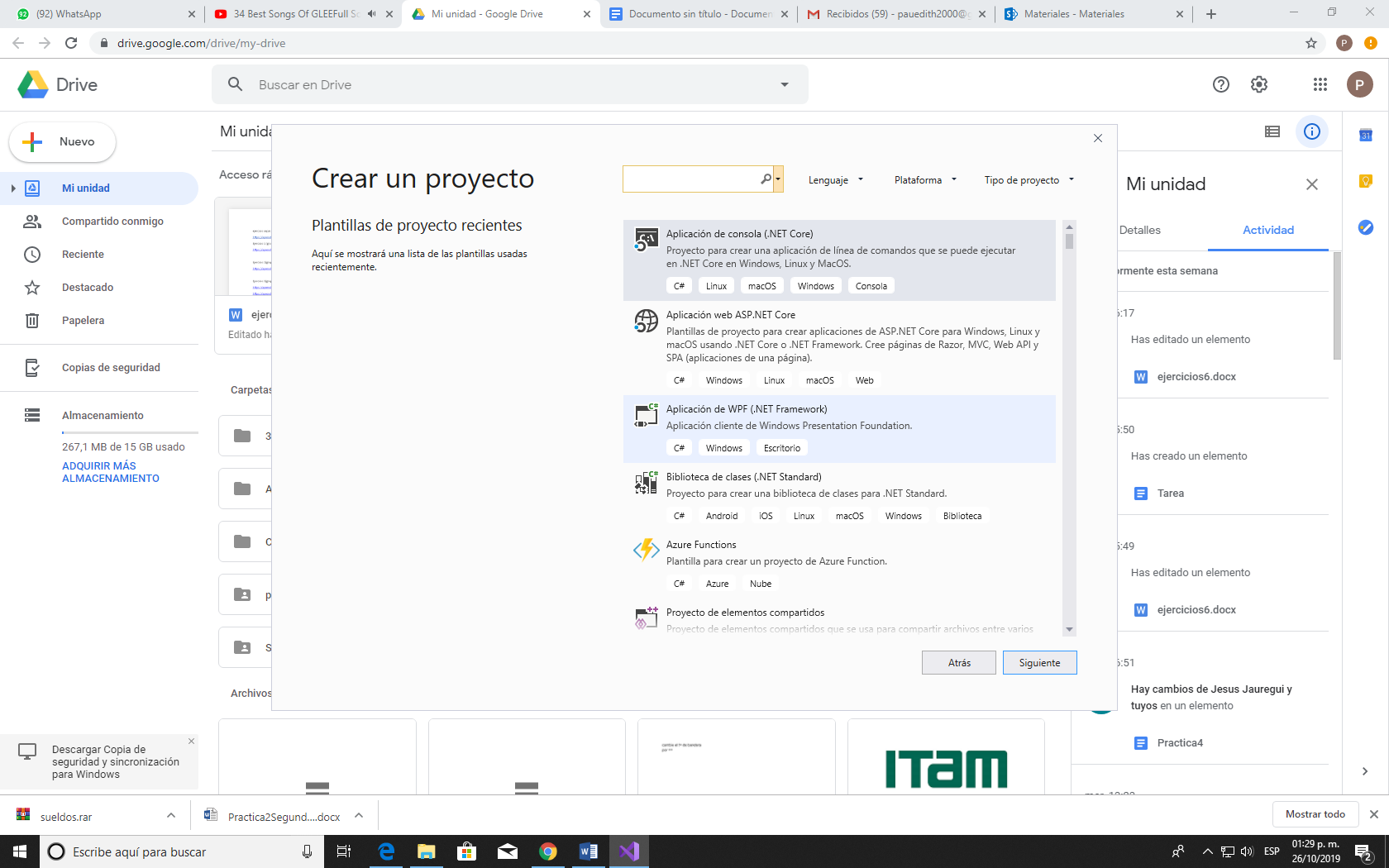
**WPF**

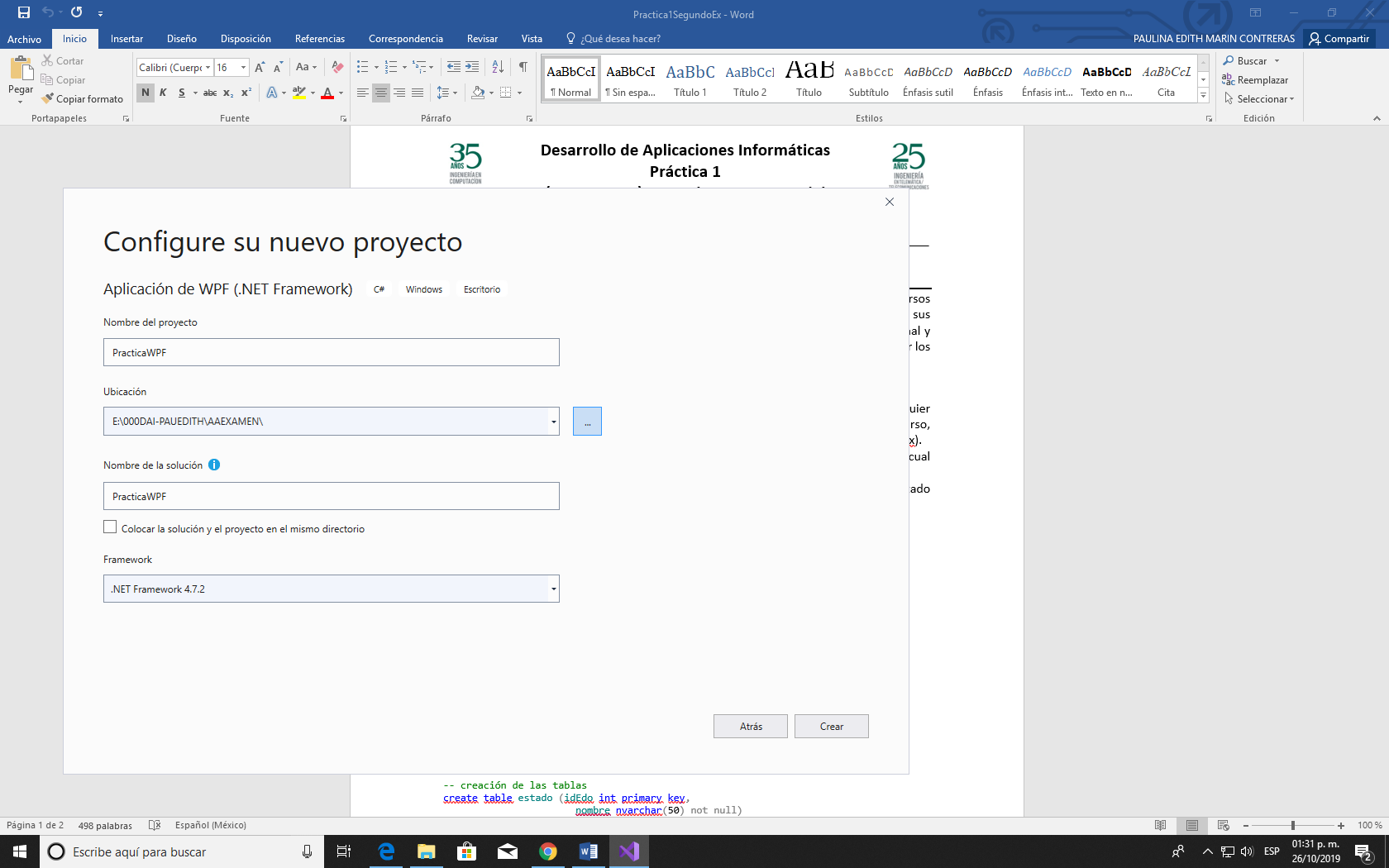
dudas?? puedes checar practica wpf y sitiopagina2

**Crear proyecto**

Usar aplicación de wpf (net framework)

