## DAO: Conexión.

using System;

using System.Collections.Generic;

using System.Data;

using System.Data.SqlClient;

using System.Linq;

using System.Web;

namespace SICAP.DAO

{

public abstract class Conexion

{

private string cdnConexion;

private SqlConnection cnn;

public Conexion()

{

cdnConexion = @"Data Source = .;Initial Catalog= SICAP ;Trusted\_Connection=True;";

//cdnConexion = @"Data Source = DESKTOP-PB21R03\SQLEXPRESS;Initial Catalog= SICAP ;user id = pedro; password = 14567524;";

cnn = new SqlConnection(cdnConexion);

}

/// <summary>

/// Consultar una tabla sql

/// </summary>

/// <param name="cmd">Se le pasa un sqlcommand</param>

/// <returns>Retorna un DataTable</returns>

protected DataTable consulta(SqlCommand cmd)

{

try

{

cmd.Connection = cnn;

cnn.Open();

DataTable tabla = new DataTable();

tabla.Load(cmd.ExecuteReader());

cnn.Close();

return tabla;

}

catch (Exception ex)

{

throw new Exception(ex.Message);

}

}

/// <summary>

/// Ejecuta la instruccion SQL para la modificacion de tablas

/// </summary>

/// <param name="cmd">Se le pasa un sqlcommand</param>

/// <returns>el numero de filas afectadas</returns>

protected int ejectuarSQL(SqlCommand cmd)

{

try

{

cnn.Open();

cmd.Connection = cnn;

int resultado = cmd.ExecuteNonQuery();

cnn.Close();

return resultado;

}

catch (Exception ex)

{

throw new Exception(ex.Message);

}

}

}

}

## Modelos: Usuario.

using SICAP.DAO;

using SICAP.Utelirias;

using System;

using System.Collections.Generic;

using System.Data;

using System.Data.SqlClient;

using System.Linq;

using System.Web;

namespace SICAP.Modelos

{

public class Usuario : Conexion

{

public int id\_usuario { set; get; }

public string nombre { set; get; }

public string paterno { set; get; }

public string materno { set; get; }

public string telefono { set; get; }

public string email { set; get; }

public string especialidad { set; get; }

public string contrasena { set; get; }

public int area { set; get; }

public int rol { set; get; }

public string ruta { get; set; }

public bool guardar()

{

try

{

string query = "INSERT INTO usuarios(nombre,paterno,materno,telefono,email,especialidad,contrasena,area,rol,ruta)"

+ "values(dbo.InitCap(@nombre),dbo.InitCap(@paterno),dbo.InitCap(@materno),@telefono,@email,dbo.InitCap(@especialidad),@contrasena,@area,@rol,@ruta)";

SqlCommand cmd = new SqlCommand(query);

cmd.Parameters.AddWithValue("@nombre", nombre);

cmd.Parameters.AddWithValue("@materno", materno);

cmd.Parameters.AddWithValue("@paterno", paterno);

cmd.Parameters.AddWithValue("@telefono", telefono);

cmd.Parameters.AddWithValue("@email", email);

cmd.Parameters.AddWithValue("@especialidad", especialidad);

cmd.Parameters.AddWithValue("@contrasena", Encriptar.GetSHA1(contrasena));

cmd.Parameters.AddWithValue("@area", area);

cmd.Parameters.AddWithValue("@rol", rol);

cmd.Parameters.AddWithValue("@ruta", ruta);

return (ejectuarSQL(cmd) > 0) ? true : false;

}

catch (Exception ex)

{

throw new Exception(ex.Message);

}

}

public bool validarAcceso()

{

try

{

string query = "SELECT id\_usuario,rol, nombre,paterno,materno,ruta,id\_activo FROM usuarios WHERE email = @email AND contrasena = @contrasena";

SqlCommand cmd = new SqlCommand(query);

cmd.Parameters.AddWithValue("@email", email);

cmd.Parameters.AddWithValue("@contrasena",Encriptar.GetSHA1(contrasena));

DataTable usuario = consulta(cmd);

if (usuario.Rows.Count > 0)

{

id\_usuario = (int)(usuario.Rows[0]["id\_usuario"]);

rol = int.Parse(usuario.Rows[0]["rol"].ToString());

nombre = (string)(usuario.Rows[0]["nombre"]);

paterno = (string)(usuario.Rows[0]["paterno"]);

materno = (string)(usuario.Rows[0]["materno"]);

ruta = (string)(usuario.Rows[0]["ruta"]);

}

return (usuario.Rows.Count > 0 && (int)(usuario.Rows[0]["id\_activo"]) == 1) ? true : false;

}

catch (Exception ex)

{

throw new Exception(ex.Message);

}

}

public bool correoExistente()

{

try

{

string query = "SELECT email FROM usuarios WHERE email = @email";

DataTable usuarios = new DataTable();

SqlCommand cmd = new SqlCommand(query);

cmd.Parameters.AddWithValue("@email", email);

usuarios = consulta(cmd);

return (usuarios.Rows.Count > 0) ? true : false;

}

catch (Exception ex)

{

throw new Exception(ex.Message);

}

}

public int asignarArea(string area)

{

try

{

string query = "SELECT id\_area from areas WHERE area = @area";

SqlCommand cmd = new SqlCommand(query);

cmd.Parameters.AddWithValue("@area", area);

return int.Parse(consulta(cmd).Rows[0]["id\_area"].ToString());

}

catch (Exception ex)

{

throw new Exception(ex.Message);

}

}

public DataTable traerAreas()

{

try

{

string query = "SELECT area FROM Areas";

SqlCommand cmd = new SqlCommand(query);

return consulta(cmd);

}

catch (Exception ex)

{

throw new Exception(ex.Message);

}

}

public DataTable traerUsuarios()

{

try

{

string query = "SELECT U.id\_usuario, U.nombre,U.paterno,U.materno,U.ruta,A.area, U.especialidad FROM usuarios U INNER JOIN areas A ON U.area = A.id\_area;";

SqlCommand cmd = new SqlCommand(query);

return consulta(cmd);

}

catch (Exception ex)

{

throw new Exception(ex.Message);

}

}

public DataTable traerUsuario()

{

try

{

string query = "SELECT U.id\_usuario, U.nombre,U.paterno,U.materno,U.email,U.telefono,U.ruta,A.area, U.especialidad, R.rol FROM usuarios U INNER JOIN "

+ "areas A ON U.area = A.id\_area INNER JOIN roles R ON U.rol = R.id\_rol WHERE U.id\_usuario = @id\_usuario;";

SqlCommand cmd = new SqlCommand(query);

cmd.Parameters.AddWithValue("@id\_usuario", id\_usuario);

return consulta(cmd);

}

catch (Exception ex)

{

throw new Exception(ex.Message);

}

}

public int modificarUsuario()

{

try

{

string query = "UPDATE usuarios SET nombre = dbo.InitCap(@nombre), paterno = dbo.InitCap(@paterno), materno = dbo.InitCap(@materno), rol = @rol, area = @area, telefono = @telefono, " +

"email = @email, especialidad = dbo.InitCap(@especialidad), ruta = @ruta WHERE id\_usuario = @id\_usuario";

SqlCommand cmd = new SqlCommand(query);

cmd.Parameters.AddWithValue("@nombre",nombre);

cmd.Parameters.AddWithValue("@paterno", paterno);

cmd.Parameters.AddWithValue("@materno", materno);

cmd.Parameters.AddWithValue("@rol", rol);

cmd.Parameters.AddWithValue("@area", area);

cmd.Parameters.AddWithValue("@telefono", telefono);

cmd.Parameters.AddWithValue("@email", email);

cmd.Parameters.AddWithValue("@especialidad", especialidad);

cmd.Parameters.AddWithValue("@ruta", ruta);

cmd.Parameters.AddWithValue("@id\_usuario",id\_usuario);

return ejectuarSQL(cmd);

}

catch (Exception ex)

{

throw new Exception(ex.Message);

}

}

public string getNombreUsuario()

{

try

{

string query = "SELECT \* FROM usuarios WHERE email = @email";

SqlCommand cmd = new SqlCommand(query);

cmd.Parameters.AddWithValue("@email",email);

DataTable usuario = consulta(cmd);

string nombre = (string)(usuario.Rows[0]["nombre"]);

nombre += " " + (string)(usuario.Rows[0]["paterno"]);

nombre += " " + (string)(usuario.Rows[0]["materno"]);

return nombre;

}

catch (Exception ex)

{

throw new Exception(ex.Message);

}

}

public int actualizarPass(string pass)

{

try

{

string query = "UPDATE usuarios SET contrasena = @contrasena WHERE email = @email";

SqlCommand cmd = new SqlCommand(query);

cmd.Parameters.AddWithValue("@contrasena",pass);

cmd.Parameters.AddWithValue("@email",email);

return ejectuarSQL(cmd);

}

catch (Exception ex)

{

throw new Exception(ex.Message);

}

}

public int CambiarPass(string pass)

{

try

{

string query = "UPDATE usuarios SET contrasena = @contrasena WHERE id\_usuario=@id\_usuario";

SqlCommand cmd = new SqlCommand(query);

cmd.Parameters.AddWithValue("@contrasena",Encriptar.GetSHA1(pass));

cmd.Parameters.AddWithValue("@id\_usuario", id\_usuario);

return ejectuarSQL(cmd);

}

catch (Exception ex)

{

throw new Exception(ex.Message);

}

}

public bool validarPass()

{

try

{

string query = "SELECT contrasena FROM usuarios WHERE contrasena = @contrasena AND id\_usuario= @id\_usuario";

SqlCommand cmd = new SqlCommand(query);

cmd.Parameters.AddWithValue("@contrasena", Encriptar.GetSHA1(contrasena));

cmd.Parameters.AddWithValue("@id\_usuario", id\_usuario);

return (consulta(cmd).Rows.Count > 0) ? true : false;

}

catch (Exception ex)

{

throw new Exception(ex.Message);

}

}

public int AgregarArea(string a)

{

try

{

string query = "INSERT INTO areas(area) VALUES (@area); ";

SqlCommand cmd = new SqlCommand(query);

cmd.Parameters.AddWithValue("@area", a);

return ejectuarSQL(cmd);

}

catch (Exception ex)

