INGENIERÍA DE SOFTWARE II

Clave: I5900

NRC: 153358

Calendario: 2019-B

Sección: D04

**FASE DE CONSTRUCCIÓN**

*Reporte Final del Proyecto*

INTEGRANTES:

* Guillermo Augusto Alcaraz Domínguez - 217294148
* Jaylene Guadalupe Ceniceros Sevilla - 213573964
* Van Kevin Garcia Garcia - 207546874



# Tabla de contenidos

[**Tabla de contenidos**](#_r8cjr8yr3vdv) **2**

[**Diagramas**](#_qqgjrhr4b3kd) **3**

[Diagramas de casos de uso](#_e60u0nx10orb) 3

[Diagrama de clases](#_hg37cngnbdde) 5

[Diagrama de objetos](#_kn05dld6ur7e) 5

[Diagramas de actividad](#_a0qqftp3trke) 6

[Diagramas de secuencia](#_cm7596q5o8yc) 12

[Diagramas de colaboración](#_3owst02w6ocx) 17

[Diagrama de componentes](#_fdyye8u38rwn) 20

[Diagrama de despliegue (deployment)](#_zffg3n5dc2ef) 20

[Diagrama Entidad-Relación](#_o2fyrewu97h3) 21

[**Código Fuente**](#_g24h1iaxchaj) **22**

[MainForm](#_qk31wj8323zu) 22

[Menu](#_2pn89wi9cdoe) 25