****

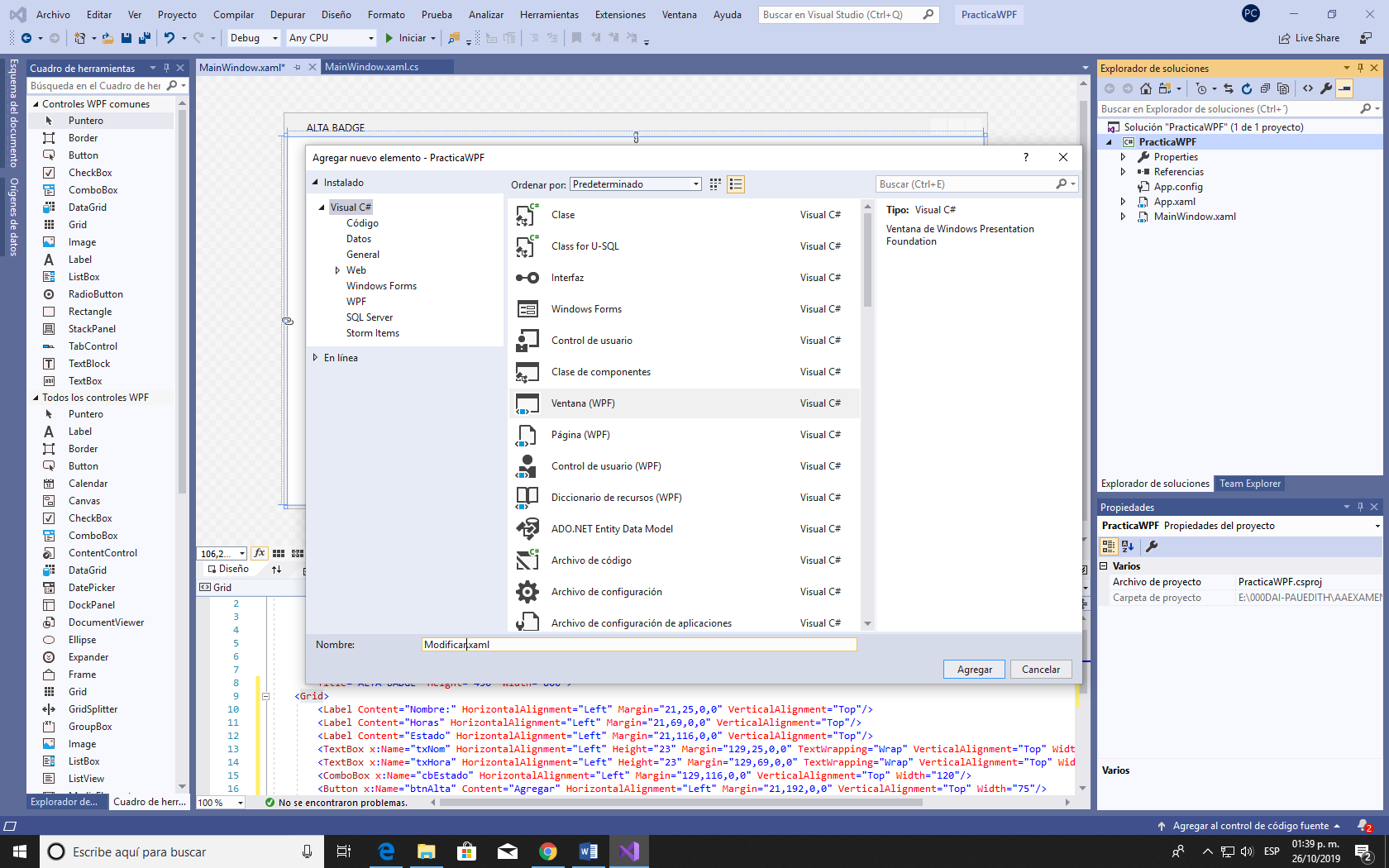
PONER NOMBRE Y SELECCIONAR DONDE SE GUARDARÁ



Para agregar otra ventana

Seleccionar “Ventana(WPF)”.

No olvides cambiar el nombre de la ventana (abajo).



Útil

Mandar llamar otra ventana u ocultarla (botón regresar)

Se programa en la ventana que está el botón

**1)usar este**

private void btRegresar\_Click(object sender, RoutedEventArgs e) {

MainWindow main = new MainWindow();

main.Show();

this.Hide();

}

**2)**

private void btRegresar\_Click(object sender, RoutedEventArgs e) {

this.Close();

Alta a = new Alta();