{

throw new Exception(ex.Message);

}

}

public bool areaEnUso(string a)

{

try

{

string query = "SELECT A.area FROM usuarios U INNER JOIN areas A ON U.area = A.id\_area WHERE A.area = @area;";

SqlCommand cmd = new SqlCommand(query);

cmd.Parameters.AddWithValue("@area", a);

return (consulta(cmd).Rows.Count > 0) ? true : false;

}

catch (Exception ex)

{

throw new Exception(ex.Message);

}

}

public int eliminarArea(string a)

{

try

{

string query = "DELETE FROM areas WHERE area=@area;";

SqlCommand cmd = new SqlCommand(query);

cmd.Parameters.AddWithValue("@area", a);

return ejectuarSQL(cmd);

}

catch (Exception ex)

{

throw new Exception(ex.Message);

}

}

public DataTable traerMisProyectos()

{

try

{

string query = "SELECT p.id\_proyecto, p.proyecto, p.fecha\_registro,CAST(p.fecha\_inicio AS varchar) as fecha\_inicio ,CAST(p.fecha\_final AS varchar) as fecha\_final FROM relaciones r INNER JOIN usuarios u ON r.id\_usuarios = u.id\_usuario " +

"INNER JOIN proyectos p ON r.id\_proyecto = p.id\_proyecto WHERE r.id\_usuarios = @id\_usuario AND estatus = 1;";

SqlCommand cmd = new SqlCommand(query);

cmd.Parameters.AddWithValue("@id\_usuario", id\_usuario);

return consulta(cmd);

}

catch (Exception ex)

{

throw new Exception(ex.Message);

}

}

public static bool validarExtensionImg(string nombreArchivo)

{

try

{

string extension = nombreArchivo.Split('.')[1];

bool extensionCorrecta = false;

if (extension == "jpg")

{

extensionCorrecta = true;

}

else if (extension == "png")

{

extensionCorrecta = true;

}

else if (extension == "gif")

{

extensionCorrecta = true;

}

return extensionCorrecta;

}

catch (Exception ex)

{

throw new Exception(ex.Message);

}

}

public int deshabilitar()

{

try

{

string query = "UPDATE usuarios SET id\_activo = 2 WHERE id\_usuario = @id\_usuario";

SqlCommand cmd = new SqlCommand(query);

cmd.Parameters.AddWithValue("@id\_usuario",id\_usuario);

return ejectuarSQL(cmd);

}

catch (Exception ex)

{

throw new Exception(ex.Message);

}

}

public int habilitar()

{

try

{

string query = "UPDATE usuarios SET id\_activo = 1 WHERE id\_usuario = @id\_usuario";

SqlCommand cmd = new SqlCommand(query);

cmd.Parameters.AddWithValue("@id\_usuario", id\_usuario);

return ejectuarSQL(cmd);

}

catch (Exception ex)

{

throw new Exception(ex.Message);

}

}

public bool IsHabiltado()

{

try

{

string query = "SELECT id\_activo FROM usuarios WHERE id\_usuario = @id\_usuario";

SqlCommand cmd = new SqlCommand(query);

cmd.Parameters.AddWithValue("@id\_usuario", id\_usuario);

int activo = (int)(consulta(cmd).Rows[0]["id\_activo"]);

return (activo == 1) ? true : false;

}

catch (Exception ex)

{

throw new Exception(ex.Message);

}

}

public bool isAdmin()

{

try

{

string query = "SELECT rol FROM usuarios WHERE id\_usuario = @id\_usuario";

SqlCommand cmd = new SqlCommand(query);

cmd.Parameters.AddWithValue("@id\_usuario",id\_usuario);

bool r = (consulta(cmd).Rows[0]["rol"].ToString() == "1") ? true : false;

return r;

}

catch (Exception ex)

{

throw new Exception(ex.Message);

}

}

}

}

## Modelos: Proyecto.

using SICAP.DAO;

using System;

using System.Collections.Generic;

using System.Data;

using System.Data.SqlClient;

using System.Linq;

using System.Web;

namespace SICAP.Modelos

{

public class Proyecto : Conexion

{

public int id\_proyecto { set; get; }

public string proyecto { set; get; }

public string observaciones { set; get; }

public string fecha\_inicio { set; get; }

public string fecha\_final { set; get; }

public string estatus { set; get; }

public int id\_estatus { get; set; }

public List<int> usuarios { set; get; }

public List<string []> actividades { set; get; }

public Proyecto()

{

usuarios = new List<int>();

actividades = new List<string[]>();

}

public int guardar()

{

try

{

string query = "INSERT INTO proyectos(proyecto,observaciones,fecha\_inicio,fecha\_final) VALUES(@proyecto,@observaciones,@fecha\_inicio,@fecha\_final)";

SqlCommand cmd = new SqlCommand(query);

cmd.Parameters.AddWithValue("@proyecto", proyecto);

cmd.Parameters.AddWithValue("@observaciones", observaciones);

cmd.Parameters.AddWithValue("@fecha\_inicio", fecha\_inicio);

cmd.Parameters.AddWithValue("fecha\_final", fecha\_final);

return ejectuarSQL(cmd);

}

catch (Exception ex)

{

throw new Exception(ex.Message);

}

}

public int getID()

{

try

{

string query= "SELECT TOP 1 id\_proyecto FROM proyectos ORDER BY id\_proyecto DESC;";

DataTable id = consulta(new SqlCommand(query));

int id\_proyectos = (int)(id.Rows[0]["id\_proyecto"]);

return id\_proyectos;

}

catch(Exception ex)

{

throw new Exception(ex.Message);

}

}

public void asignarUsuarios()

{

try

{

string query = "INSERT INTO relaciones (id\_usuarios, id\_proyecto)VALUES(@id\_usuario,@id\_proyecto);";

for(int i = 0; i<usuarios.Count; i++)

{

SqlCommand cmd = new SqlCommand(query);

cmd.Parameters.AddWithValue("@id\_proyecto", id\_proyecto);

cmd.Parameters.AddWithValue("@id\_usuario", usuarios[i]);

ejectuarSQL(cmd);

}

}

catch (Exception ex)

{

throw new Exception(ex.Message);

}

}

public void asignarActividades()

{

try

{

string query = "INSERT INTO actividades (actividad,observaciones,fecha\_entrega,id\_proyecto) " +

"VALUES (@actividad,@observaciones,@fecha\_entrega,@id\_proyecto);";

for(int i = 0; i < actividades.Count; i++)

{

SqlCommand cmd = new SqlCommand(query);

cmd.Parameters.AddWithValue("@actividad", actividades[i][0]);

cmd.Parameters.AddWithValue("@observaciones",actividades[i][1]);

cmd.Parameters.AddWithValue("@fecha\_entrega", actividades[i][2]);

cmd.Parameters.AddWithValue("@id\_proyecto", id\_proyecto);

ejectuarSQL(cmd);

}

}

catch (Exception ex)

{

throw new Exception(ex.Message);

}

}

public DataTable traerProyectos()

{

try

{

string query = "SELECT p.id\_proyecto, p.proyecto, p.fecha\_registro, CAST(p.fecha\_inicio AS VARCHAR) AS fecha\_inicio" +

",CAST(p.fecha\_final AS VARCHAR) AS fecha\_final, e.estatus FROM proyectos" +

" p INNER JOIN estatus e ON p.estatus = e.id\_estatus WHERE p.estatus = @estatus";

SqlCommand cmd = new SqlCommand(query);

cmd.Parameters.AddWithValue("@estatus", id\_estatus);

return consulta(cmd);

}

catch (Exception ex)

{

throw new Exception(ex.Message);

}

}

public void cargarDatos() // aun tenemos que ver como cargar a los usuarios

{

try

{

string query = "SELECT p.id\_proyecto, p.proyecto, p.fecha\_registro,CAST(p.fecha\_inicio AS varchar) as fecha\_inicio ,CAST(p.fecha\_final AS varchar) as fecha\_final ,p.observaciones," +

" e.estatus FROM proyectos p INNER JOIN estatus e ON p.estatus = e.id\_estatus WHERE id\_proyecto = @id\_proyecto;";

SqlCommand cmd = new SqlCommand(query);

cmd.Parameters.AddWithValue("@id\_proyecto",id\_proyecto);

DataTable pro = consulta(cmd);

proyecto = (string)(pro.Rows[0]["proyecto"]);

observaciones = (string)(pro.Rows[0]["observaciones"]);

fecha\_inicio = (string)(pro.Rows[0]["fecha\_inicio"]);

fecha\_final = (string)(pro.Rows[0]["fecha\_final"]);

estatus = (string)(pro.Rows[0]["estatus"]);

}

catch (Exception ex)

{

throw new Exception(ex.Message);

}

}

public string[] traerUsuariosLista()

{

try

{

string query = "SELECT u.id\_usuario, u.nombre, u.paterno, u.materno, u.especialidad FROM relaciones" +

" r INNER JOIN usuarios u ON r.id\_usuarios = u.id\_usuario WHERE r.id\_proyecto = @id\_proyecto;";

SqlCommand cmd = new SqlCommand(query);

cmd.Parameters.AddWithValue("@id\_proyecto", id\_proyecto);

DataTable pro = consulta(cmd);

string[] user = new string[pro.Rows.Count];

for(int i = 0; i < pro.Rows.Count; i++)

{

string usuario = Convert.ToString((int)(pro.Rows[i]["id\_usuario"]));

usuario += " - " + (string)(pro.Rows[i]["nombre"]);

usuario += " " + (string)(pro.Rows[i]["paterno"]);

usuario += " " + (string)(pro.Rows[i]["materno"]);

usuario += " - " + (string)(pro.Rows[i]["especialidad"]);

user[i] = usuario;

}

return user;

}

catch (Exception ex)

{

throw new Exception(ex.Message);

}

}

public int cambiarEstatus(int e)

{

try

{

string query = "UPDATE proyectos SET estatus = @estatus WHERE id\_proyecto = @id\_proyecto";

SqlCommand cmd = new SqlCommand(query);

cmd.Parameters.AddWithValue("@estatus", e);

cmd.Parameters.AddWithValue("@id\_proyecto", id\_proyecto);

return ejectuarSQL(cmd);

}

catch (Exception ex)