[AgregarMaterial](#_ixj75oeiogah) 27

[AgregarPrestamo](#_366zh2vkq775) 30

[AgregarUsuario](#_u3iujt9zwz) 44

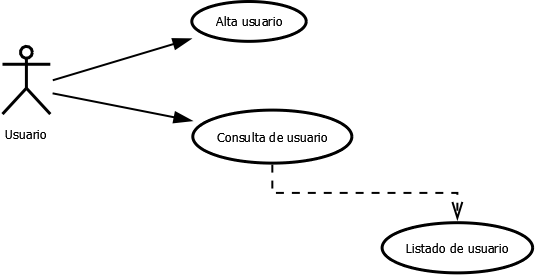
[MostrarMaterial](#_ula5frs3ogws) 47

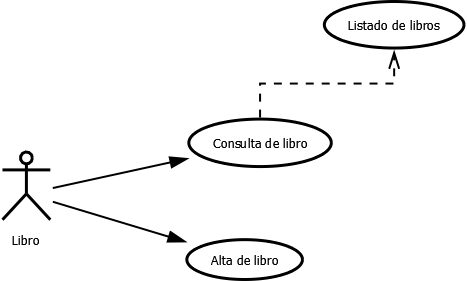
[MostrarUsuario](#_uz6dt25izhux) 50

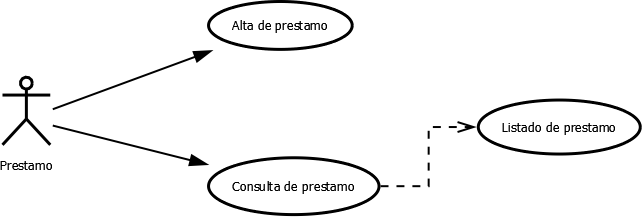
[BuscarPrestamo](#_vcs30116hfm) 52

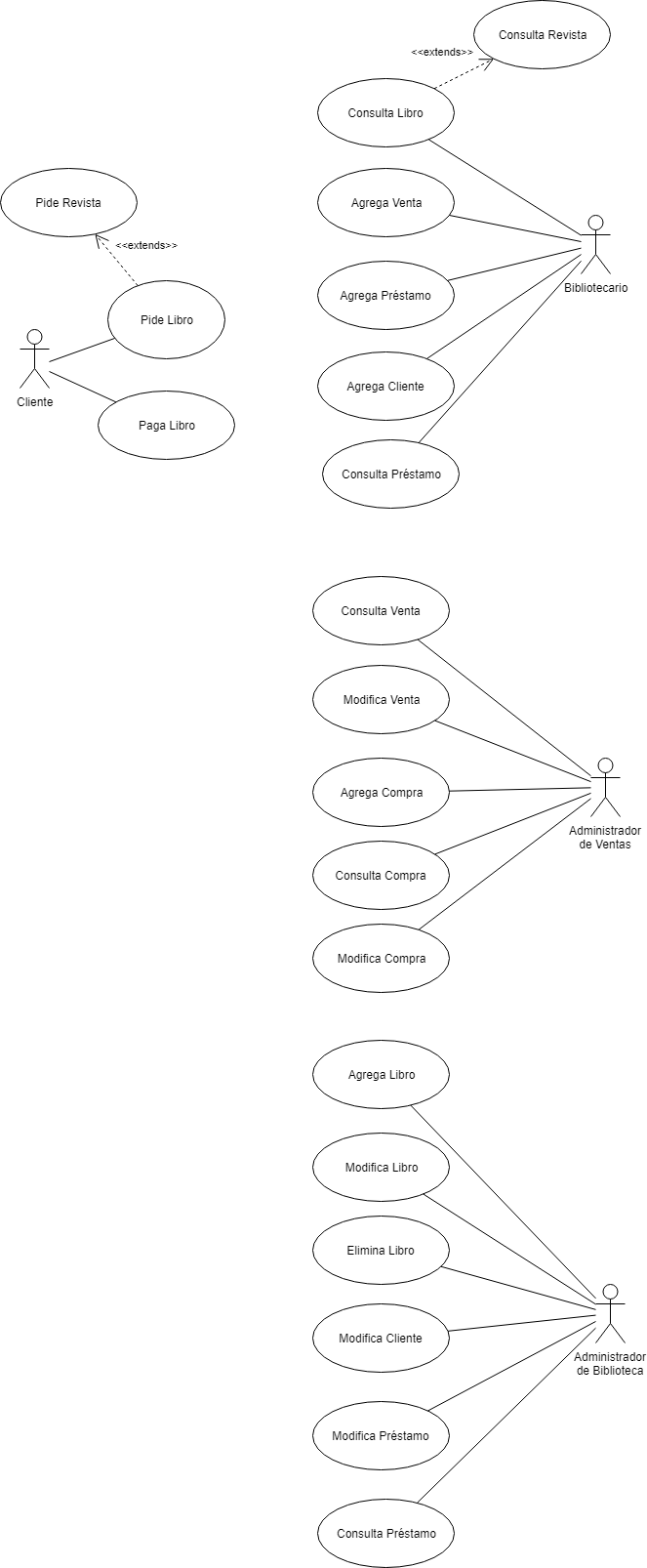
# Diagramas

## Diagramas de casos de uso

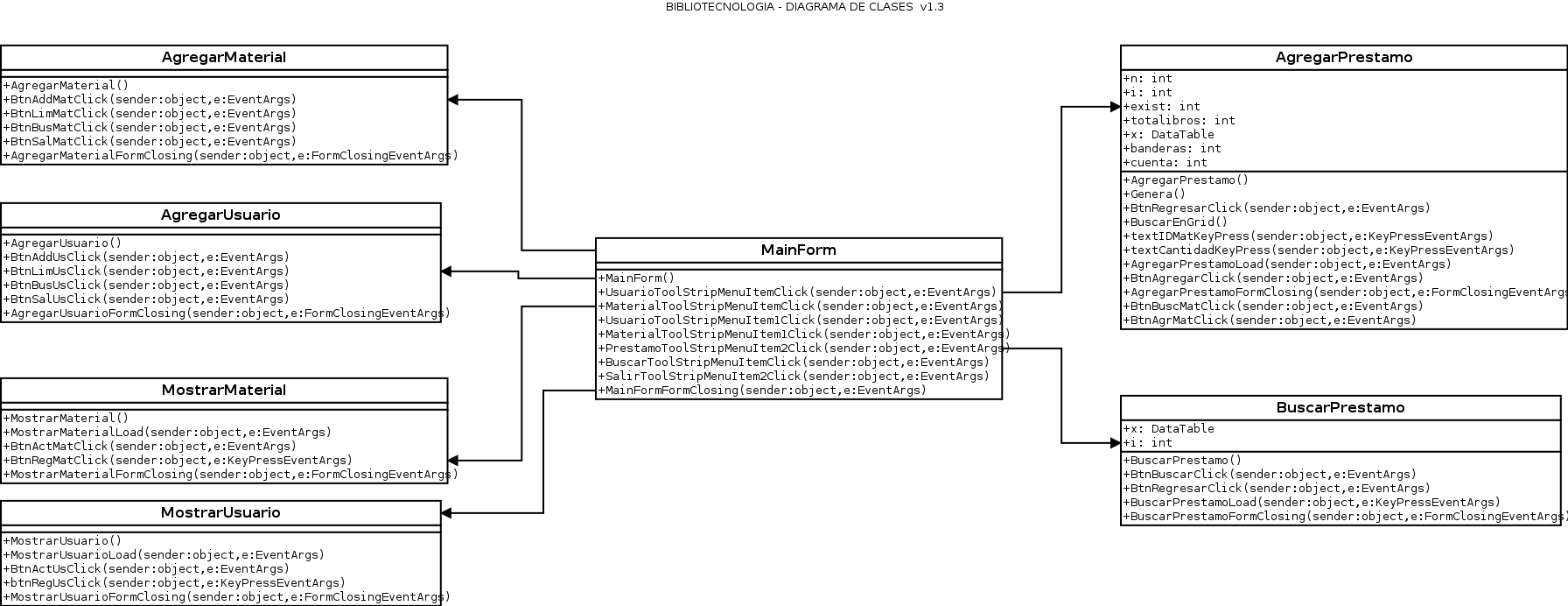






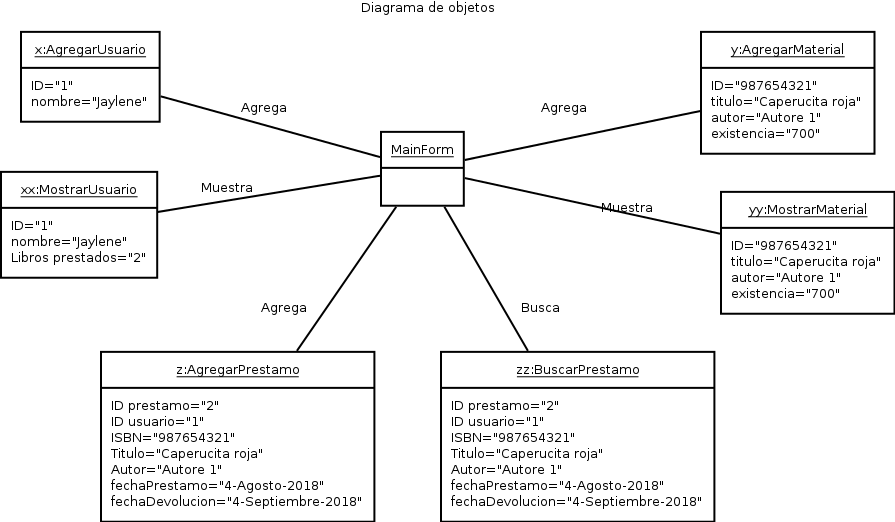
****

## Diagrama de clases



## 

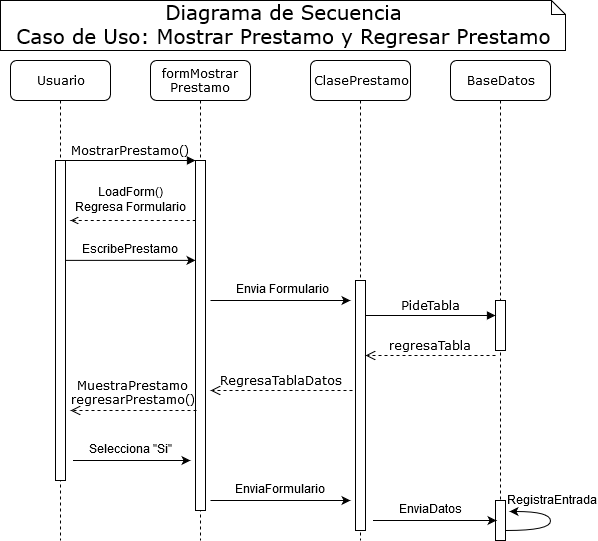
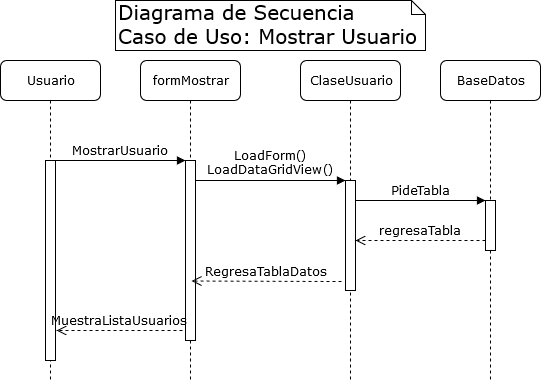
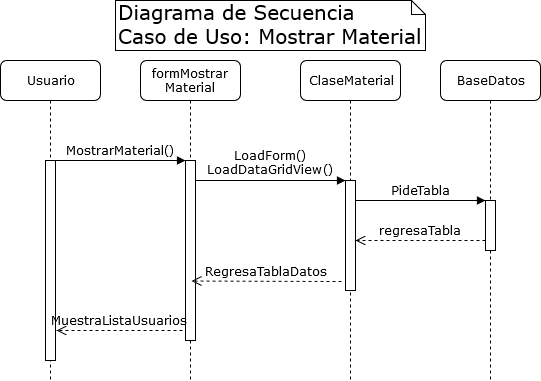
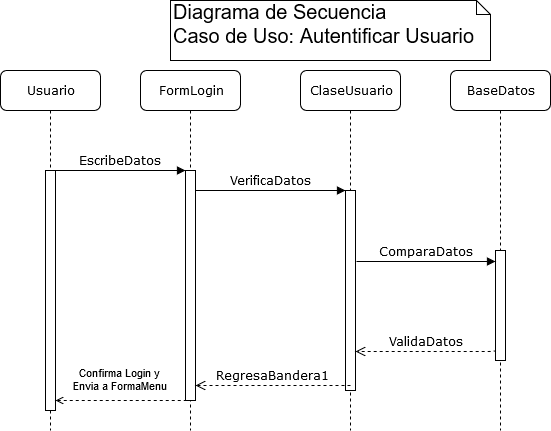
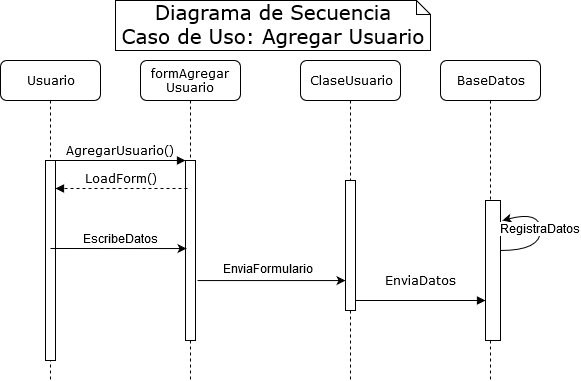
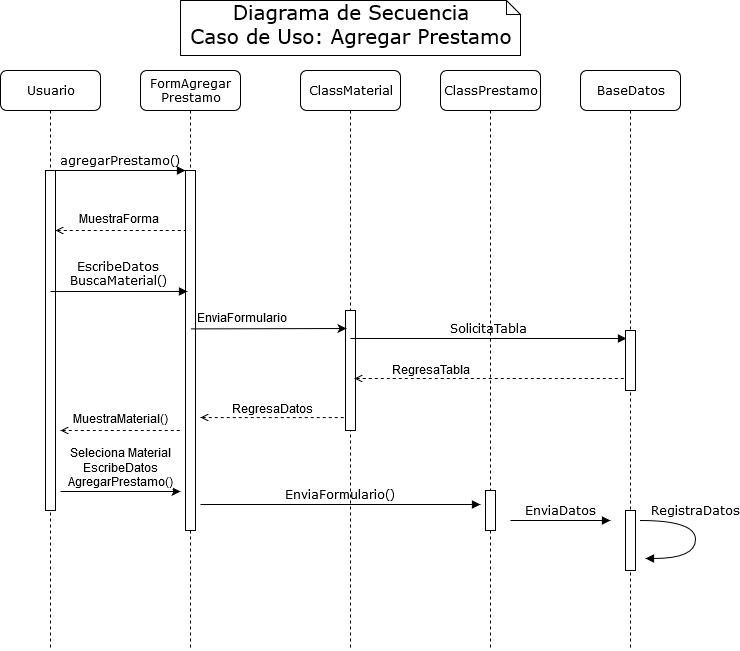
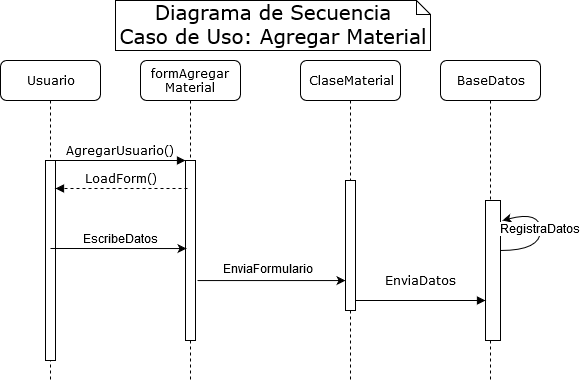
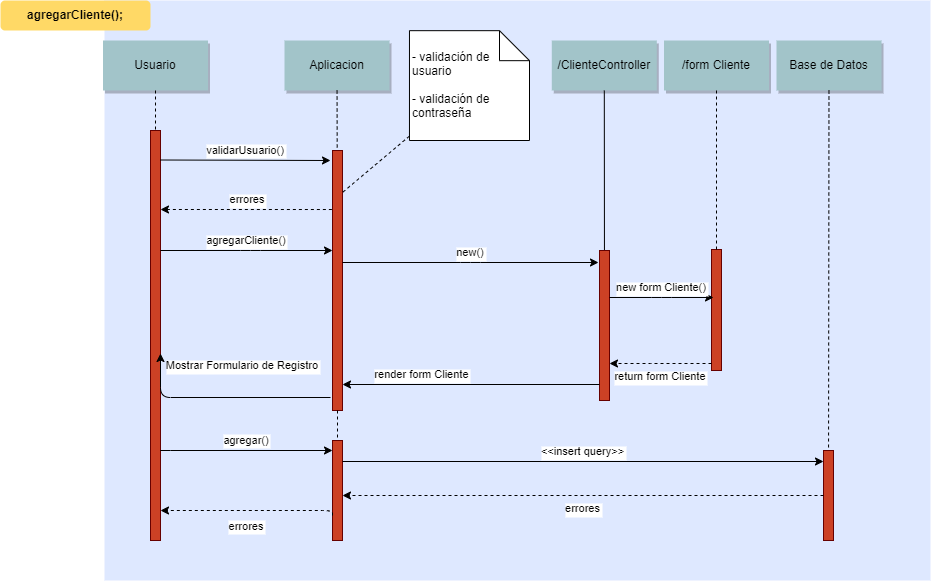
## Diagrama de objetos