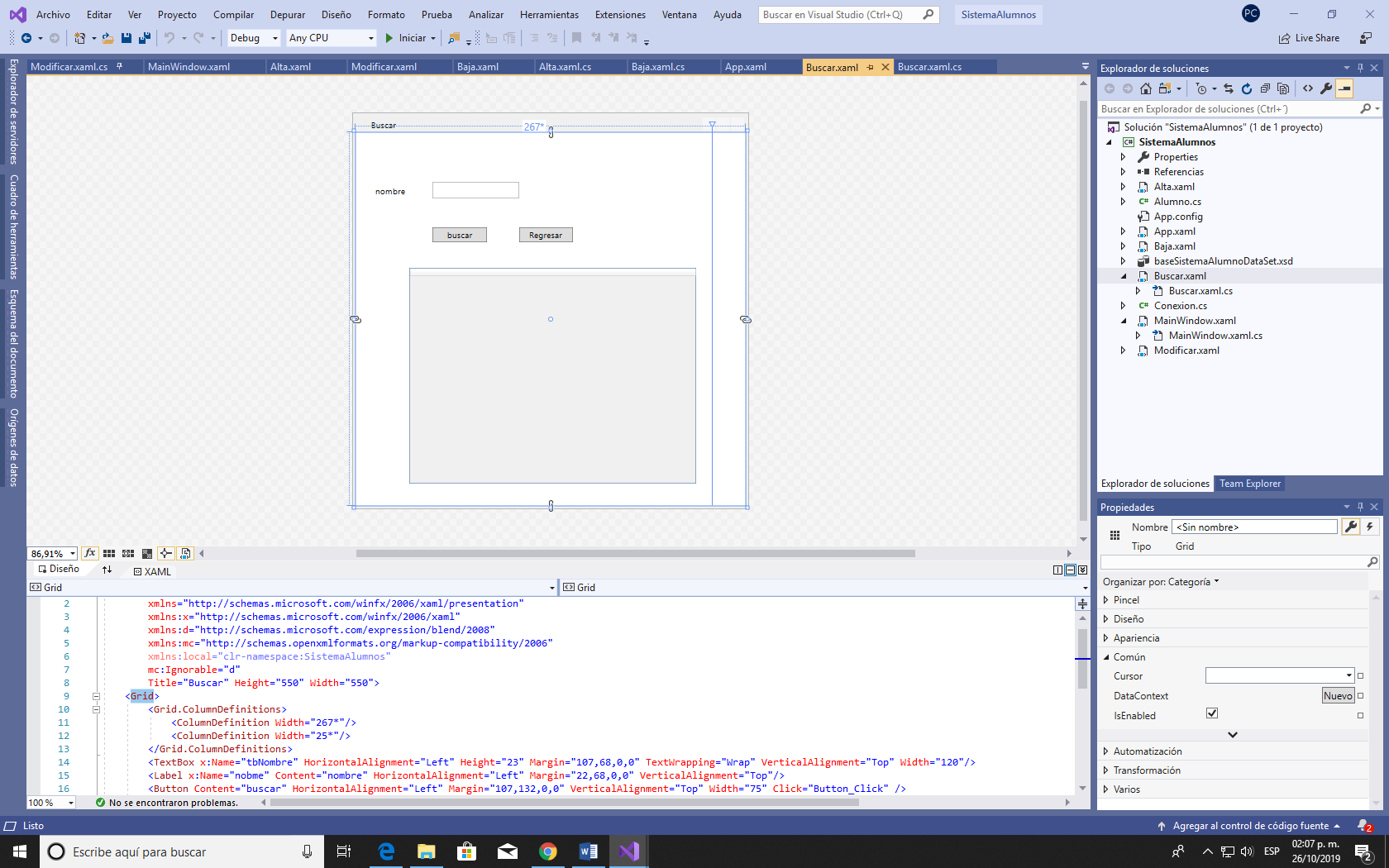
//sustitúyase “alta” por el nombre de la ventana que quiero abrir

a.Show();

}

Para mostrar varios datos tipo “búsqueda”, “consulta”, “reporte”.

Usar dataGrid



**Crear conexión**

Add------New item-----clase----ponerle nombre a la clase-----Conexion.cs ---agregar

namespace SistemaAlumnos

{

class Conexion

{

//desde aquí

SqlCommand cmd;

SqlDataReader rd;

public static SqlConnection conectar() {

SqlConnection con;

try {

con = new SqlConnection("Data Source=CC102-06\\SA;Initial Catalog=baseSistemaAlumno;User ID=sa;Password=adminadmin");

con.Open();

MessageBox.Show("si se pudo hacer la conexión");

} catch (Exception ex) {

con = null;

MessageBox.Show("no se pudo hacer la conexión" + ex);

}

return con;

}

//importar librerías para SQL Y MESSAGEBOX

using System.Data.SqlClient;

using System.Windows;

**llenar combo box**

//importar librerias

using System.Windows.Controls;

Se pone en la clase conexion y se manda llamar en el WINDOW\_LOADED de la ventana donde se usará

enConexion

//desde aqui

public static void llenarComboProgramas(ComboBox cb)

{

try {

SqlConnection conexion =conectar();

if(conexion != null)

{

SqlCommand cmd = new SqlCommand("select nombre from programas", conexion);

SqlDataReader rd = cmd.ExecuteReader();

while (rd.Read())