{

throw new Exception(ex.Message);

}

}

public bool expirado()

{

try

{

DateTime fecha = DateTime.Parse(fecha\_final);

DateTime hoy = DateTime.Today;

return (DateTime.Compare(hoy, fecha) > 0) ? true : false;

}

catch (Exception ex)

{

throw new Exception(ex.Message);

}

}

public int concluir()

{

try

{

string query = "UPDATE proyectos SET estatus = @estatus WHERE id\_proyecto = @id\_proyecto;";

SqlCommand cmd = new SqlCommand(query);

cmd.Parameters.AddWithValue("@estatus", 2);

cmd.Parameters.AddWithValue("id\_proyecto", id\_proyecto);

return ejectuarSQL(cmd);

}

catch (Exception ex)

{

throw new Exception(ex.Message);

}

}

public bool eliminarProyecto()

{

try

{

string ocupado = "SELECT estatus FROM proyectos WHERE id\_proyecto = @id\_proyecto";

SqlCommand cmd = new SqlCommand(ocupado);

cmd.Parameters.AddWithValue("@id\_proyecto",id\_proyecto);

bool eliminado = (consulta(cmd).Rows[0]["estatus"].ToString() == "1") ? true : false;

if (!eliminado)

{

string queryRelaciones = "DELETE FROM relaciones WHERE id\_proyecto = @id\_proyecto; ";

string queryAvances = "DELETE FROM avances WHERE id\_proyecto = @id\_proyecto; ";

string queryActividades = "DELETE FROM actividades WHERE id\_proyecto = @id\_proyecto; ";

string queryProyectos = "DELETE FROM proyectos WHERE id\_proyecto = @id\_proyecto; ";

SqlCommand cmd2 = new SqlCommand(queryRelaciones);

cmd2.Parameters.AddWithValue("@id\_proyecto", id\_proyecto);

int e = ejectuarSQL(cmd2);

SqlCommand cmd3 = new SqlCommand(queryAvances);

cmd3.Parameters.AddWithValue("@id\_proyecto", id\_proyecto);

e = ejectuarSQL(cmd3);

SqlCommand cmd4 = new SqlCommand(queryActividades);

cmd4.Parameters.AddWithValue("@id\_proyecto", id\_proyecto);

e = ejectuarSQL(cmd4);

SqlCommand cmd5 = new SqlCommand(queryProyectos);

cmd5.Parameters.AddWithValue("@id\_proyecto", id\_proyecto);

e = ejectuarSQL(cmd5);

}

return eliminado;

}

catch (Exception ex)

{

throw new Exception(ex.Message);

}

}

}

}

## Modelos: Avance.

using SICAP.DAO;

using System;

using System.Collections.Generic;

using System.Data;

using System.Data.SqlClient;

using System.Linq;

using System.Web;

namespace SICAP.Modelos

{

public class Avance : Conexion

{

public int id\_avance { set; get; }

public string NombreAvance { set; get; }

public string fecha\_registro { set; get; }

public string observaciones { set; get; }

public string rutaDoc { set; get; }

public int id\_actividad { set; get; }

public string actividad { set; get; }

public int id\_proyecto { set; get; }

public int id\_usuario { set; get; }

public string usuario { set; get; }

public Avance() { }

public Avance(int id\_avance)

{

try

{

string query = "SELECT a.id\_avance AS ID,a.ArchivoRuta ,a.avance,a.observaciones,CAST(a.fecha\_registro AS varchar) " +

"AS fecha\_registro,CONCAT(u.nombre,' ',u.paterno,' ',u.materno) AS Nombre FROM avances a INNER " +

"JOIN usuarios u ON a.id\_usuario = u.id\_usuario WHERE a.id\_avance = @ID;";

SqlCommand cmd = new SqlCommand(query);

cmd.Parameters.AddWithValue("@ID",id\_avance);

DataTable a = consulta(cmd);

NombreAvance = (string)(a.Rows[0]["avance"]);

observaciones = (string)(a.Rows[0]["observaciones"]);

rutaDoc = (string)(a.Rows[0]["ArchivoRuta"]);

fecha\_registro = (string)(a.Rows[0]["fecha\_registro"]);

usuario = (string)(a.Rows[0]["Nombre"]);

}

catch (Exception ex)

{

throw new Exception(ex.Message);

}

}

public List<string> getActividades()

{

try

{

string query = "SELECT \* FROM actividades WHERE id\_proyecto = @id\_proyecto AND fecha\_entrega >= GETDATE() ORDER BY fecha\_entrega ASC;";

SqlCommand cmd = new SqlCommand(query);

cmd.Parameters.AddWithValue("@id\_proyecto", id\_proyecto);

DataTable act = consulta(cmd);

List<string> actividades = new List<string>();

for(int i = 0; i < act.Rows.Count; i++)

{

string fila = act.Rows[i]["id\_actividad"].ToString() + " - " + act.Rows[i]["actividad"].ToString();

actividades.Add(fila);

}

return actividades;

}

catch (Exception ex)

{

throw new Exception(ex.Message);

}

}

public int guardar()

{

try

{

string query = "INSERT INTO avances(avance,observaciones,id\_actividad,id\_usuario,id\_proyecto) " +

"VALUES(@avance,@observaciones,@id\_actividad,@id\_usuario,@id\_proyecto)";

SqlCommand cmd = new SqlCommand(query);

cmd.Parameters.AddWithValue("@avance",NombreAvance);

cmd.Parameters.AddWithValue("@observaciones", observaciones);

cmd.Parameters.AddWithValue("@id\_actividad", id\_actividad);

cmd.Parameters.AddWithValue("@id\_usuario", id\_usuario);

cmd.Parameters.AddWithValue("@id\_proyecto", id\_proyecto);

return ejectuarSQL(cmd);

}

catch (Exception ex)

{

throw new Exception(ex.Message);

}

}

public int guardar(string ruta)

{

try

{

string query = "INSERT INTO avances(avance,observaciones,ArchivoRuta,id\_actividad,id\_usuario,id\_proyecto) " +

"VALUES(@avance,@observaciones,@ruta,@id\_actividad,@id\_usuario,@id\_proyecto)";

SqlCommand cmd = new SqlCommand(query);

cmd.Parameters.AddWithValue("@avance", NombreAvance);

cmd.Parameters.AddWithValue("@observaciones", observaciones);

cmd.Parameters.AddWithValue("@id\_actividad", id\_actividad);

cmd.Parameters.AddWithValue("@id\_usuario", id\_usuario);

cmd.Parameters.AddWithValue("@id\_proyecto", id\_proyecto);

cmd.Parameters.AddWithValue("@ruta", ruta);

return ejectuarSQL(cmd);

}

catch (Exception ex)

{

throw new Exception(ex.Message);

}

}

public DataTable traerAvances()

{

try

{

string query = "SELECT a.id\_avance AS ID ,a.avance,a.observaciones,ac.actividad,a.fecha\_registro,CONCAT(u.nombre,' ',u.paterno,' ',u.materno)" +

" AS Nombre FROM avances a INNER JOIN usuarios u ON a.id\_usuario = u.id\_usuario INNER JOIN actividades ac ON a.id\_actividad = ac.id\_actividad " +

"WHERE a.id\_proyecto = @id\_proyecto";

SqlCommand cmd = new SqlCommand(query);

cmd.Parameters.AddWithValue("@id\_proyecto", id\_proyecto);

return consulta(cmd);

}

catch (Exception ex)

{

throw new Exception(ex.Message);

}

}

public static bool validarDocumento(string nombreArchivo)

{

try

{

string extension = nombreArchivo.Split('.')[1];

bool validado = false;

if (extension == "docx")

{

validado = true;

}

else if(extension == "pdf")

{

validado = true;

}

else if (extension == "xlsx")

{

validado = true;

}

else if (extension == "txt")

{

validado = true;

}

else if (extension == "pptx")

{

validado = true;

}

else if (extension == "jpg")

{

validado = true;

}

else if (extension == "xml")

{

validado = true;

}

else if (extension == "png")

{

validado = true;

}

return validado;

}

catch (Exception ex)

{

throw new Exception(ex.Message);

}

}

public bool pertenenAlproyecto()

{

try

{

string query = "SELECT \* FROM relaciones WHERE id\_usuarios = @id\_usuario";

SqlCommand cmd = new SqlCommand(query);

cmd.Parameters.AddWithValue("@id\_usuario",id\_usuario);

return (consulta(cmd).Rows.Count > 0) ? true : false;

}

catch (Exception ex)

{

throw new Exception(ex.Message);

}

}

}

}

## Modelos: Actividad.

using System;

using System.Collections.Generic;

using System.Data.SqlClient;

using SICAP.DAO;

using System.Data;

using System.Linq;

using System.Web;

namespace SICAP.Modelos

{

public class Actividad : Conexion

{

public int id\_actividad { set; get; }

public bool expirado()

{

try

{

string query = "SELECT CAST(fecha\_entrega AS VARCHAR) AS fecha\_entrega FROM actividades WHERE id\_actividad = @id\_actividad";

SqlCommand cmd = new SqlCommand(query);

cmd.Parameters.AddWithValue("@id\_actividad", id\_actividad);

DateTime fecha = DateTime.Parse(consulta(cmd).Rows[0]["fecha\_entrega"].ToString());

DateTime hoy = DateTime.Today;

return (DateTime.Compare(hoy, fecha) > 0) ? true : false;

}

catch (Exception ex)

{

throw new Exception(ex.Message);

}

}

public int concluir()

{

try

{

string query = "UPDATE actividades SET estatus = @estatus WHERE id\_actividad = @id\_actividad;";

SqlCommand cmd = new SqlCommand(query);

cmd.Parameters.AddWithValue("@estatus", 2);

cmd.Parameters.AddWithValue("id\_actividad", id\_actividad);

return ejectuarSQL(cmd);

}

catch (Exception ex)

{

throw new Exception(ex.Message);

}

}

public DataTable traerActividades(int id\_proyecto) // esto lo tenemos que presentar en un gridbview.

