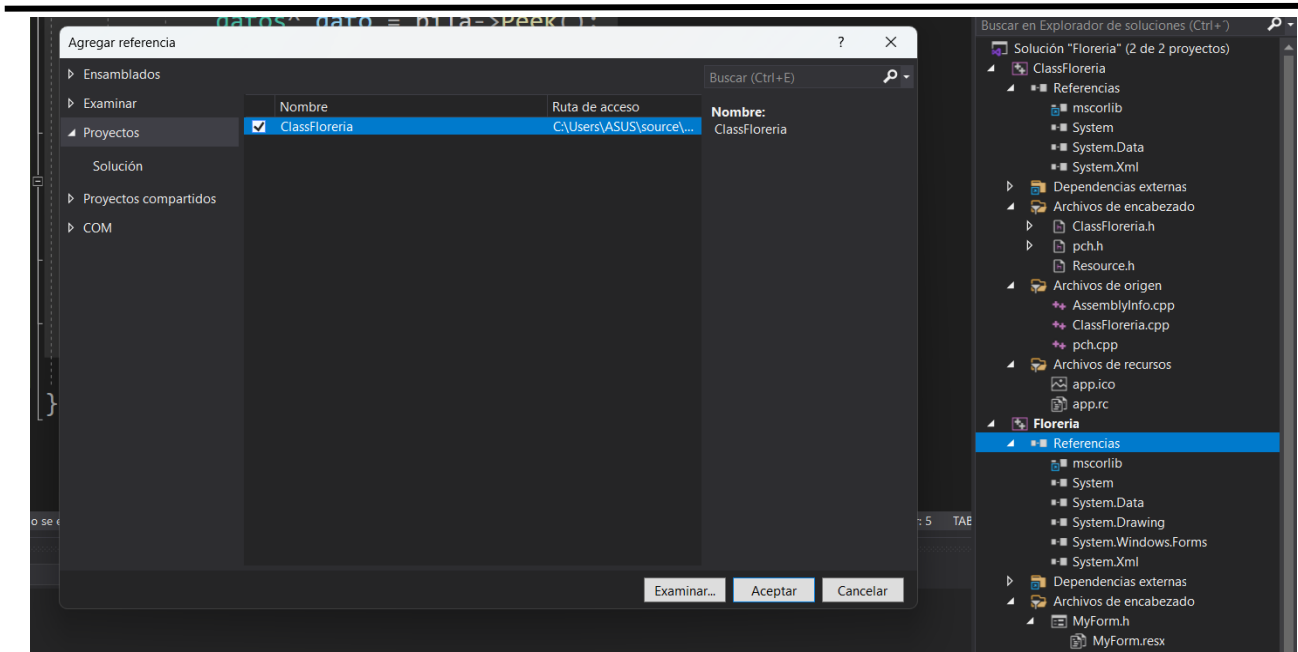
```
{  
private:  
LinkedList<datos^> cola; //defino la decola  
  
public:  
cola_veterinaria() {  
cola = gcnew LinkedList<datos^>();  
}  
void AgregarInicio(datos^ c) {  
cola->AddFirst(c);  
}  
void AgregarFinal(datos^ c) {  
cola->AddLast(c);  
}  
datos^ EliminarInicio() {  
if (cola->Count == 0) return nullptr;  
datos^ dato = cola->First->Value;  
cola->RemoveFirst();  
return dato;  
}  
datos^ EliminarFinal() {  
if (cola->Count == 0) return nullptr;  
datos^ dato = cola->Last->Value;  
cola->RemoveLast();  
return dato;  
}  
  
LinkedList<datos^> ObtenerLista() {  
return cola;  
}
```

```
}  
  
int ObtenerCantidad() {  
  
return cola->Count;  
  
}  
  
};
```

1.- Adicionara en el proyecto ventana (según caso, tendra veterinaria, consultorio o floreria), la referencia a las clases, para ello, ira a referencia, según corresponda:



Hacer clic derecho sobre referencia, adicionar referencia, aparecerá la siguiente ventana, donde seleccionara el proyecto al que quiere referenciar, en este caso sera, el proyecto de clases que ya ha creado, y hara click en aceptar:



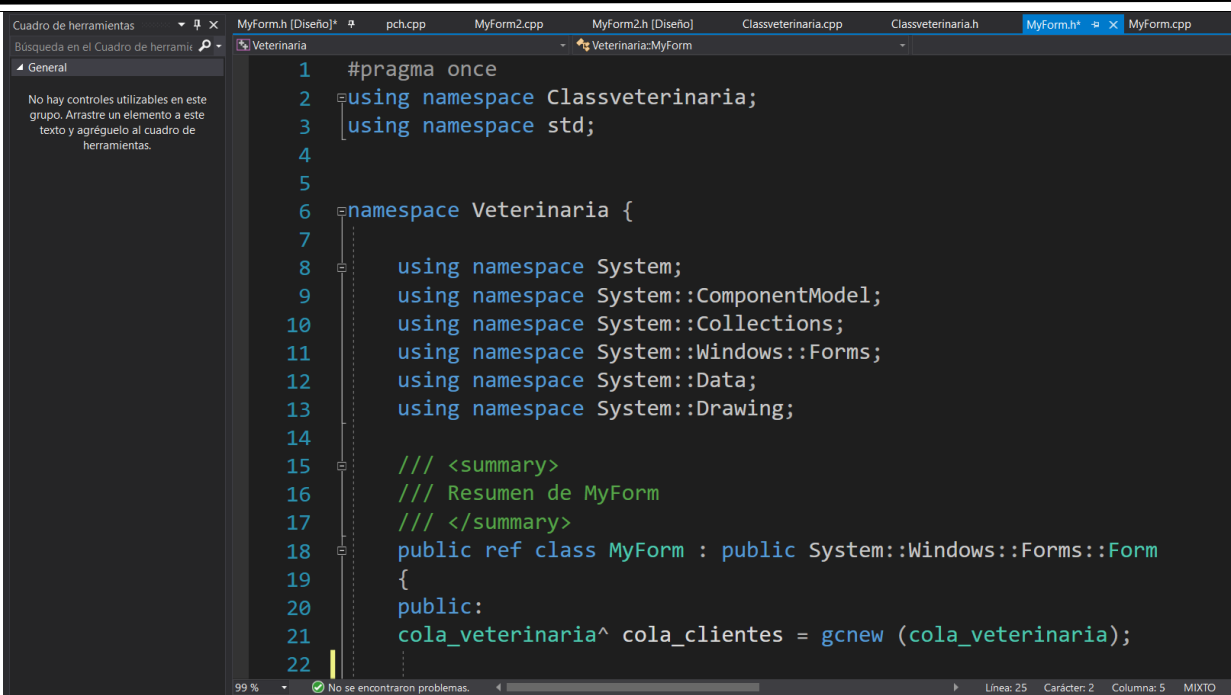
Una vez creada la ventana y la clase cola, habra que crear un objeto cola, y programar los eventos de los botones agregar, eliminar, ya que al hacer click en agregar, se agregara un elemento a la cola, al hacer click en el boton eliminar, se eliminará un elemento de la cola, ademas que se deberá mostrar a la vez, la cola actualizada en el control DataGridView.

1.- En Archivo MyForm.h deberá de poner al inicio (tomar en cuenta el nombre de su namespace):

Para el caso de la Veterinaria
using namespace Classveterinaria;
using namespace std;

2.- Deberá definir la cola con la que trabajará:

Para el caso de la verinaria:
cola_veterinaria^ cola_clientes = gcnew (cola_veterinaria);



3.- Agregar código al botón agregar, hacer doble clic sobre el control y adicionar el código que se encargará de adicionar un elemento a la cola, según sea el caso, primero se crea un objeto datos (con la información necesaria), y ese objeto se inserta a la cola, además se adicionará una fila en el dataGridView. Quedando el código de esta manera:

Para el caso de la veterinaria:

```

String^ texto = numericUpDown1->Text;
int numero = Int32::Parse(texto);
datos^ nueva = gcnew datos(txtDueño->Text, txtContacto->Text, dateTimePicker1->Text, numero);
cola_clientes->AgregarFinal(nueva); //se adiciona en la cola
//utilizando grid, se adiciona una fila con los datos
array<String^>^ fila = gcnew array<String^>{ txtDueño->Text, txtContacto->Text, dateTimePicker1->Text, texto };
dataGridView1->Rows->Add(fila);
dataGridView1->SelectionMode = DataGridViewSelectionMode::FullRowSelect;

```

4.- Agregar código al botón eliminar, hacer doble clic sobre el control y adicionar el código que se encargará de eliminar un elemento a la cola, según sea el caso, primero se define un objeto datos, a ese objeto se le asigna el elemento eliminado de la cola, y se elimina la fila correspondiente del control dataGridView, quedando el código de esta manera:

Para el Caso de la Veterinaria:

```
datos^ primero;
primero = cola_clientes->EliminarInicio();
if (dataGridView1->Rows->Count > 0) {
    //Accede a la primera fila
    DataGridViewRow^ fila = dataGridView1->Rows[0];
    // Guarda los valores en los TextBox
    txtDueño->Text = fila->Cells[0]->Value != nullptr ? fila->Cells[0]->Value->ToString() : "";
    txtContacto->Text = fila->Cells[1]->Value != nullptr ? fila->Cells[1]->Value->ToString() : "";
    numericUpDown1->Text = fila->Cells[3]->Value != nullptr ? fila->Cells[3]->Value->ToString() : "";
    // Elimina la fila del grid
    dataGridView1->Rows->RemoveAt(0);
    // Opcional: seleccionar la nueva primera fila si existe
    if (dataGridView1->Rows->Count > 0) {
        dataGridView1->ClearSelection();
        dataGridView1->Rows[0]->Selected = true;
    }
}
```

5.- Agregar código al botón limpiar, que se encargará de poner en blanco los controles, hacer doble clic sobre el control y adicionar el código que se encargará de limpiar los datos, asignando a los Text de dichos controles valor igual a espacio en blanco.