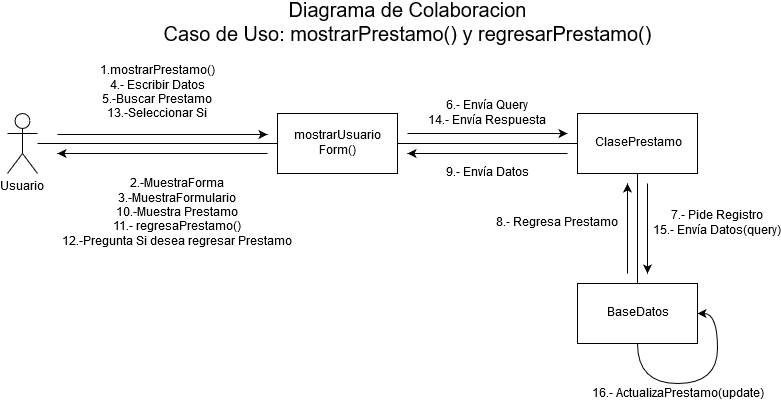
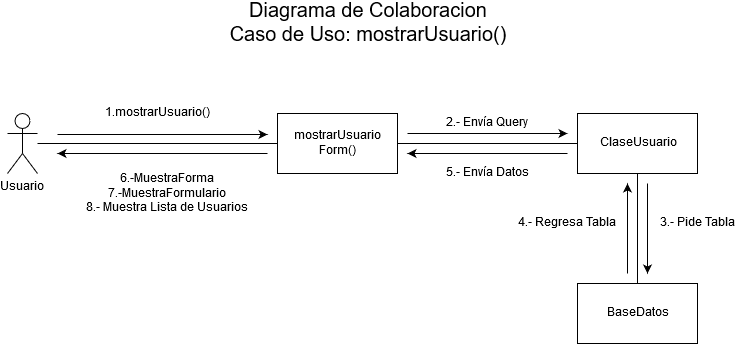
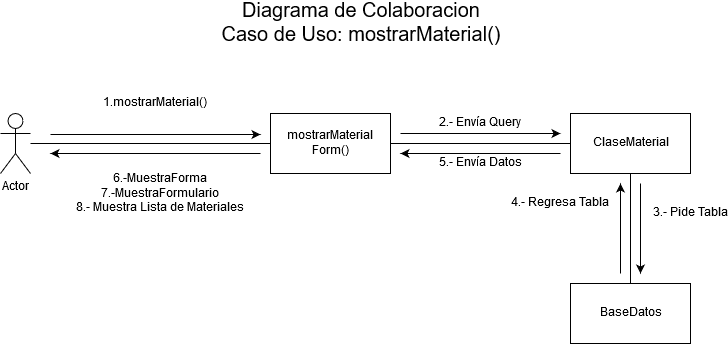
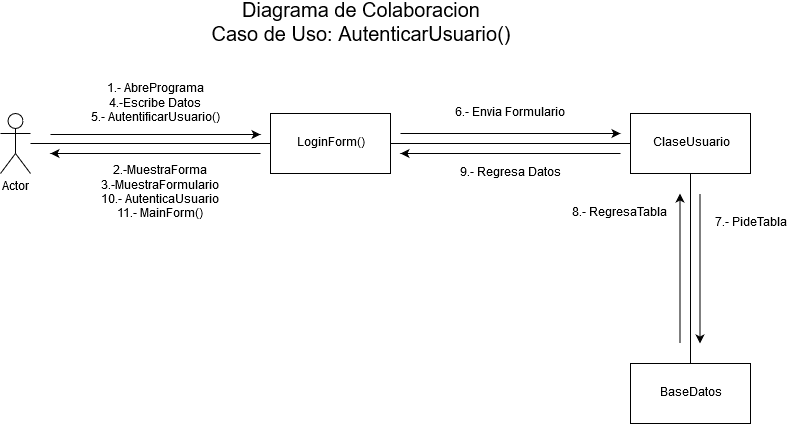
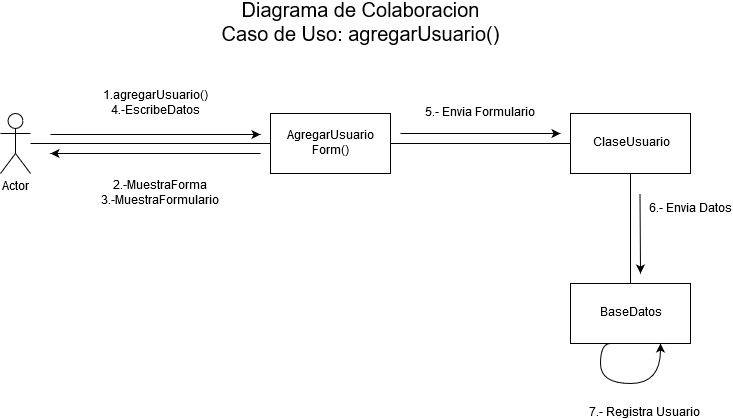
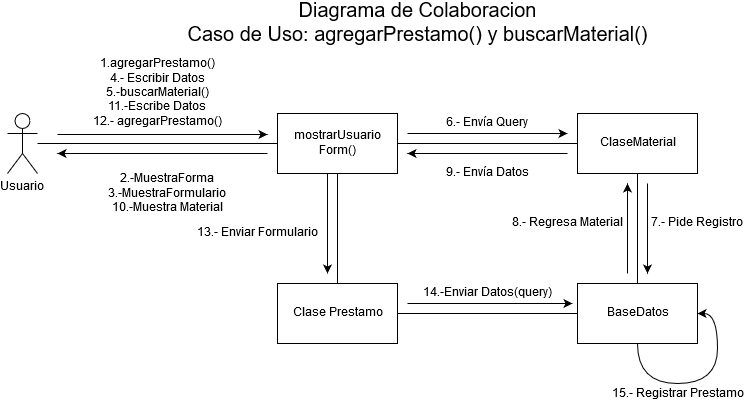
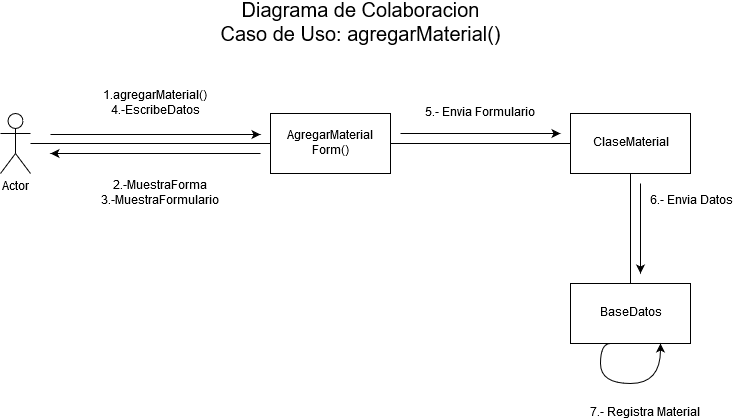
## Diagramas de actividad