{

try

{

string query = "SELECT a.id\_actividad, a.actividad,a.observaciones,a.fecha\_entrega,e.estatus,a.id\_proyecto" +

" FROM actividades a INNER JOIN estatus e ON a.estatus = e.id\_estatus WHERE id\_proyecto = @id\_proyecto";

SqlCommand cmd = new SqlCommand(query);

cmd.Parameters.AddWithValue("@id\_proyecto", id\_proyecto);

return consulta(cmd);

}

catch (Exception ex)

{

throw new Exception(ex.Message);

}

}

}

}

## Admin.Master : CS

using System;

using System.Collections.Generic;

using System.Linq;

using System.Web;

using System.Web.UI;

using System.Web.UI.WebControls;

namespace SICAP

{

public partial class admin : System.Web.UI.MasterPage

{

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

{

if (Session["id\_usuario"] == null)

{

Response.Redirect("default.aspx");

}

else if((int)(Session["rol"]) == 2)

{

hlAdministrar.Visible = false;

hlProyectos.Visible = false;

hlEmpleados.Visible = false;

hlRegistrar.Visible = false;

hlVerProyectos.Visible = false;

hlAreas.Visible = false;

string js = "var p = location.href;var r = p.split('/');" +

"if(r[3] == 'misProyectos.aspx') { } else " +

"{location.href = './misProyectos.aspx'} } ";

ScriptManager.RegisterClientScriptBlock(this, this.GetType(), "mensaje",

js, true);

}

if (!IsPostBack)

{

hlMiPerfil.NavigateUrl = "usuario.aspx?id\_usuario=" + (int)(Session["id\_usuario"]);

}

string nombreUsuario = (string)(Session["nombre"]) + " " + (string)(Session["paterno"]) + " " + (string)(Session["materno"]);

lblNombreUsuario.Text = nombreUsuario;

userFoto.ImageUrl = (string)(Session["ruta"]);

}

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

{

Session["id\_usuario"] = null;

Response.Redirect("default.aspx");

}

}

}

## Areas.aspx

<%@ Page Title="" Language="C#" MasterPageFile="~/admin.Master" AutoEventWireup="true" CodeBehind="areas.aspx.cs" Inherits="SICAP.areas" %>

<asp:Content ID="Content1" ContentPlaceHolderID="head" runat="server">

</asp:Content>

<asp:Content ID="Content2" ContentPlaceHolderID="ContentPlaceHolder1" runat="server">

<div class="contenido card-panel animated fadeIn">

<h5 class="section">Areas</h5>

<div class="divider"></div>

<div class="row">

<div class="col s12 m6 card-panel">

<asp:GridView ID="gvAreas" OnSelectedIndexChanged="gvAreas\_SelectedIndexChanged" AutoGenerateColumns="false" runat="server">

<Columns>

<asp:BoundField DataField="area" HeaderText="Area" />

<asp:ButtonField ButtonType="Link" CommandName="Select" Text="Eliminar"/>

</Columns>

</asp:GridView>

</div>

<div class="col s12 m6">

<div class="row">

<div class="input-field col s12">

<asp:Label Text="Nombre del area" AssociatedControlID="txtArea" runat="server" />

<asp:TextBox runat="server" required ="required" ID="txtArea"/>

</div>

</div>

<div class="row">

<div class="col s12">

<asp:Button Text="Agregar area" ID="btnAgregar" CssClass="btn right waves-effect waves-light white-text" runat="server" OnClick="btnAgregar\_Click" />

</div>

</div>

</div>

</div>

</div>

</asp:Content>

## Areas: CS

using System;

using System.Collections.Generic;

using System.Linq;

using System.Web;

using System.Web.UI;

using System.Web.UI.WebControls;

namespace SICAP

{

public partial class areas : System.Web.UI.Page

{

SICAP.Modelos.Usuario usu;

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

{

usu = new SICAP.Modelos.Usuario();

gvAreas.DataSource = usu.traerAreas();

gvAreas.DataBind();

}

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

{

usu.AgregarArea(txtArea.Text.Trim());

ScriptManager.RegisterClientScriptBlock(this, this.GetType(), "mensaje",

"alert('Area agregada correctamente'); location.href='./areas.aspx'", true);

}

protected void gvAreas\_SelectedIndexChanged(object sender, EventArgs e)

{

GridViewRow fila = gvAreas.SelectedRow;

if (!usu.areaEnUso(fila.Cells[0].Text))

{

usu.eliminarArea(fila.Cells[0].Text);

ScriptManager.RegisterClientScriptBlock(this, this.GetType(), "mensaje",

"alert('Area eliminada correctamente'); location.href='./areas.aspx'", true);

}

else

{

ScriptManager.RegisterClientScriptBlock(this, this.GetType(), "mensaje",

"alert('El area se encuentra en uso'); location.href='./areas.aspx'", true);

}

}

}

}

## Avance.aspx

<%@ Page Title="" Language="C#" MasterPageFile="~/admin.Master" AutoEventWireup="true" CodeBehind="avance.aspx.cs" Inherits="SICAP.avance" %>

<asp:Content ID="Content1" ContentPlaceHolderID="head" runat="server">

</asp:Content>

<asp:Content ID="Content2" ContentPlaceHolderID="ContentPlaceHolder1" runat="server">

<div class="contenido animated fadeIn card-panel">

<h5 class="section">Registrar nuevo avance</h5>

<div class="divider"></div>

<div class="row">

<div class="input-field col s12">

<asp:Label Text="Nombre del avance" AssociatedControlID="txtNombreAvance" runat="server" />

<asp:TextBox required ="required" runat="server" ID="txtNombreAvance" />

</div>

</div>

<div class="row">

<div class="input-field col s12 m3">

<asp:Label Text="Fecha de Registro" runat="server" />

<asp:TextBox runat="server" Enabled="false" TextMode="Date" ID="txtFecha"/>

</div>

<div class="input-field col s12 m4">

<asp:Label Text="Activdad" runat="server" />

<asp:DropDownList ID="dllActividades" runat="server"></asp:DropDownList>

</div>

</div>

<div class="row">

<div class="input-field col s12">

<asp:Label Text="Observaciones" AssociatedControlID="txtObservaciones" runat="server" />

<asp:TextBox required="required" runat="server" ID="txtObservaciones" />

</div>

</div>

<h5 class="section">Cargar Archivos</h5>

<div class="row">

<div class="file-field input-field col s12 valign-wrapper">

<div class="btn">

<span>Seleccionar archivo</span>

<asp:FileUpload ID="fuArchivos" runat="server" />

</div>

<div class="file-path-wrapper">

<input class="file-path validate" type="text" placeholder="adjuntar">

</div>

</div>

</div>

<div class="row">

<div class="col s12">

<p class="red-text">\* Nota: no se pueden subir archvivos mayores 4MB</p>

</div>

</div>

<div class="row">

<div class="col s12">

<asp:Button Text="Guardar Avance" CssClass="btn right" ID="btnGuardar" OnClick="btnGuardar\_Click" runat="server" />

</div>

</div>

</div>

</asp:Content>

## Avance CS

using System;

using System.Collections.Generic;

using System.IO;

using System.Linq;

using System.Web;

using System.Web.UI;

using System.Web.UI.WebControls;

namespace SICAP

{

public partial class avance : System.Web.UI.Page

{

SICAP.Modelos.Avance avan;

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

{

if(Request.Params["id\_proyecto"] == null)

{

Response.Redirect("misProyectos.aspx");

}

if (!IsPostBack)

{

txtFecha.Text = DateTime.Today.ToString("yyyy-MM-dd");

avan = new SICAP.Modelos.Avance();

avan.id\_proyecto = int.Parse(Request.Params["id\_proyecto"]);

foreach(string actividad in avan.getActividades())

{

dllActividades.Items.Add(actividad);

}

}

}

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

{

avan = new SICAP.Modelos.Avance();

avan.NombreAvance = txtNombreAvance.Text.Trim();

avan.observaciones = txtObservaciones.Text;

avan.id\_proyecto = int.Parse(Request.Params["id\_proyecto"]);

avan.id\_usuario = (int)(Session["id\_usuario"]);

avan.id\_actividad = int.Parse(char.ToString(dllActividades.SelectedItem.Text[0]));

if (fuArchivos.HasFile)

{

string[] extension = fuArchivos.FileName.Split('.');

string ruta = Server.MapPath(".") + "./Documentos/" + extension[1];

if (!Directory.Exists(ruta))

{

Directory.CreateDirectory(ruta);

}

if (SICAP.Modelos.Avance.validarDocumento(fuArchivos.FileName)) // probar esta parte

{

ruta += "./" + fuArchivos.FileName;

fuArchivos.SaveAs(ruta);

ruta = "~/Documentos/" + extension[1] + "/" + fuArchivos.FileName;

avan.guardar(ruta);

ScriptManager.RegisterClientScriptBlock(this, this.GetType(), "mensaje",

string.Format("alert('avance registrado'); location.href='./avances.aspx?id\_proyecto={0}'", avan.id\_proyecto), true);

}

else

{

ScriptManager.RegisterClientScriptBlock(this, this.GetType(), "mensaje",

string.Format("alert('El archivo no es formato admnitido'); ", avan.id\_proyecto), true);

}

}

else

{

avan.guardar();

ScriptManager.RegisterClientScriptBlock(this, this.GetType(), "mensaje",

string.Format("alert('avance registrado'); location.href='./avances.aspx?id\_proyecto={0}'", avan.id\_proyecto), true);

}

}

}

}

## Avances.aspx

<%@ Page Title="" Language="C#" MasterPageFile="~/admin.Master" AutoEventWireup="true" CodeBehind="avances.aspx.cs" Inherits="SICAP.avances" %>

<asp:Content ID="Content1" ContentPlaceHolderID="head" runat="server">

</asp:Content>

<asp:Content ID="Content2" ContentPlaceHolderID="ContentPlaceHolder1" runat="server">

<div class="contenido card-panel animated fadeIn">

<h6>Avances del proyecto</h6>

<h4><asp:Label ID="lblNombreProyecto" Text="Nombre del proyecto aqui" runat="server"></asp:Label></h4>

<div class="divider"></div>

<div class="row">

<div class="col s12">

<asp:GridView runat="server" ID="gvAvances" AutoGenerateColumns="false" CssClass="responsive-table">