{

cb.Items.Add(rd.GetString(0));

}

cb.SelectedIndex = 0; // sirve para poner el primer elemento en el combo

rd.Close();

conexion.Close();

}

}

catch(Exception ex) {

MessageBox.Show("error" + ex);

}

}

en ventana

private void Window\_Loaded(object sender, RoutedEventArgs e)

{

//desde aqui

Conexion.llenarComboProgramas(cbPrograma);

}

**crear clase “alumno”**

namespace registroAspirantes

{

class Aspirante

{

//atributos

public String nombre, apellidoPaterno, apellidoMat, sexo, correo, fechaNac;

public int grado, programa;

//constructores

public Aspirante(string nombre, string apellidoPaterno, string apellidoMat, string sexo, string correo, string fechaNac, int grado, int programa)

{

this.nombre = nombre;

this.apellidoPaterno = apellidoPaterno;

this.apellidoMat = apellidoMat;

this.sexo = sexo;

this.correo = correo;

this.fechaNac = fechaNac;

this.grado = grado;

this.programa = programa;

}

public Aspirante() {

}

//alta

public String alta(Aspirante a)

{

String res = "";

int i;

try {

SqlConnection con = Conexion.agregarConexion();

SqlCommand cmd = new SqlCommand("select top (1) idAspirante from aspirantes order by idAspirante desc", con);

SqlDataReader rd = cmd.ExecuteReader();

if (rd.Read()) {

i = rd.GetInt32(0)+1;

rd.Close();

SqlCommand cmd2 = new SqlCommand(String.Format("insert into aspirantes values" +

"({0},'{1}', '{2}', '{3}', '{4}', '{5}', '{6}',{7},{8})", i, a.nombre, a.apellidoPaterno, a.apellidoMat, a.fechaNac, a.correo,a.grado, a.programa, con);

cmd2.ExecuteNonQuery();

}

else

{

rd.Close();

SqlCommand cmd2 = new SqlCommand(String.Format("insert into aspirantes values" +

"({0},'{1}', '{2}', '{3}', '{4}', '{5}', '{6}',{7},{8})", 1, a.nombre, a.apellidoPaterno, a.apellidoMat, a.fechaNac, a.correo, a.grado, a.programa, con);

cmd2.ExecuteNonQuery();

}

res = "alta exitosa";

con.Close();

} catch(Exception ex)

{

res = "alta no exitosa" + ex.Message;

}

return res;

}//cierre metodo

//modificar

public String modifica(int programa, String nombre) {

String res="";

try {

SqlConnection con = Conexion.agregarConexion();

SqlCommand cmd = new SqlCommand(String.Format("update aspirantes set clavePrograma={0} where nombre='{1}'", programa, nombre), con);

cmd.ExecuteNonQuery();

res = "alta exitosa";

} catch(Exception ex) {

res = "error" + ex;

}

return res;

}

//busca consulta

public String generaReporte(DataGrid dgreporte, int id)

{

String res = "";

try

{

SqlConnection con = Conexion.conectar();

String query = String.Format("select \* from curso where idEdo={0}", id);

SqlCommand cmd = new SqlCommand(query, con);

SqlDataReader rd = cmd.ExecuteReader();

//rd.Read();

dgreporte.ItemsSource = rd;

res = "reporte exitoso";

}

catch (Exception ex)

{

res = "error" + ex;

}

return res;

}

}

//elimina, baja

public String darDeBaja(String nombre){

String res = "";

try {

SqlConnection con = Conexion.agregaConexion();

String query = String.Format(" delete from aspirante where nombre='{0}’ “, nombre);

SqlCommand cmd = new SqlCommand(query, con);

cmd.ExecuteNonQuery();

con.Close();

res = "Baja exitosa";

}

catch (Exception e ){

res = "No se pudo dar de baja" + e.ToString();

}

return res;

}

**Mandar llamar**

**alta, baja, modifica, busca en botones**

//alta

private void BtnAlta\_Click(object sender, RoutedEventArgs e) {

String res="";

Badge a = new Badge(txNom.Text, Int32.Parse(txHora.Text), (int)cbEstado.SelectedIndex);

res = a.alta(a);

if (res.Equals("alta exitosa"))

MessageBox.Show(" dado de alta");

else

MessageBox.Show("no se pudo dar de alta");

}

//baja

private void btBaja\_Click(object sender, RoutedEventArgs e) {

Aspirantes a = new Aspirantes(); MessageBox.Show(a.darDeBaja(cbClaveUica.SelectedItem.ToString()));