## 

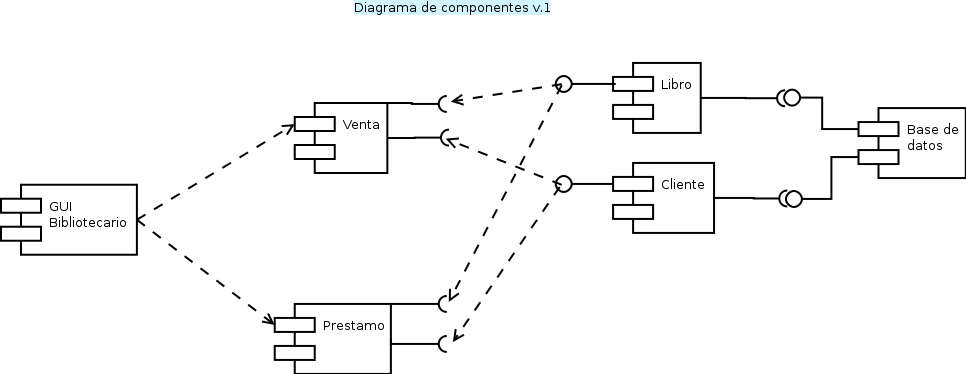
## Diagramas de secuencia



## Diagramas de colaboración



## Diagrama de componentes

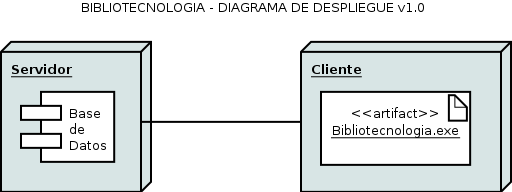


## 

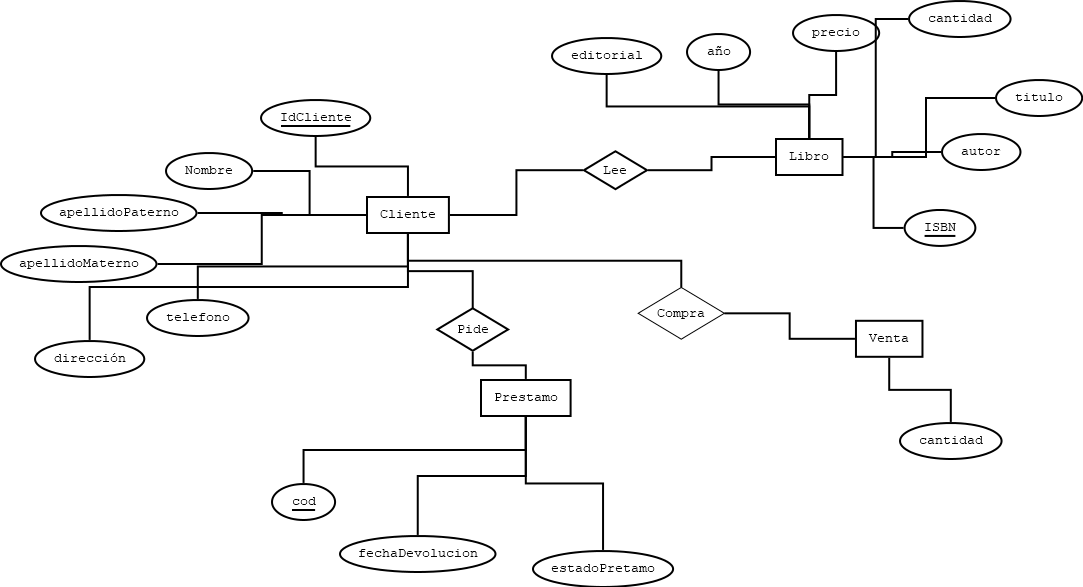
## 

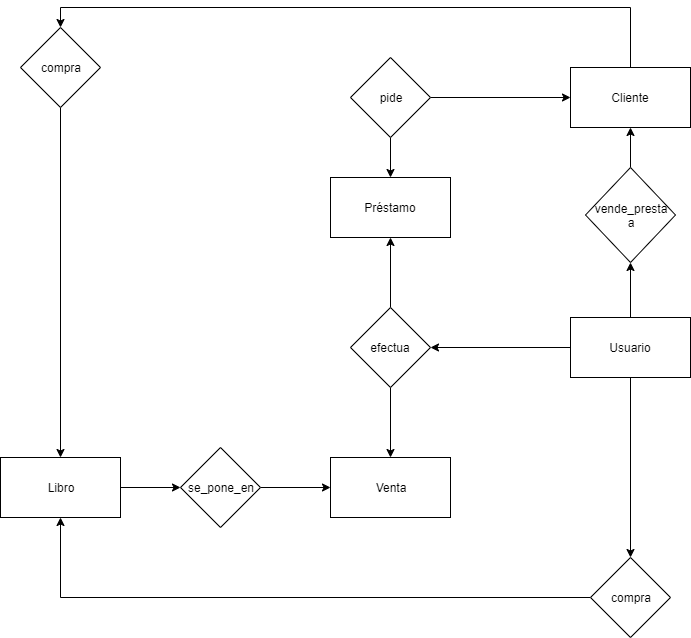
## 

## Diagrama de despliegue (deployment)



## Diagrama Entidad-Relación





# Código Fuente

## MainForm

using System;

using System.Collections.Generic;

using System.ComponentModel;

using System.Data;

using System.Drawing;

using System.Linq;

using System.Text;

using Npgsql;

using System.Windows.Forms;

using System.Diagnostics;

namespace Biblioteca

{

/// <summary>

/// Description of MainForm.

/// </summary>

public partial class MainForm : Form

{

public MainForm()

{

//

// The InitializeComponent() call is required for Windows Forms designer support.

//

InitializeComponent();

//

// TODO: Add constructor code after the InitializeComponent() call.

//

}

/\*private NpgsqlConnection conn;

string connstring = string.Format("Server={0};Port={1};" +

"User Id={2};Password={3};Database={4};",

"localhost", "5432", "Kevin", "", "biblioteca");

//new NpgsqlConnection("localhost;"+ "Database=biblioteca;" +"User ID=Kevin;");

private NpgsqlCommand cmd;

private string sql = null;\*/

void Button1Click(object sender, EventArgs e)

{

try

{

NpgsqlConnection conn = new NpgsqlConnection ("User Id=Kevin;Host=localhost;Database=biblioteca;");

NpgsqlDataAdapter adp = new NpgsqlDataAdapter("select \* from usuario where nombre='"+textBoxNom.Text+"'and contrasena='"+textBoxCon.Text+"'",conn);

DataTable dt = new DataTable();

adp.Fill(dt);

if (dt.Rows[0][0].ToString()=="1")

{

this.Hide();

Form1 z = new Form1();

z.Show();

}else

{

MessageBox.Show("Verifique su usuario y contraseña");

}

/\*//NpgsqlDataAdapter loginConsulta = new NpgsqlDataAdapter("Select nombre, ", conexion); //Esta parte de codigo era antiguo metodo de login y ya no sirve

conn.Open();

sql = @"select \* from usuario(:\_nombre, :\_contrasena)";

cmd = new NpgsqlCommand(sql, conn);

cmd.Parameters.Add("\_nombre",textBoxNom.Text);

cmd.Parameters.Add("\_contrasena",textBoxCon.Text);

int result = (int)cmd.ExecuteScalar();

conn.Close();

if (result == 1)//sesion exitosa

{

this.Hide();

Form1 z = new Form1();

//new Form1(textBoxNom.Text).Show();

z.Show();

}else

{

MessageBox.Show("Usuario o contraseña incorrectos", "No se pudo conectar", MessageBoxButtons.OK, MessageBoxIcon.Asterisk);

return;

}\*/

}

catch (Exception msg)

{

MessageBox.Show(msg.ToString());

}

}

void MainFormFormClosing(object sender, FormClosingEventArgs e)

{

Application.Exit();

}

/\*private void MainFormLoad(object sender, EventArgs e)//Parte del codigo del antiguo metodo de login tampoco sirve, no con esta version de postgresql

{

conn = new NpgsqlConnection(connstring);

}\*/

}

}

## 

## Menu

using System;

using System.Drawing;

using System.Windows.Forms;

using System.Collections.Generic;

using System.ComponentModel;

using System.Data;

using System.Linq;

using System.Text;

using Npgsql;

using System.Diagnostics;

namespace Biblioteca

{

/// <summary>

/// Description of Form1.