<Columns>

<asp:BoundField DataField="ID" HeaderText="ID" />

<asp:BoundField DataField="avance" HeaderText="Avance" />

<asp:BoundField DataField="observaciones" HeaderText="Observaciones" />

<asp:BoundField DataField="actividad" HeaderText="Actividad" />

<asp:BoundField DataField="fecha\_registro" HeaderText="Fecha De Registro" />

<asp:BoundField DataField="nombre" HeaderText="Nombre" />

<asp:HyperLinkField Text="Ver Avance" DataNavigateUrlFields="ID" DataNavigateUrlFormatString="perfilAvance.aspx?ID={0}" />

</Columns>

</asp:GridView>

</div>

</div>

</div>

<div class="fixed-action-btn">

<asp:HyperLink CssClass="btn-floating btn-large tooltipped" NavigateUrl="#" ID="hlAgregarAvance" runat="server" data-position="left" data-tooltip="Agregar Avance">

<i class="large material-icons">add</i>

</asp:HyperLink>

</div>

</asp:Content>

## Avances CS

using System;

using System.Collections.Generic;

using System.Linq;

using System.Web;

using System.Web.UI;

using System.Web.UI.WebControls;

namespace SICAP

{

public partial class avances : System.Web.UI.Page

{

SICAP.Modelos.Avance act;

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

{

if (Request.Params["id\_proyecto"] == null)

{

Response.Redirect("proyectos.aspx");

}

if (!IsPostBack)

{

SICAP.Modelos.Proyecto pro = new SICAP.Modelos.Proyecto();

pro.id\_proyecto = int.Parse(Request.Params["id\_proyecto"]);

pro.cargarDatos();

lblNombreProyecto.Text = pro.proyecto;

SICAP.Modelos.Avance avan = new SICAP.Modelos.Avance();

avan.id\_proyecto = pro.id\_proyecto;

gvAvances.DataSource = avan.traerAvances();

gvAvances.DataBind();

hlAgregarAvance.NavigateUrl = "avance.aspx?id\_proyecto=" + pro.id\_proyecto;

if (pro.estatus != "Activo")

{

hlAgregarAvance.Enabled = false;

hlAgregarAvance.CssClass = "btn-floating btn-large tooltipped disabled";

}

avan.id\_usuario = (int)(Session["id\_usuario"]);

if (!avan.pertenenAlproyecto())

{

hlAgregarAvance.Enabled = false;

hlAgregarAvance.CssClass = "btn-floating btn-large tooltipped disabled";

}

}

}

}

}

## Default.aspx

<%@ Page Language="C#" AutoEventWireup="true" CodeBehind="default.aspx.cs" Inherits="SICAP.\_default" %>

<!DOCTYPE html>

<html xmlns="http://www.w3.org/1999/xhtml">

<head runat="server">

<meta http-equiv="Content-Type" content="text/html; charset=utf-8"/>

<meta name="viewport" content="width=device-width, initial-scale=1.0"/>

<title>Inicio</title>

<link href="Recursos/materialize.css" rel="stylesheet" />

<link href="Recursos/material-icons.css" rel="stylesheet" />

<style>

form {

position: fixed;

top: 0;

left:0;

right:0;

bottom:0;

display: flex;

justify-content:center;

align-items:center;

}

.fm {

padding:24px;

}

.fx {

padding:0px 24px;

}

</style>

</head>

<body class="grey lighten-2">

<form id="form1" runat="server">

<div class="container">

<div class="row">

<div class="col s12 m4"></div>

<div class="card-panel col s12 m4">

<div class="row fm">

<div class="col s12 center-align">

<img class="responsive-img" src="Recursos/logoSICAP.png" />

</div>

</div>

<div class="row fm">

<div class="input-field col s12">

<asp:Label Text="Correo Electronico" AssociatedControlID="txtCorreo" runat="server" />

<asp:TextBox runat="server" ID="txtCorreo" required="required" TextMode="Email"/>

</div>

<div class="input-field col s12">

<asp:Label Text="Contraseña" AssociatedControlID="txtContrasena" runat="server" />

<asp:TextBox runat="server" ID="txtContrasena" required="required" TextMode="Password"/>

</div>

<div class="col s12">

<asp:Button Text="Iniciar" ID="btnIniciar" CssClass="btn blue" runat="server" OnClick="btnIniciar\_Click1" />

</div>

</div>

<div class="row fx">

<div class="col s12">

<a href="./recover.aspx">recuperar contraseña</a>

</div>

</div>

</div>

</div>

</div>

</form>

<script src="Recursos/materialize.js"></script>

<script>

document.addEventListener('DOMContentLoaded', function () {

M.AutoInit();

});

</script>

</body>

</html>

## Default CS

using System;

using System.Collections.Generic;

using System.Linq;

using System.Web;

using System.Web.UI;

using System.Web.UI.WebControls;

namespace SICAP

{

public partial class \_default : System.Web.UI.Page

{

SICAP.Modelos.Usuario usu;

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

{

usu = new Modelos.Usuario();

}

protected void btnIniciar\_Click1(object sender, EventArgs e)

{

usu.email = txtCorreo.Text.Trim();

usu.contrasena = txtContrasena.Text;

if (usu.validarAcceso())

{

Session["id\_usuario"] = usu.id\_usuario;

Session["nombre"] = usu.nombre;

Session["paterno"] = usu.paterno;

Session["materno"] = usu.materno;

Session["ruta"] = usu.ruta;

Session["rol"] = usu.rol;

if (usu.rol == 1)

{

Response.Redirect("proyectos.aspx");

}

else

{

Response.Redirect("misProyectos.aspx");

}

}

else

{

ScriptManager.RegisterClientScriptBlock(this, this.GetType(), "mensaje",

"alert('Usuario y/o Contraseña incorrectos')", true);

}

}

}

}

## MisProyectos.aspx

<%@ Page Title="" Language="C#" MasterPageFile="~/admin.Master" AutoEventWireup="true" CodeBehind="misProyectos.aspx.cs" Inherits="SICAP.misProyectos" %>

<asp:Content ID="Content1" ContentPlaceHolderID="head" runat="server">

</asp:Content>

<asp:Content ID="Content2" ContentPlaceHolderID="ContentPlaceHolder1" runat="server">

<div class="contenido card-panel animated fadeIn">

<h5 class="section">Mis proyectos</h5>

<div class="divider"></div>

<div class="row">

<div class="col s12">

<asp:GridView ID="gvMisProyectos" AutoGenerateColumns="false" runat="server">

<Columns>

<asp:BoundField DataField="id\_proyecto" HeaderText ="ID" />

<asp:BoundField DataField="proyecto" HeaderText="Proyecto" />

<asp:BoundField DataField ="fecha\_registro" HeaderText="Registro" />

<asp:BoundField DataField ="fecha\_inicio" HeaderText="Inicio" />

<asp:BoundField DataField ="fecha\_final" HeaderText="Termino" />

<asp:HyperLinkField DataNavigateUrlFields="id\_proyecto" DataNavigateUrlFormatString="perfilProyecto.aspx?id\_proyecto={0}" Text="ver datos" />

</Columns>

</asp:GridView>

</div>

</div>

</div>

</asp:Content>

## MisProyectos CS

using System;

using System.Collections.Generic;

using System.Linq;

using System.Web;

using System.Web.UI;

using System.Web.UI.WebControls;

namespace SICAP

{

public partial class misProyectos : System.Web.UI.Page

{

SICAP.Modelos.Usuario usu;

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

{

if (Session["id\_usuario"] == null)

{

Response.Redirect("default.aspx");

}

if (!IsPostBack)

{

var pro = new SICAP.Modelos.Proyecto();

usu = new SICAP.Modelos.Usuario();

usu.id\_usuario = (int)(Session["id\_usuario"]);

gvMisProyectos.DataSource = usu.traerMisProyectos();

gvMisProyectos.DataBind();

foreach (GridViewRow fila in gvMisProyectos.Rows)

{

pro.id\_proyecto = int.Parse(fila.Cells[0].Text);

pro.fecha\_final = fila.Cells[4].Text;

if (pro.expirado())

{

pro.concluir();

}

}

gvMisProyectos.DataSource = usu.traerMisProyectos();

gvMisProyectos.DataBind();

}

}

}

}

## PerfilAvance.aspx

<%@ Page Title="" Language="C#" MasterPageFile="~/admin.Master" AutoEventWireup="true" CodeBehind="perfilAvance.aspx.cs" Inherits="SICAP.perfilAvance" %>

<asp:Content ID="Content1" ContentPlaceHolderID="head" runat="server">

</asp:Content>

<asp:Content ID="Content2" ContentPlaceHolderID="ContentPlaceHolder1" runat="server">

<div class="contenido animated fadeIn card-panel ">

<div class="row">

<div class="col s12">

<label>Nombre del Avance</label>

<h5 class="section"><asp:Label Text="" ID="lbAvance" runat="server" /></h5>

<div class="divider"></div>

</div>

</div>

<div class="row">

<div class="col s12 m4">

<label>Fecha de registro</label>

<h6><asp:Label Text="" ID="lblFechaRegistro" runat="server" /></h6>

</div>

<div class="col s12 m4">

<label>Propietario</label>

<p><asp:Label Text="" ID="lblPropietario" runat="server" /></p>

</div>

</div>

<div class="row">

<div class="col s12">

<label>Observaciones</label>

<p><asp:Label Text="" ID="lbObservaciones" runat="server" /></p>

</div>

</div>

<div class="row">

<div class="col s12">

<h6><asp:Label Text="No se adjunto ningún archivo" ID ="lblArchivo" runat="server" /></h6>

<asp:HyperLink Text="Descargar Archivo" ID="hlDescargar" download CssClass="btn" runat="server" />

</div>

</div>

</div>

</asp:Content>

## PerfilAvance CS

using System;

using System.Collections.Generic;

using System.Linq;

using System.Web;

using System.Web.UI;

using System.Web.UI.WebControls;

namespace SICAP

{

public partial class perfilAvance : System.Web.UI.Page

{

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

{

if(Request.Params["ID"] == null)

{

Response.Redirect("misProyectos.aspx");

}

if (!IsPostBack)