}

//modifica

private void BtnModificar\_Click(object sender, RoutedEventArgs e) {

Badge a = new Badge();

MessageBox.Show(a.modifica(Convert.ToInt32( cbClave.SelectedItem), Convert.ToInt32( txHoras.Text)));

}

//consulta, reporte, busca

private void BtnRep\_Click(object sender, RoutedEventArgs e) {

try

{

Badge a = new Badge();

a.generaReporte(dgReporte, cbEstado.SelectedIndex);

}catch(Exception ex) {

MessageBox.Show("No se puede mostrar" + ex);

}

}

**Web**

**Conexion**

Siempre al inicio de cada página

arriba del load

protected OdbcConnection conectarBD()

{

String stringConexion = "Driver={SQL Server Native Client 11.0};Server=112SALAS07;Uid=sa;Pwd=sqladmin;Database=datosCursos";

try

{

OdbcConnection conexion = new OdbcConnection(stringConexion);

conexion.Open();

lbContador.Text = "conexion exitosa";

return conexion;

}

catch (Exception ex)

{

lbContador.Text = ex.StackTrace.ToString();

return null;

}

}

**Revisar contraseña y password(o similar)**

agrega una etiqueta o messagebox para mostrar cuando son incorrectos

protected void btnSig\_Click(object sender, EventArgs e)

{

OdbcConnection miConexion = conectarBD();

if (miConexion != null)//o sea si se conecto

{

String query = "select idPersona from persona where nombre='" + txNom.Text + "' and correo='" + txCorreo.Text + "'";

OdbcCommand cmd = new OdbcCommand(query, miConexion);

OdbcDataReader rd = cmd.ExecuteReader(); //trae datos

if (rd.HasRows)

{ // si hya renglones entonces...

rd.Read(); //estoy en el primer dato

Session["id"] = rd.GetInt32(0).ToString();

Response.Redirect("Altas.aspx");

rd.Close();

}

else

lbContador.Text = "el nombre o correo son incorrectos";

}

}

**Cargar un dropdown desde BD**

Cambia el query

protected void Page\_Load(object sender, EventArgs e)

{

if (!IsPostBack)

{

OdbcConnection miConexion = conectarBD();

if (miConexion != null)

{

String query = String.Format("select nombre from curso ");

OdbcCommand cmd = new OdbcCommand(query, miConexion);

OdbcDataReader rd = cmd.ExecuteReader();

ddCursos.Items.Clear();

while (rd.Read())

{

ddCursos.Items.Add(rd.GetString(0));

}

rd.Close();

}

}

}

**Alta, Registrar**

OdbcConnection con = conectarBD();

if (con != null)

{

String query2 = "select max(idCursoTomado) from cursoTomado";

OdbcCommand sql2 = new OdbcCommand(query2, con);

int folio = Int16.Parse(sql2.ExecuteScalar().ToString()) + 1;

String query = String.Format("insert into cursoTomado values ({0},{1},{2})", folio, Session["id"], ddCursos.SelectedIndex);

OdbcCommand sql = new OdbcCommand(query, con);

sql.ExecuteNonQuery();

Response.Redirect("Reporte.aspx");

con.Close();

}

**Baja, Eliminar**

protected void Button1\_Click(object sender, EventArgs e)

{

OdbcConnection con = conectarBD();

if (con != null)

{

String query = String.Format("delete from datosaAlumno18 where id='{0}'",ddCurso.SelectedValue);

OdbcCommand com = new OdbcCommand(query, con);

com.ExecuteNonQuery();

}

Response.Redirect("Pagina2.aspx");

}

**Ir a otra página**

Response.Redirect("Reporte.aspx");

**Mostrar en GridView**

if (!IsPostBack)

{

OdbcConnection con = conectarBD();

if (con != null)

{

String query = String.Format("select curso.\* from curso inner join cursoTomado on curso.idCurso=cursoTomado.idCursoTomado " +

"inner join persona on persona.idPersona = cursoTomado.idPersona" +

" where cursoTomado.idPersona = {0}", Session["id"]);

OdbcCommand com = new OdbcCommand(query, con);

OdbcDataReader rd = com.ExecuteReader();

gvReporte.DataSource = rd;

gvReporte.DataBind();

con.Close();

}

}

**Session**

protected void bSiguiente\_Click(object sender, EventArgs e){

Session["nombre"] = tbNombre.Text;

Response.Redirect("Pagina2t.aspx");

}