/// </summary>

public partial class Form1 : Form

{

public Form1()

{

//

// The InitializeComponent() call is required for Windows Forms designer support.

//

InitializeComponent();

//

// TODO: Add constructor code after the InitializeComponent() call.

//

}

void AgregarToolStripMenuItemClick(object sender, EventArgs e)

{

this.Visible=false;

AgregarUsuario x=new AgregarUsuario();

x.Show();

}

void MostrarToolStripMenuItemClick(object sender, EventArgs e)

{

this.Visible=false;

MostrarUsuario xx=new MostrarUsuario();

xx.Show();

}

void AgregarToolStripMenuItem1Click(object sender, EventArgs e)

{

this.Visible=false;

AgregarMaterial y=new AgregarMaterial();

y.Show();

}

void MostrarToolStripMenuItem1Click(object sender, EventArgs e)

{

this.Visible=false;

MostrarMaterial yy=new MostrarMaterial();

yy.Show();

}

void AgregarToolStripMenuItem2Click(object sender, EventArgs e)

{

this.Visible=false;

AgregarPrestamo z=new AgregarPrestamo();

z.Show();

}

void MostrarToolStripMenuItem2Click(object sender, EventArgs e)

{

this.Visible=false;

BuscarPrestamo zz=new BuscarPrestamo();

zz.Show();

}

void SalirToolStripMenuItemClick(object sender, EventArgs e)

{

Application.Exit();

}

}

}

## 

## AgregarMaterial

using System;

using System.Collections.Generic;

using System.ComponentModel;

using System.Data;

using System.Drawing;

using System.Linq;

using System.Text;

using Npgsql;

using System.Windows.Forms;

using System.Diagnostics;

namespace Biblioteca

{

/// <summary>

/// Description of AgregarMaterial.

/// </summary>

public partial class AgregarMaterial : Form

{

public AgregarMaterial()

{

//

// The InitializeComponent() call is required for Windows Forms designer support.

//

InitializeComponent();

//

// TODO: Add constructor code after the InitializeComponent() call.

//

}

void BtnAddMatClick(object sender, EventArgs e)

{

if(textIDMat.Text=="" || textTitMat.Text=="" || textAutMat.Text=="" || textExiMat.Text=="")

{

MessageBox.Show("Debes introducir todos los datos...");

}

else{

try

{

IDbConnection dbcon = new NpgsqlConnection

("Server=localhost;" +

"Database=biblioteca;" +

"User ID=Kevin;");

dbcon.Open();

IDbCommand dbcmd = dbcon.CreateCommand();

dbcmd.CommandText =

"insert into material values('" +textIDMat.Text + "','" +textTitMat.Text + "','" +textAutMat.Text + "','" +textExiMat.Text + "')";

IDataReader reader = dbcmd.ExecuteReader();

dbcon.Close();

btnLimMat.PerformClick();

MessageBox.Show("Registro Guardado correctamente");

}

catch (Exception msg)

{

MessageBox.Show("Error, base de datos desconectado.....\n\n" + msg.ToString());

}

}

}

void BtnLimMatClick(object sender, EventArgs e)

{

textIDMat.Clear();

textTitMat.Clear();

textAutMat.Clear();

textExiMat.Clear();

}

void BtnBusMatClick(object sender, EventArgs e)

{

try

{

int band = 0;

IDbConnection dbcon = new NpgsqlConnection("Server=localhost;" +"Database=biblioteca;" +"User ID=Kevin;");

dbcon.Open();

IDbCommand dbcmd = dbcon.CreateCommand();

dbcmd.CommandText = "select \* from material where idmaterial='" + textIDMat.Text + "'";

IDataReader reader = dbcmd.ExecuteReader();

if (reader.Read())

{

textTitMat.Text = reader.GetString

(reader.GetOrdinal("titulo"));

textAutMat.Text = reader.GetString

(reader.GetOrdinal("autores"));

textExiMat.Text = Convert.ToString(reader.GetInt32

(reader.GetOrdinal("existencia")));

/\*

txtID.Text = Convert.ToString(reader.GetInt32

(reader.GetOrdinal("id")));\*/

band = 1;

}

dbcon.Close();

if (band == 0)

{

MessageBox.Show("Material no encontrado...");

btnLimMat.PerformClick();

}

}

catch (Exception msg)

{

MessageBox.Show(msg.ToString());

}

}

void BtnSalMatClick(object sender, EventArgs e)

{

this.Visible=false;

MainForm x=new MainForm();

x.Show();

}

void AgregarMaterialFormClosing(object sender, FormClosingEventArgs e)

{

Application.Exit();

}

}

}

## 

## AgregarPrestamo

using System;

using System.Collections.Generic;

using System.ComponentModel;

using System.Data;

using System.Drawing;

using System.Linq;

using System.Text;

using Npgsql;

using System.Windows.Forms;

using System.Diagnostics;

namespace Biblioteca

{

/// <summary>

/// Description of AgregarPrestamo.

/// </summary>

public partial class AgregarPrestamo : Form

{

public AgregarPrestamo()

{

//

// The InitializeComponent() call is required for Windows Forms designer support.

//

InitializeComponent();

//

// TODO: Add constructor code after the InitializeComponent() call.

//

}

int n,i,exist,totalibros;

DataTable x = new DataTable();

public void Genera()

{

try

{

int sigue;

NpgsqlConnection dbcon = new NpgsqlConnection("Server=localhost;" +

"Database=biblioteca;" +

"User ID=Kevin;");

dbcon.Open();

IDbCommand dbcmd = dbcon.CreateCommand();

dbcmd.CommandText =

"select folio from prestamo order by folio desc limit 1";

IDataReader reader = dbcmd.ExecuteReader();

if (reader.Read())

{

sigue =

Convert.ToInt32

(reader.GetInt32(reader.GetOrdinal("folio")) + 1);

lblFolio.Text=Convert.ToString(sigue);

dbcon.Close();

}

else

{

lblFolio.Text = "1";

}

}

catch (Exception msg)

{

MessageBox.Show(msg.ToString());

}

}

void BtnRegresarClick(object sender, EventArgs e)

{

this.Visible=false;

MainForm x=new MainForm();

x.Show();

}

//No existe

/\*void textIDMatKeyPress(object sender, KeyPressEventArgs e)

{

if (e.KeyChar == 13)

{

if (textIDMat.Text == "")

{

MessageBox.Show("DEBES INTRODUCIR EL ID " +

"DEL PRODUCTO....");

textIDMat.Focus();

}

else

{

try

{

IDbConnection dbcon = new

NpgsqlConnection("Server=localhost;" +

"Database=biblioteca;" +

"User ID=Kevin;");

dbcon.Open();

n = Convert.ToInt32(textIDMat.Text);

IDbCommand dbcmd = dbcon.CreateCommand();

dbcmd.CommandText = "select \* from " +

"material where idmaterial=" + n + "";

IDataReader reader = dbcmd.ExecuteReader();

if (reader.Read())

{

textTitulo.Text =

reader.GetString

(reader.GetOrdinal("titulo"));

textAutores.Text =

reader.GetString

(reader.GetOrdinal("autores"));

exist = reader.GetInt32

(reader.GetOrdinal("existencia"));

dbcon.Close();

//textCantidad.Focus();

}

else

{

textIDMat.Clear() ;

textTitulo.Clear();

textAutores.Clear();

MessageBox.Show

("Producto no encontrado");

}

}

catch (Exception msg)

{

MessageBox.Show(msg.ToString());

}

}

}

}\*/

int banderas,cuenta;

public void BuscarEnGrid()

{

cuenta=0;

banderas=0;

do

{

int h= Convert.ToInt16(dtaPrestamo.Rows[cuenta].Cells[0].Value);

if(h==n)

{

banderas=1;

break;