{

SICAP.Modelos.Avance avan = new SICAP.Modelos.Avance(int.Parse(Request.Params["ID"]));

lbAvance.Text = avan.NombreAvance;

lbObservaciones.Text = avan.observaciones;

lblFechaRegistro.Text = avan.fecha\_registro;

lblPropietario.Text = avan.usuario;

string nombreArchivo = avan.rutaDoc.Split('/').Last();

lblArchivo.Text = "Archivo: " + nombreArchivo;

if(avan.rutaDoc == "")

{

hlDescargar.CssClass = "btn disabled";

lblArchivo.Text = "No se adjunto ningún archivo";

}

else

{

hlDescargar.NavigateUrl = avan.rutaDoc;

}

}

}

}

}

## PerfilProyecto.aspx

<%@ Page Title="" Language="C#" MasterPageFile="~/admin.Master" AutoEventWireup="true" CodeBehind="perfilProyecto.aspx.cs" Inherits="SICAP.perfilProyecto" %>

<asp:Content ID="Content1" ContentPlaceHolderID="head" runat="server">

</asp:Content>

<asp:Content ID="Content2" ContentPlaceHolderID="ContentPlaceHolder1" runat="server">

<div class="contenido card-panel animated fadeIn">

<h5 class="section">Datos del proyecto</h5>

<div class="divider"></div>

<div class="section"></div>

<div class="row">

<div class="col s12">

<asp:LinkButton ID="btnAvances" OnClick="btnAvances\_Click" CssClass="btn right" Text="Avances" runat="server" />

<asp:HyperLink NavigateUrl="#modal" ID="hlEliminarProyecto" Text="Eliminar Proyecto" CssClass="btn-flat red left modal-trigger" runat="server" />

</div>

</div>

<div class="row">

<div class="input-field col s12 m4 ">

<asp:Label Text="Estatus del Proyecto" ID="lblEstatus" runat="server" />

<asp:DropDownList AutoPostBack="true" OnSelectedIndexChanged="dllEstatus\_SelectedIndexChanged" ID="dllEstatus" runat="server" Enabled="False">

<asp:ListItem Text="text" Selected="True" disabled/>

<asp:ListItem Text="Activo" Value="1" />

<asp:ListItem Text="Concluido" Value="2" />

<asp:ListItem Text="Cancelado" Value="3" />

</asp:DropDownList>

</div>

<div class="col s12 m4">

<div class="section"></div>

<asp:LinkButton ID="btnEstatus" Text="Modificar estatus" CssClass="btn" runat="server" OnClick="btnEstatus\_Click" />

</div>

</div>

<div class="row">

<div class="input-field col s12 m4">

<p><asp:Label Text="\*Los proyectos expirados no se pueden Activar" Visible="false" ID="lblMensaje" runat="server" /></p>

</div>

</div>

<div class="row">

<div class="col s12">

<h5><asp:Label ID="lblNombreProyecto" runat="server" Text=""></asp:Label></h5>

</div>

</div>

<div class="row">

<div class="input-field col s12 m3">

<asp:Label ID="lblFechaInicio" runat="server" Text="Fecha Inicio"></asp:Label>

<asp:TextBox ID="txtFechaInicio" Enabled="false" TextMode="Date" runat="server"></asp:TextBox>

</div>

<div class="input-field col s12 m3">

<asp:Label ID="lblFechaFinal" runat="server" Text="Fecha Final"></asp:Label>

<asp:TextBox ID="txtFechaFinal" Enabled="false" runat="server" TextMode="Date"></asp:TextBox>

</div>

</div>

<div class="row">

<div class="col s12">

<h5 class="section">Observaciones</h5>

<p style="text-align: justify;">

<asp:Label ID="lblObservaciones" runat="server" Text="Observaciones"></asp:Label>

</p>

</div>

</div>

<div class="row">

<div class="col s12 m6">

<h5>Usuarios Selecionados</h5>

<asp:ListBox ID="lbxUsuarios" CssClass="browser-default fixSelect z-depth-2" runat="server"></asp:ListBox>

</div>

<div class="col s12 m2"></div>

</div>

<div class="row">

<div class="col s12">

<h5 class="section">Actividades</h5>

<div class="divider"></div>

<asp:GridView ID="gvActividades" CssClass="responsive-table fixTablaActividades" runat="server"></asp:GridView>

</div>

</div>

</div>

<div id="modal" class="modal">

<div class="modal-content">

<h4 class="section">¿Seguro que desea eliminar el proyecto?</h4>

<div class="divider"></div>

<p class="section">Todos los avances y actvidades del proyecto se perderan por completo</p>

</div>

<div class="modal-footer">

<asp:LinkButton CssClass="modal-close waves-effect waves-red btn-flat" OnClick="eliminarBoton\_Click" Text="Eliminar" ID="eliminarBoton" runat="server" />

</div>

</div>

<script src="Utelirias/PerfilProyecto.js"></script>

</asp:Content>

## PerfilProyecto CS

using System;

using System.Collections.Generic;

using System.Linq;

using System.Web;

using System.Web.UI;

using System.Web.UI.WebControls;

namespace SICAP

{

public partial class perfilProyecto : System.Web.UI.Page

{

SICAP.Modelos.Proyecto proyect;

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

{

if (Request.Params["id\_proyecto"] == null)

{

Response.Redirect("proyectos.aspx");

}

if ((int)(Session["rol"]) == 2)

{

dllEstatus.Visible = false;

btnEstatus.Visible = false;

lblEstatus.Visible = false;

eliminarBoton.Visible = false;

hlEliminarProyecto.Visible = false;

}

if (!IsPostBack)

{

proyect = new SICAP.Modelos.Proyecto();

proyect.id\_proyecto = int.Parse(Request.Params["id\_proyecto"]);

proyect.cargarDatos();

lblNombreProyecto.Text = proyect.proyecto;

lblObservaciones.Text = proyect.observaciones;

txtFechaInicio.Text = proyect.fecha\_inicio;

txtFechaFinal.Text = proyect.fecha\_final;

dllEstatus.SelectedItem.Text = proyect.estatus;

SICAP.Modelos.Actividad act = new SICAP.Modelos.Actividad();

gvActividades.DataSource = act.traerActividades(proyect.id\_proyecto);

gvActividades.DataBind();

for (int i = 0; i < gvActividades.Rows.Count; i++)

{

act.id\_actividad = int.Parse(gvActividades.Rows[i].Cells[0].Text);

if (act.expirado())

{

act.concluir();

}

}

gvActividades.DataSource = act.traerActividades(proyect.id\_proyecto);

gvActividades.DataBind();

if (proyect.expirado())

{

proyect.concluir();

dllEstatus.SelectedItem.Text = proyect.estatus;

btnEstatus.Enabled = false;

btnEstatus.CssClass = "btn disabled";

btnEstatus.Text = "Proyecto expirado";

lblMensaje.Visible = true;

}

}

if (lbxUsuarios.Items.Count == 0)

{

foreach(string user in proyect.traerUsuariosLista())

{

lbxUsuarios.Items.Add(user);

}

}

}

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

{

if(btnEstatus.Text != "Cancelar modificacion")

{

dllEstatus.Enabled = true;

btnEstatus.Text = "Cancelar modificacion";

btnEstatus.CssClass = "btn red";

}

else

{

dllEstatus.Enabled = false;

btnEstatus.Text = "Modificar estatus";

btnEstatus.CssClass = "btn";

}

}

protected void dllEstatus\_SelectedIndexChanged(object sender, EventArgs e)

{

proyect = new SICAP.Modelos.Proyecto();

proyect.id\_proyecto = int.Parse(Request.Params["id\_proyecto"]);

proyect.cambiarEstatus(int.Parse(dllEstatus.SelectedItem.Value));

dllEstatus.Enabled = false;

btnEstatus.CssClass = "btn";

btnEstatus.Text = "Modificar estatus";

}

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

{

string pagina = string.Format("avances.aspx?id\_proyecto= {0}",Request.Params["id\_proyecto"]);

Response.Redirect(pagina);

}

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

{

SICAP.Modelos.Proyecto pro = new SICAP.Modelos.Proyecto();

pro.id\_proyecto = int.Parse(Request.Params["id\_proyecto"]);

if (!pro.eliminarProyecto())

{

ScriptManager.RegisterClientScriptBlock(this, this.GetType(), "mensaje",

"alert('Proyeco eliminado'); location.href='./proyectos.aspx'", true);

}

else

{

ScriptManager.RegisterClientScriptBlock(this, this.GetType(), "mensaje",

"alert('El proyecto se encuentra actualmente activo'); ", true);

}

}

}

}

## Proyecto.aspx

<%@ Page Title="" Language="C#" MasterPageFile="~/admin.Master" AutoEventWireup="true" CodeBehind="proyecto.aspx.cs" Inherits="SICAP.proyecto" %>

<asp:Content ID="Content1" ContentPlaceHolderID="head" runat="server">

</asp:Content>

<asp:Content ID="Content2" ContentPlaceHolderID="ContentPlaceHolder1" runat="server">

<div class="contenido card-panel animated fadeIn">

<h3 class="section">Crear un nuevo proyecto</h3>

<div class="divider"></div>

<div class="section"></div>

<div class="row">

<div class="input-field col s12">

<asp:Label Text="Nombre del proyecto" AssociatedControlID="txtNombre" runat="server" />

<asp:TextBox runat="server" ID="txtNombre" />

</div>

</div>

<div class="row">

<div class="input-field col s12 m2">

<asp:Label Text="Fecha de inicio" runat="server" />

<asp:TextBox ID="txtFechaInicialProyecto" AutoPostBack="true" OnTextChanged="txtFechaInicialProyecto\_TextChanged" required="required" CssClass="validate" runat="server" TextMode="Date" />

</div>

<div class="input-field col s12 m3">

<asp:Label Text="Fecha de Termino" runat="server" />

<asp:TextBox ID="txtFechaFinalProyecto" AutoPostBack="true" OnTextChanged="txtFechaFinalProyecto\_TextChanged" required="required" CssClass="validate" runat="server" TextMode="Date" />

</div>

<div class="col s12 m7">

</div>

</div>

<div class="row">

<div class="input-field col s12">

<asp:Label Text="Obeservaciones" AssociatedControlID="txtObservaciones" runat="server" />

<asp:TextBox ID="txtObservaciones" required="required" CssClass="materialize-textarea validate" TextMode ="MultiLine" runat="server" />

</div>

</div>

<%-- Es la parte de seleccion de usuario --%>

<h5 class="section">Usuarios</h5>

<div class="divider"></div>

<div class="section"></div>

<div class="row">

<div class="col s12 m5">

<asp:ListBox CssClass="browser-default fixSelect z-depth-1 section" ID="lbxUsuarios" SelectionMode="Multiple" runat="server">

</asp:ListBox>

</div>

<div class="col s12 m2 center-align">

<div class="section">

<asp:LinkButton CssClass="btn" ID="btnPasar" runat="server" OnClick="btnPasar\_Click" ><i class="material-icons">arrow\_forward</i></asp:LinkButton>

</div>

<div class="section">

<asp:LinkButton CssClass="btn" ID="btnRegresar" runat="server" OnClick="btnRegresar\_Click"><i class="material-icons">arrow\_back</i></asp:LinkButton>

</div>

</div>

<div class="col s12 m5">

<asp:ListBox CssClass="browser-default fixSelect z-depth-1 section" ID="lbxUsuariosSeleccionados" SelectionMode="Multiple" runat="server">

</asp:ListBox>

</div>

</div>

<div class="row">

<div class="col s12">

<p>

\*Puede seleccionar varios usuarios manteniendo presionado Ctrl y dando clic en los usuarios

</p>

</div>

</div>

<%-- aqui viene las actividades // sera util,o bien, se cancela todo --%>

<h5 class="section">Actividades</h5>

<div class="divider"></div>

<div class="row">

<div class="col s12">

<ul class="collapsible">

<li>

<div class="collapsible-header"><i class="material-icons">add</i>Agregar Actividad</div>

<div class="collapsible-body">

<asp:Label Text="\* No se ha selecionado fecha de termino" CssClass="red-text" ID="lblActividadesSinFecha" runat="server" />

<div class="row">

<div class="input-field col s12 m9">

<asp:Label Text="Nombre de la actividad" runat="server" />

<asp:TextBox ID="txtNombreActividad" placeholder="Escriba el nombre la actividad" runat="server" />

</div>

<div class="input-field col s12 m3">

<asp:Label Text="Fecha de entrega" runat="server" />

<asp:TextBox ID="txtfechaEntregaActividad" Enabled="false" TextMode="Date" runat="server" />

</div>

</div>

<div class="row">

<div class="col s12">

<asp:Label Text="Observaciones" AssociatedControlID="txtObservacionesActividad" runat="server" />

<asp:TextBox ID="txtObservacionesActividad" TextMode="MultiLine" placeholder="observciones" CssClass="materialize-textarea" runat="server" />

</div>

</div>

<div class="row">

<asp:LinkButton runat="server" ID="btnAgregarActividad" OnClick="btnAgregarActividad\_Click" >Agregar Actividad<i class="material-icons left">add</i></asp:LinkButton>

</div>

</div>

</li>

</ul>

</div>

</div>

<div class="row">

<div class="col s12">

<asp:ListBox ID="lbxActividades" CssClass="browser-default fixSelect z-depth-2" runat="server"></asp:ListBox>

</div>

<div class ="col s12">

<br />

<asp:LinkButton Text="Eliminar Actividad" CssClass="btn-flat red white-text section" ID = "btnEliminarActividad" OnClick="btnEliminarActividad\_Click" runat="server" />

</div>

</div>

<div class="row">

<div class="section"></div>

<asp:Button ID="btnCrearProyecto" CssClass="btn right" runat="server" Text="Crear Proyecto" OnClick="btnCrearProyecto\_Click" />

</div>

</div>

<%-- <script src="Utelirias/Proyecto.js"></script>--%>

</asp:Content>

## Proyecto CS

using System;

using System.Data;

using System.Collections.Generic;

using System.Linq;

using System.Web;

using System.Web.UI;

using System.Web.UI.WebControls;

using SICAP;

namespace SICAP

{

public partial class proyecto : System.Web.UI.Page

{

SICAP.Modelos.Usuario usu;

SICAP.Modelos.Proyecto proyect;

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

{

usu = new SICAP.Modelos.Usuario();

proyect = new SICAP.Modelos.Proyecto();

if (!IsPostBack)

{

txtFechaInicialProyecto.Text = DateTime.Today.ToString("yyyy-MM-dd");

txtFechaInicialProyecto.Attributes.Add("min", DateTime.Today.ToString("yyyy-MM-dd"));

txtFechaFinalProyecto.Attributes.Add("min", DateTime.Today.ToString("yyyy-MM-dd"));

txtfechaEntregaActividad.Attributes.Add("min", DateTime.Today.ToString("yyyy-MM-dd"));

DataTable usuarios = usu.traerUsuarios();

foreach (DataRow usuario in usuarios.Rows)

{

/// aqui es donde lo concateno tengo que hacer forech entonces??

int id = (int)(usuario["id\_usuario"]);

string nombre = usuario["nombre"].ToString();

string paterno = usuario["paterno"].ToString();

string materno = usuario["materno"].ToString();

string especialidad = usuario["especialidad"].ToString();

string area = usuario["area"].ToString();

string item = id + " - " + " " + nombre + " " + paterno + " " + materno + " - " + especialidad;

lbxUsuarios.Items.Add(item);

}

}

}

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

{

while (lbxUsuarios.GetSelectedIndices().Length > 0)

{

lbxUsuariosSeleccionados.Items.Add(lbxUsuarios.SelectedItem);

lbxUsuarios.Items.Remove(lbxUsuarios.SelectedItem);

}

}

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

{

while (lbxUsuariosSeleccionados.GetSelectedIndices().Length > 0)

{

lbxUsuarios.Items.Add(lbxUsuariosSeleccionados.SelectedItem);

lbxUsuariosSeleccionados.Items.Remove(lbxUsuariosSeleccionados.SelectedItem);

}

}

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

{

if (!string.IsNullOrEmpty(txtNombreActividad.Text) && !string.IsNullOrEmpty(txtObservaciones.Text))

{

DateTime incial = DateTime.Parse(txtFechaInicialProyecto.Text);

DateTime final = DateTime.Parse(txtFechaFinalProyecto.Text);

DateTime fechaActividad = DateTime.Parse(txtfechaEntregaActividad.Text);

if (DateTime.Compare(fechaActividad, final) <= 0 && DateTime.Compare(fechaActividad, incial) >= 0)

{

string actividad = txtNombreActividad.Text.Trim() + "/" + txtObservacionesActividad.Text.Trim() + "/" + txtfechaEntregaActividad.Text;

lbxActividades.Items.Add(actividad);

txtNombreActividad.Text = null;

txtObservacionesActividad.Text = null;

}

else

{

ScriptManager.RegisterClientScriptBlock(this, this.GetType(), "mensaje",

"alert('La fecha de entrega de la actividad esta fuera del plazo del proyecto');", true);

}

}

}

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

{

foreach(ListItem item in lbxActividades.Items) // con esto asignamos las actividades

{

string[] actividad = item.Text.Split('/');

proyect.actividades.Add(actividad);

}

foreach(ListItem item in lbxUsuariosSeleccionados.Items) // con esto asignamos los usuarios al proyecto

{

int usuario = int.Parse(char.ToString(item.Text[0]));

proyect.usuarios.Add(usuario);

}

proyect.proyecto = txtNombre.Text.Trim();

proyect.fecha\_inicio = txtFechaInicialProyecto.Text;

proyect.fecha\_final = txtFechaFinalProyecto.Text;

proyect.observaciones = txtObservaciones.Text;

if(lbxUsuariosSeleccionados.Items.Count > 0)

{

if(lbxActividades.Items.Count > 0)

{

DateTime incial = DateTime.Parse(proyect.fecha\_inicio);

DateTime final = DateTime.Parse(proyect.fecha\_final);

if(DateTime.Compare(incial,final) == 0)

{

ScriptManager.RegisterClientScriptBlock(this, this.GetType(), "mensaje",

"alert('Error: la fecha de incio y termino son identicas,favor de verificar las fechas');", true);

}

else if(DateTime.Compare(incial,final) > 0)

{

ScriptManager.RegisterClientScriptBlock(this, this.GetType(), "mensaje",

"alert('Error: la fecha de inicio sobrepasa a fecha de termino') location.href= './proyecto.aspx';", true);

}

else

{

proyect.guardar();

proyect.id\_proyecto = proyect.getID();

proyect.asignarUsuarios();

proyect.asignarActividades();

ScriptManager.RegisterClientScriptBlock(this, this.GetType(), "mensaje",

"alert('Proyecto Creado correctamente'); location.href='./proyectos.aspx'", true);

}

}

else

{

ScriptManager.RegisterClientScriptBlock(this, this.GetType(), "mensaje",

"alert('No se ha agregado ninguna actividad');", true);

}

}

else

{

ScriptManager.RegisterClientScriptBlock(this, this.GetType(), "mensaje",

"alert('No se ha selecionado ningun usuario');", true);

}

}

protected void txtFechaFinalProyecto\_TextChanged(object sender, EventArgs e)

{

txtfechaEntregaActividad.Attributes.Add("max",txtFechaFinalProyecto.Text);

txtfechaEntregaActividad.Enabled = true;

lblActividadesSinFecha.Visible = false;

}

protected void txtFechaInicialProyecto\_TextChanged(object sender, EventArgs e)

{

txtFechaFinalProyecto.Attributes.Add("min",txtFechaInicialProyecto.Text);

txtfechaEntregaActividad.Attributes.Add("min",txtFechaInicialProyecto.Text);

}

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

{

while (lbxActividades.GetSelectedIndices().Length > 0)

{

lbxActividades.Items.Remove(lbxActividades.SelectedItem);

}

}

}

}

## Proyectos.aspx

<%@ Page Title="" Language="C#" MasterPageFile="~/admin.Master" AutoEventWireup="true" CodeBehind="proyectos.aspx.cs" Inherits="SICAP.proyectos" %>

<asp:Content ID="Content1" ContentPlaceHolderID="head" runat="server">

</asp:Content>

<asp:Content ID="Content2" ContentPlaceHolderID="ContentPlaceHolder1" runat="server">