}

cuenta++;

}

while(cuenta<i);

}

void textIDMatKeyPress(object sender, KeyPressEventArgs e)

{

if (e.KeyChar == 13)

{

if (textIDMat.Text == "")

{

MessageBox.Show("Debes introducir el ID del material bibliografico....");

textIDMat.Focus();

}

else

{

try

{

IDbConnection dbcon = new

NpgsqlConnection("Server=localhost;" +

"Database=biblioteca;" +

"User ID=Kevin;");

dbcon.Open();

n = Convert.ToInt32(textIDMat.Text);

IDbCommand dbcmd = dbcon.CreateCommand();

dbcmd.CommandText = "select \* from " +

"material where idmaterial=" + n + "";

IDataReader reader = dbcmd.ExecuteReader();

if (reader.Read())

{

textTitulo.Text =

reader.GetString

(reader.GetOrdinal("titulo"));

textAutores.Text =

reader.GetString

(reader.GetOrdinal("autores"));

exist = reader.GetInt32

(reader.GetOrdinal("existencia"));

dbcon.Close();

}

else

{

textIDMat.Clear() ;

textTitulo.Clear();

textAutores.Clear();

MessageBox.Show("Material bibliografico no encontrado");

}

}

catch (Exception msg)

{

MessageBox.Show(msg.ToString());

}

}

}

}

void textCantidadKeyPress(object sender, KeyPressEventArgs e)

{

if (e.KeyChar == 13)

{

//if(textCantidad.Text=="")

//{

// MessageBox.Show("DEBES INTRODUCIR UNA CANTIDAD....");

// textCantidad.Focus();

//}

//else{

//if (Convert.ToInt16( textCantidad.Text ) <=0 )

// {

// MessageBox.Show("Debes retirar al menos 1 ejemplar....");

//textCantidad.Clear();

//textCantidad.Focus();

// }

//else if (Convert.ToInt16( textCantidad.Text ) >1 )

// {

// MessageBox.Show("Solo puedes retirar 1 ejemplar....");

//textCantidad.Clear();

//textCantidad.Focus();

// }

//if //else

//{

BuscarEnGrid();

if (1 <= exist)

{

if (MessageBox.Show("Deseas agregar el material bibliografico al prestamo?", "", MessageBoxButtons.YesNo) == DialogResult.Yes)

{

if(banderas==1)

{

if((Convert.ToDouble(dtaPrestamo.Rows[cuenta].Cells[2].Value)+1)>exist)

{

MessageBox.Show("Excedes la existencia de ejemplares..");

}

else

{

totalibros=totalibros-1;

dtaPrestamo.Rows[cuenta].Cells[3].Value = Convert.ToInt16(dtaPrestamo.Rows[cuenta].Cells[3].Value)+

1;

totalibros = totalibros

+ Convert.ToInt16

(dtaPrestamo.Rows[cuenta].Cells[3].Value);

}

}

else

{

x.Rows.Add();

dtaPrestamo.Rows[i].Cells[0].Value = n;

dtaPrestamo.Rows[i].Cells[1].Value =

textTitulo.Text;

dtaPrestamo.Rows[i].Cells[2].Value =

textAutores.Text;

dtaPrestamo.Rows[i].Cells[3].Value = 1;

//textCantidad.Text;

totalibros = totalibros

+ Convert.ToInt16

(dtaPrestamo.Rows[i].Cells[3].Value);

dtaPrestamo.Refresh();

//textCantidad.Text = "";

textTitulo.Text = "";

textAutores.Text = "";

textIDMat.Focus();

i++;

}

lblTotal.Text = Convert.ToString(totalibros);

}

textTitulo.Clear();

textAutores.Clear();

textIDMat.Focus();

textIDMat.Clear();

}

else

{

MessageBox.Show("Excedes la existencia de ejemplares, la disponibilidad es de " + exist + " Ejemplares");

//textCantidad.Text = "";

//textCantidad.Focus();

}

}

}

void AgregarPrestamoLoad(object sender, EventArgs e)

{

x.Columns.Add("ID");

x.Columns.Add("Titulo");

x.Columns.Add("Autor");

x.Columns.Add("Cantidad");

dtaPrestamo.DataSource = x.DefaultView;

Genera();

lblFechaPre.Text=DateTime.Now.ToString("dd/MM/yyyy");

lblFechaDev.Text=DateTime.Now.AddDays(30).ToString("dd/MM/yyyy");

textIDMat.Focus();

}

void BtnAgregarClick(object sender, EventArgs e)

{

if (MessageBox.Show("Desea guardar el prestamo?", "Confirmación",

MessageBoxButtons.YesNo,MessageBoxIcon.Question,MessageBoxDefaultButton.Button2) ==

System.Windows.Forms.DialogResult.Yes)

{

if (textUsuario.Text == "")

{

MessageBox.Show("Debes agregar El ID de Usuario");

textUsuario.Focus();

}

else

{

if (i > 0)

{

try

{

int contador = 0, existnew, saldoalumno;

NpgsqlConnection dbcon = new NpgsqlConnection("Server=localhost;" +

"Database=biblioteca;" +

"User ID=Kevin;" );

dbcon.Open();

//Esto es para verificar si el usuario tiene mas de 5 libros prestados

IDbCommand dbcmdd = dbcon.CreateCommand();

dbcmdd.CommandText

= "select librosprestados from usuario where idusuario="+ textUsuario.Text +"";

IDataReader reader3 = dbcmdd.ExecuteReader();

reader3.Read();

saldoalumno = reader3.GetInt32

(reader3.GetOrdinal("librosprestados")) ;

//si los tiene aparecera este mensaje

if (saldoalumno >= 5)

{

MessageBox.Show("Este Usuario ha excedido el numero de prestamos permitidos");

}

//si no los tiene se prosigue con el prestamo

else

{

while (i != contador)

{

n =

Convert.ToInt32

(dtaPrestamo.Rows[contador].Cells[0].Value);

existnew =

Convert.ToInt32(dtaPrestamo.Rows[contador].Cells[3].Value);

string queryUpMaterial =

"update material set existencia=existencia-" + existnew +

" where idmaterial=" + n + "";

NpgsqlCommand cmdUpMaterial =

new NpgsqlCommand(queryUpMaterial, dbcon);

cmdUpMaterial.ExecuteNonQuery();

string queryUpUsuario =

"update usuario set librosprestados=librosprestados+" + existnew +

" where idusuario=" + textUsuario.Text + "";

NpgsqlCommand cmdUpUsuario =

new NpgsqlCommand(queryUpUsuario, dbcon);

cmdUpUsuario.ExecuteNonQuery();

contador++;

}

IDbCommand dbcmdVenta =dbcon.CreateCommand();

dbcmdVenta.CommandText =

"insert into prestamo values("+ Convert.ToInt16(lblFolio.Text)+",'" + lblFechaPre.Text + "','" + lblFechaDev.Text + "','" + textUsuario.Text+"')";

IDataReader readerr = dbcmdVenta.ExecuteReader();

MessageBox.Show("El prestamo se ha realizado...");

textIDMat.Focus();

dbcon.Close();

contador = 0;

while (i != contador)

{

IDbConnection dbconDetalle = new NpgsqlConnection

("Server=localhost;" +

"Database=biblioteca;" +

"User ID=Kevin;");

dbconDetalle.Open();

IDbCommand dbcmdDetalleVenta =dbconDetalle.CreateCommand();

dbcmdDetalleVenta.CommandText =

"insert into detalleprestamo values("+ Convert.ToInt16(lblFolio.Text)+"," + Convert.ToInt32(dtaPrestamo.Rows[contador].Cells[0].Value) + ")";

IDataReader reader2 = dbcmdDetalleVenta.ExecuteReader();

contador++;

}

Genera();

i = 0;

lblTotal.Text = "";

totalibros = 0;

x.Rows.Clear();

}

}

catch (Exception msg)

{

MessageBox.Show(msg.ToString());

}

}

else

{

MessageBox.Show("Primero debes registrar material bibliografico.");

}

}

}

}

void AgregarPrestamoFormClosing(object sender, FormClosingEventArgs e)

{

Application.Exit();

}

void BtnBuscMatClick(object sender, EventArgs e)

{

if (textIDMat.Text == "")

{

MessageBox.Show("Debes introducir el ID del material bibliografico....");

textIDMat.Focus();

}

else

{

try

{

IDbConnection dbcon = new NpgsqlConnection("Server=localhost;" +"Database=biblioteca;" +"User ID=Kevin;");

dbcon.Open();

n = Convert.ToInt32(textIDMat.Text);

IDbCommand dbcmd = dbcon.CreateCommand();

dbcmd.CommandText = "select \* from " +"material where idmaterial=" + n + "";

IDataReader reader = dbcmd.ExecuteReader();

if (reader.Read())

{

textTitulo.Text =

reader.GetString

(reader.GetOrdinal("titulo"));

textAutores.Text =

reader.GetString

(reader.GetOrdinal("autores"));

exist = reader.GetInt32

(reader.GetOrdinal("existencia"));

dbcon.Close();

}

else

{

textIDMat.Clear() ;

textTitulo.Clear();

textAutores.Clear();

MessageBox.Show("Material bibliografico no encontrado");

}

}

catch (Exception msg)

{

MessageBox.Show(msg.ToString());

}

}

}

void BtnAgrMatClick(object sender, EventArgs e)

{

BuscarEnGrid();

if (1 <= exist)

{

if (MessageBox.Show("Deseas agregar el material bibliografico al prestamo?", "", MessageBoxButtons.YesNo) == DialogResult.Yes)

{

if (totalibros > 5)

{

MessageBox.Show("Solo puedes agregar 5 libros");

}

else

{

if(banderas==1)

{

if((Convert.ToDouble(dtaPrestamo.Rows[cuenta].Cells[2].Value)+1)>exist)

{

MessageBox.Show("Excedes la existencia de ejemplares..");

}

else

{

totalibros=totalibros-1;

dtaPrestamo.Rows[cuenta].Cells[3].Value = Convert.ToInt16(dtaPrestamo.Rows[cuenta].Cells[3].Value)+

1;

totalibros = totalibros

+ Convert.ToInt16

(dtaPrestamo.Rows[cuenta].Cells[3].Value);

}

}

else

{

x.Rows.Add();

dtaPrestamo.Rows[i].Cells[0].Value = n;

dtaPrestamo.Rows[i].Cells[1].Value =

textTitulo.Text;

dtaPrestamo.Rows[i].Cells[2].Value =

textAutores.Text;

dtaPrestamo.Rows[i].Cells[3].Value = 1;

totalibros = totalibros

+ Convert.ToInt16

(dtaPrestamo.Rows[i].Cells[3].Value);

dtaPrestamo.Refresh();

textTitulo.Text = "";

textAutores.Text = "";

textIDMat.Text = "";

textIDMat.Focus();

i++;

}

}

}

lblTotal.Text = Convert.ToString(totalibros);

textTitulo.Clear();

textAutores.Clear();

textIDMat.Focus();

}

else

{

MessageBox.Show("Excedes la existencia, la disponibilidad es de " + exist + " Ejemplares");

}

}

}

}

## 

## AgregarUsuario

using System;

using System.Collections.Generic;

using System.ComponentModel;

using System.Data;

using System.Drawing;

using System.Linq;

using System.Text;

using Npgsql;

using System.Windows.Forms;

using System.Diagnostics;

namespace Biblioteca

{

/// <summary>

/// Description of AgregarUsuario.

/// </summary>

public partial class AgregarUsuario : Form

{

public AgregarUsuario()

{

//

// The InitializeComponent() call is required for Windows Forms designer support.

//

InitializeComponent();

//

// TODO: Add constructor code after the InitializeComponent() call.

//

}

void BtnAddUsClick(object sender, EventArgs e)

{

if(textNomUs.Text=="" || textIdUs.Text=="" || textContrUs.Text=="")

{

MessageBox.Show("Debes introducir todos los Datos...");

}

else{

try

{

IDbConnection dbcon = new NpgsqlConnection("Server=localhost;" +"Database=biblioteca;" +"User ID=Kevin;");

dbcon.Open();

IDbCommand dbcmd = dbcon.CreateCommand();

dbcmd.CommandText =

"insert into usuario values('" +textIdUs.Text + "','" +textNomUs.Text + "','" +textContrUs.Text + "')";

IDataReader reader = dbcmd.ExecuteReader();

dbcon.Close();

btnLimUs.PerformClick();

MessageBox.Show("Registro Guardado correctamente");

}

catch (Exception msg)

{

MessageBox.Show("Error, base de datos desconectado.....\n\n" + msg.ToString());

}

}

}

void BtnLimUsClick(object sender, EventArgs e)

{

textIdUs.Clear();

textNomUs.Clear();

}

void BtnBusUsClick(object sender, EventArgs e)

{

try

{

int band = 0;

IDbConnection dbcon = new NpgsqlConnection("Server=localhost;" +"Database=biblioteca;" +"User ID=Kevin;");

dbcon.Open();

IDbCommand dbcmd = dbcon.CreateCommand();

dbcmd.CommandText = "select \* from usuario where idusuario='" + textIdUs.Text + "'";

IDataReader reader = dbcmd.ExecuteReader();

if (reader.Read())

{

textNomUs.Text = reader.GetString

(reader.GetOrdinal("nombre"));

textNomUs.Text = reader.GetString

(reader.GetOrdinal("contrasena"));

/\*

txtID.Text = Convert.ToString(reader.GetInt32

(reader.GetOrdinal("id")));\*/

band = 1;

}

dbcon.Close();

if (band == 0)

{

MessageBox.Show("Usuario no encontrado...");

textIdUs.Enabled=true;

btnLimUs.PerformClick();

}

}

catch (Exception msg)

{

MessageBox.Show(msg.ToString());

}

}

void BtnSalUsClick(object sender, EventArgs e)

{

this.Visible=false;

MainForm x=new MainForm();

x.Show();

}

void AgregarUsuarioFormClosing(object sender, FormClosingEventArgs e)

{

Application.Exit();

}

}

}

## 

## MostrarMaterial

using System;

using System.Collections.Generic;

using System.ComponentModel;

using System.Data;

using System.Drawing;

using System.Linq;

using System.Text;

using Npgsql;

using System.Windows.Forms;

using System.Diagnostics;

namespace Biblioteca

{

/// <summary>

/// Description of MostrarMaterial.

/// </summary>

public partial class MostrarMaterial : Form

{

public MostrarMaterial()

{

//

// The InitializeComponent() call is required for Windows Forms designer support.

//

InitializeComponent();

//

// TODO: Add constructor code after the InitializeComponent() call.

//

}

void MostrarMaterialLoad(object sender, EventArgs e)

{

DataTable memoria = new DataTable();

try

{

NpgsqlConnection conexion = new NpgsqlConnection("Server=localhost;"+ "Database=biblioteca;" +"User ID=Kevin;");

NpgsqlDataAdapter datosConsulta = new NpgsqlDataAdapter("select idmaterial,titulo,autores,existencia from material", conexion);

datosConsulta.Fill(memoria);

dgvMaterial.DataSource = memoria.DefaultView;

}

catch (Exception msg)

{

MessageBox.Show(msg.ToString());

}

}

void BtnActMatClick(object sender, EventArgs e)

{

DataTable memoria = new DataTable();

try

{

NpgsqlConnection conexion = new NpgsqlConnection("Server=localhost;"+ "Database=biblioteca;" +"User ID=Kevin;");

NpgsqlDataAdapter datosConsulta = new NpgsqlDataAdapter("select idmaterial,titulo,autores,existencia from material", conexion);

datosConsulta.Fill(memoria);

dgvMaterial.DataSource = memoria.DefaultView;

}

catch (Exception msg)

{

MessageBox.Show(msg.ToString());

}

}

void BtnRegMatClick(object sender, EventArgs e)

{

this.Visible=false;

MainForm x=new MainForm();

x.Show();

}

void MostrarMaterialFormClosing(object sender, FormClosingEventArgs e)

{

Application.Exit();

}

}

}

## 

## MostrarUsuario

using System;

using System.Collections.Generic;

using System.ComponentModel;

using System.Data;

using System.Drawing;

using System.Linq;

using System.Text;

using Npgsql;

using System.Windows.Forms;

using System.Diagnostics;

namespace Biblioteca

{

/// <summary>

/// Description of MostrarUsuario.

/// </summary>

public partial class MostrarUsuario : Form

{

public MostrarUsuario()

{

//

// The InitializeComponent() call is required for Windows Forms designer support.

//

InitializeComponent();

//

// TODO: Add constructor code after the InitializeComponent() call.

//

}

void MostrarUsuarioLoad(object sender, EventArgs e)

{

DataTable memoria = new DataTable();

try

{

NpgsqlConnection conexion = new NpgsqlConnection("Server=localhost;"+ "Database=biblioteca;" +"User ID=Kevin;");

NpgsqlDataAdapter datosConsulta = new NpgsqlDataAdapter("select idusuario,nombre,librosprestados,contrasena from usuario", conexion);

datosConsulta.Fill(memoria);

dgvUsuario.DataSource = memoria.DefaultView;

}

catch (Exception msg)

{

MessageBox.Show(msg.ToString());

}

}

void BtnActUsClick(object sender, EventArgs e)

{

DataTable memoria = new DataTable();

try

{

NpgsqlConnection conexion = new NpgsqlConnection("Server=localhost;"

+ "Database=biblioteca;" +

"User ID=Kevin;");

NpgsqlDataAdapter datosConsulta = new NpgsqlDataAdapter("select idusuario,nombre,librosprestados,contrasena from usuario", conexion);

datosConsulta.Fill(memoria);

dgvUsuario.DataSource = memoria.DefaultView;

}

catch (Exception msg)

{

MessageBox.Show(msg.ToString());

}

}

void btnRegUsClick(object sender, EventArgs e)

{

this.Visible=false;

MainForm x=new MainForm();

x.Show();

}

void MostrarUsuarioFormClosing(object sender, FormClosingEventArgs e)

{

Application.Exit();

}

}

}

## 

## BuscarPrestamo

using System;

using System.Collections.Generic;

using System.ComponentModel;

using System.Data;

using System.Drawing;

using System.Linq;

using System.Text;

using Npgsql;

using System.Windows.Forms;

using System.Diagnostics;

namespace Biblioteca

{

/// <summary>

/// Description of BuscarPrestamo.

/// </summary>

public partial class BuscarPrestamo : Form

{

public BuscarPrestamo()

{

//

// The InitializeComponent() call is required for Windows Forms designer support.

//

InitializeComponent();

//

// TODO: Add constructor code after the InitializeComponent() call.

//

}

DataTable x = new DataTable();

int i;

void BtnBuscarClick(object sender, EventArgs e)

{

if(textIDPrestamo.Text=="")

{

MessageBox.Show("Debes introducir un folio a buscar....");

textIDPrestamo.Focus();

}

else

{

try

{

x.Rows.Clear();

int band=0;

int totalibros=0;

bool devuelto;

IDbConnection dbcon = new NpgsqlConnection

("Server=localhost;"+"Database=biblioteca;"+"User ID=Kevin;");

dbcon.Open();

IDbCommand dbcmd = dbcon.CreateCommand();

dbcmd.CommandText = "select \* from prestamo where folio=" + textIDPrestamo.Text + "";

IDataReader reader = dbcmd.ExecuteReader();

if (reader.Read())

{

lblFolio.Text = textIDPrestamo.Text;

//lblIDUsuario.Text = Convert.ToString(reader.GetInt32

//(reader.GetOrdinal("idusuario")));

lblFechaPre.Text = Convert.ToString(reader.GetDateTime

(reader.GetOrdinal("fechaprestamo")));

lblFechaDev.Text = Convert.ToString(reader.GetDateTime

(reader.GetOrdinal("fechadevolucion")));

lblStatus.Text = Convert.ToString(reader.GetBoolean

(reader.GetOrdinal("status")));

band = 1;

}

reader.Close();

if (band == 1)

{

IDbCommand comandoDetallePrestamo = dbcon.CreateCommand();

comandoDetallePrestamo.CommandText = "select \* from detalleprestamo where foliopre=" +Convert.ToInt16(lblFolio.Text) + "";

IDataReader readerDetallePrestamo = comandoDetallePrestamo.ExecuteReader();

i=0;

while(readerDetallePrestamo.Read())

{

x.Rows.Add();

dtaBuscPrestamo.Rows[i].Cells[0].Value =readerDetallePrestamo.GetInt32(readerDetallePrestamo.GetOrdinal("foliopre"));

dtaBuscPrestamo.Rows[i].Cells[1].Value =readerDetallePrestamo.GetInt32(readerDetallePrestamo.GetOrdinal("idmat"));

i++;

}

readerDetallePrestamo.Close();

int contador=0;

IDbCommand comandoConsultaProducto = dbcon.CreateCommand();

if (comandoConsultaProducto == null)

return;

while(contador != i)

{

comandoConsultaProducto.CommandText = "select titulo, autores from material where idmaterial=" +Convert.ToInt32(dtaBuscPrestamo.Rows[contador].Cells[1].Value) + "";

IDataReader readerProducto = comandoConsultaProducto.ExecuteReader();

if(readerProducto.Read())

{

dtaBuscPrestamo.Rows[contador].Cells[2].Value =readerProducto.GetString(readerProducto.GetOrdinal("titulo"));

dtaBuscPrestamo.Rows[contador].Cells[3].Value =readerProducto.GetString(readerProducto.GetOrdinal("autores"));

dtaBuscPrestamo.Rows[contador].Cells[4].Value =Convert.ToString(1);

totalibros = totalibros + Convert.ToInt16(dtaBuscPrestamo.Rows[contador].Cells[4].Value);

contador++;

readerProducto.Close();

}

}

lblTotal.Text = Convert.ToString(totalibros);

}

//Esto es para verificar si el prestamo ha sido devuelto

IDbCommand dbcmddd = dbcon.CreateCommand();

dbcmddd.CommandText = "select status from prestamo where folio="+ textIDPrestamo.Text +"";

IDataReader reader4 = dbcmddd.ExecuteReader();

reader4.Read();

devuelto = reader4.GetBoolean

(reader4.GetOrdinal("status")) ;

//si los tiene aparecera este mensaje

if (devuelto == false)

{

if (MessageBox.Show("El prestamo sera devuelto?", "Confirmación",

MessageBoxButtons.YesNo,MessageBoxIcon.Question,MessageBoxDefaultButton.Button2) ==

System.Windows.Forms.DialogResult.Yes)

{

NpgsqlConnection dbconn = new NpgsqlConnection("Server=localhost;" +

"Database=biblioteca;" +

"User ID=Kevin;" );

dbconn.Open();

IDbCommand dbcmdddd = dbconn.CreateCommand();

dbcmdddd.CommandText = "update prestamo set status = true where folio="+ lblFolio.Text +"";

IDataReader reader5 = dbcmdddd.ExecuteReader();

reader5.Read();

int n, contador=0, existnew;

while (i != contador)

{

n =

Convert.ToInt32

(dtaBuscPrestamo.Rows[contador].Cells[1].Value);

existnew =

Convert.ToInt32(dtaBuscPrestamo.Rows[contador].Cells[4].Value);

string queryUppMaterial =

"update material set existencia=existencia+" + existnew +

" where idmaterial=" + n + "";

NpgsqlCommand cmdUppMaterial =

new NpgsqlCommand(queryUppMaterial, dbconn);

cmdUppMaterial.ExecuteNonQuery();

string queryUppUsuario =

"update usuario set librosprestados=librosprestados-" + existnew +

" where idusuario=" + lblIDUsuario.Text + "";

NpgsqlCommand cmdUppUsuario =

new NpgsqlCommand(queryUppUsuario, dbconn);

cmdUppUsuario.ExecuteNonQuery();

contador++;

}

IDbCommand dbcmd90 = dbcon.CreateCommand();

dbcmd90.CommandText = "select status from prestamo where folio=" + lblFolio.Text + "";

IDataReader reader90 = dbcmd90.ExecuteReader();

reader90.Read();

lblStatus.Text = Convert.ToString(reader.GetBoolean

(reader.GetOrdinal("status")));

MessageBox.Show("Devolucion Exitosa.");

}

}

//si ya esta devuelto aparece un mensaje

else

{

MessageBox.Show("Este prestamo ya ha sido devuelto");

}

dbcon.Close();

if (band == 0)

{

MessageBox.Show("Prestamo No Encontrado...");

}

}

catch (Exception msg)

{

MessageBox.Show(msg.ToString());

MessageBox.Show("Fallo la conversion");

}

textIDPrestamo.Clear();

textIDPrestamo.Focus();

}

}

void BtnRegresarClick(object sender, EventArgs e)

{

this.Visible=false;

MainForm x=new MainForm();

x.Show();

}

void BuscarPrestamoLoad(object sender, EventArgs e)

{

x.Columns.Add("Folio");

x.Columns.Add("ID Material");

x.Columns.Add("Titulo");

x.Columns.Add("Autor");

x.Columns.Add("Cantidad");

dtaBuscPrestamo.DataSource = x.DefaultView;

}

void BuscarPrestamoFormClosing(object sender, FormClosingEventArgs e)

{

Application.Exit();

}

}

}