<div class="contenido card-panel animated fadeIn">

<div class="row">

<div class="col s12">

<h4 class="section">Proyectos</h4>

<div class="divider"></div>

</div>

</div>

<div class="row">

<div class="col s12 m1">

<p>Mostrar proyectos</p>

</div>

<div class="col s12 m4">

<asp:DropDownList ID="dllEstatus" runat="server" AutoPostBack="true" OnSelectedIndexChanged="dllEstatus\_SelectedIndexChanged">

<asp:ListItem Text="Activo" Value="1" />

<asp:ListItem Text="Concluido" Value="2" />

<asp:ListItem Text="Cancelado" Value="3" />

</asp:DropDownList>

</div>

</div>

<div class="row">

<div class="col s12">

<asp:GridView ID="gvProyectos" DataKeyNames="id\_proyecto" AutoGenerateColumns="false" runat="server">

<Columns>

<asp:BoundField DataField="id\_proyecto" HeaderText="ID" />

<asp:BoundField DataField="proyecto" HeaderText="Nombre del Proyecto" />

<asp:BoundField DataField="fecha\_registro" HeaderText="Fecha de Crecion" />

<asp:BoundField DataField="fecha\_inicio" HeaderText ="Incio" />

<asp:BoundField DataField="fecha\_final" HeaderText="Termino" />

<asp:BoundField DataField="estatus" HeaderText="Estatus" />

<asp:HyperLinkField DataNavigateUrlFields="id\_proyecto" DataNavigateUrlFormatString="perfilProyecto.aspx?id\_proyecto={0}" Text="ver datos" />

</Columns>

</asp:GridView>

</div>

</div>

</div>

</asp:Content>

## Proyectos CS

using System;

using System.Collections.Generic;

using System.Linq;

using System.Web;

using System.Web.UI;

using System.Web.UI.WebControls;

namespace SICAP

{

public partial class proyectos : System.Web.UI.Page

{

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

{

if (!IsPostBack)

{

SICAP.Modelos.Proyecto proyect = new SICAP.Modelos.Proyecto();

proyect.id\_estatus = int.Parse(dllEstatus.SelectedValue);

gvProyectos.DataSource = proyect.traerProyectos();

gvProyectos.DataBind();

foreach (GridViewRow fila in gvProyectos.Rows)

{

proyect.id\_proyecto = int.Parse(fila.Cells[0].Text);

proyect.fecha\_final = fila.Cells[4].Text;

if (proyect.expirado())

{

proyect.concluir();

}

}

gvProyectos.DataSource = proyect.traerProyectos();

gvProyectos.DataBind();

}

}

protected void dllEstatus\_SelectedIndexChanged(object sender, EventArgs e)

{

SICAP.Modelos.Proyecto proyect = new SICAP.Modelos.Proyecto();

proyect.id\_estatus = int.Parse(dllEstatus.SelectedValue);

gvProyectos.DataSource = proyect.traerProyectos();

gvProyectos.DataBind();

}

}

}

## Recover.aspx

<%@ Page Language="C#" AutoEventWireup="true" CodeBehind="recover.aspx.cs" Inherits="SICAP.recover" %>

<!DOCTYPE html>

<html xmlns="http://www.w3.org/1999/xhtml">

<head runat="server">

<meta http-equiv="Content-Type" content="text/html; charset=utf-8"/>

<title>recover</title>

<link href="Recursos/materialize.css" rel="stylesheet" />

<link href="Recursos/material-icons.css" rel="stylesheet" />

<style>

form {

position: fixed;

top: 0;

left:0;

right:0;

bottom:0;

display: flex;

justify-content:center;

align-items:center;

}

.fm {

padding:24px;

}

</style>

</head>

<body>

<form id="form1" runat="server">

<div class="container">

<div class="row">

<div class="col s12 m3"></div>

<div class="card-panel col s12 m6">

<div class="row fm">

<a href="./default.aspx" class="right">volver</a>

<h5 class="section">Recuperar contraseña</h5>

<div class="divider"></div>

<p class="section">

Ingresa la direccion de correo electronico con la que te encuentres registrado para que se te mande una nueva contraseña.

</p>

<div class="input-field">

<asp:Label Text="Correo electronico" AssociatedControlID="txtEmail" runat="server" />

<asp:TextBox ID="txtEmail" TextMode="Email" CssClass="activate" runat="server" />

</div>

<asp:Button Text="Enviar Correo" ID="btnRecuperar" CssClass="btn right" runat="server" OnClick="btnRecuperar\_Click" />

</div>

</div>

</div>

</div>

</form>

<script src="Recursos/materialize.js"></script>

<script>

document.addEventListener('DOMContentLoaded', function () {

M.AutoInit();

});

</script>

</body>

</html>

## Recover CS

using System;

using System.Collections.Generic;

using System.Linq;

using System.Web;

using System.Web.UI;

using System.Web.UI.WebControls;

namespace SICAP

{

public partial class recover : System.Web.UI.Page

{

SICAP.Modelos.Usuario usu;

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

{

usu = new SICAP.Modelos.Usuario();

}

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

{

usu.email = txtEmail.Text.Trim();

if (usu.correoExistente())

{

string contrasena = SICAP.Utelirias.GenerarPass.get();

string mensaje = "Hola " + usu.getNombreUsuario() + ", tu nueva contraseña es: <stronG>" + contrasena + "</strong>";

contrasena = SICAP.Utelirias.Encriptar.GetSHA1(contrasena);

usu.actualizarPass(contrasena);

SICAP.Utelirias.Correo.EnviarCorreo(usu.email, "nueva contraseña SICAP", mensaje);

ScriptManager.RegisterClientScriptBlock(this, this.GetType(), "mensaje",

"alert('La nueva contraseña se ha enviado a tu correo electronico'); location.href= './default.aspx'", true);

}

else

{

ScriptManager.RegisterClientScriptBlock(this, this.GetType(), "mensaje",

"alert('El correo no se encuentra registrado');", true);

}

}

}

}

## Registro.aspx

<%@ Page Title="" Language="C#" MasterPageFile="~/admin.Master" AutoEventWireup="true" CodeBehind="registro.aspx.cs" Inherits="SICAP.registro" %>

<asp:Content ID="Content1" ContentPlaceHolderID="head" runat="server">

</asp:Content>

<asp:Content ID="Content2" ContentPlaceHolderID="ContentPlaceHolder1" runat="server">

<div class="contenido card-panel animated fadeIn">

<div class="row">

<div class="col s12">

<h4 class="section">Registro</h4>

<div class="divider"></div>

</div>

</div>

<div class="row">

<asp:Image ID="imgPerfil" ImageUrl="Imagenes/default.png" CssClass="imagen responsive-img circle col s4 m1" runat="server"/>

<div class="file-field input-field col s12 m4">

<div class="btn blue">

<span>Seleccionar Foto</span>

<asp:FileUpload ID="fuPerfil" runat="server" />

</div>

<div class="file-path-wrapper">

<input class="file-path validate" type="text" placeholder="Favor de cargar una foto pata el perfil">

</div>

</div>

<asp:LinkButton ID="lbPerfil" CssClass="col s12 m2" OnClick="lbPerfil\_Click" runat="server">Cargar Foto</asp:LinkButton>

</div>

<div class="row">

<div class="input-field col s12 m4">

<i class="material-icons prefix">account\_circle</i>

<asp:TextBox ID="txtNombre" required ="required" CssClass="validate" runat="server" ></asp:TextBox>

<asp:Label ID="lblNombre" AssociatedControlID="txtNombre" runat="server" Text="Nombre"></asp:Label>

</div>

<div class="input-field col s12 m4">

<asp:TextBox ID="txtPaterno" runat="server" />

<asp:Label Text="Apellido Paterno" AssociatedControlID="txtPaterno" runat="server" />

</div>

<div class="input-field col s12 m4">

<asp:TextBox runat="server" ID="txtMaterno" required ="required" CssClass="validate"/>

<asp:Label Text="Apellido Materno" AssociatedControlID="txtMaterno" runat="server" />

</div>

</div>

<div class="row">

<div class="input-field col s12 m4">

<i class="material-icons prefix">email</i>

<asp:TextBox ID="txtEmail" TextMode="Email" CssClass="validate" required="required" runat="server"></asp:TextBox>

<asp:Label ID="lblEmail" AssociatedControlID="txtEmail" runat="server" Text="Correo Electronico"></asp:Label>

</div>

<div class="input-field col s12 m4">

<i class="material-icons prefix">phone</i>

<asp:TextBox ID="txtTelefono" TextMode="Phone" runat="server" required ="required" CssClass="validate"></asp:TextBox>

<asp:Label Text="Numero telefonico" AssociatedControlID="txtTelefono" runat="server" />

</div>

</div>

<div class="row">

<div class="input-field col s12 m4">

<asp:DropDownList ID="ddlRol" runat="server">

<asp:ListItem Text="Elija rol de usuario" disabled Selected />

<asp:ListItem Text="Administrador" />

<asp:ListItem Text="Usuario" />

</asp:DropDownList>

</div>

<div class="input-field col s12 m4">

<i class="material-icons prefix">vpn\_key</i>

<asp:TextBox ID="txtContrasena" CssClass="validate" TextMode="Password" required="required" runat="server"></asp:TextBox>

<asp:Label ID="lblContrasena" AssociatedControlID="txtContrasena" runat="server" Text="Contrasena"></asp:Label>

</div>

<div class="input-field col s12 m4">

<i class="material-icons prefix">vpn\_key</i>

<asp:TextBox ID="txtConfirmarContrasena" TextMode="Password" CssClass="validate" required="required" runat="server"></asp:TextBox>

<asp:Label Text="Confirmar contraseña" AssociatedControlID="txtConfirmarContrasena" runat="server" />

</div>

</div>

<div class="row">

<div class="input-field col s12 m4">

<i class="material-icons prefix">school</i>

<asp:TextBox ID="txtEspecialidad" CssClass="validate" required="required" runat="server"></asp:TextBox>

<asp:Label Text="Especialidad" AssociatedControlID="txtEspecialidad" runat="server" />

</div>

<div class="input-field col s12 m4">

<i class="material-icons prefix">supervised\_user\_circle</i>

<asp:DropDownList ID="ddlArea" runat="server">

<asp:ListItem Text="Elija area" disabled Selected />

</asp:DropDownList>

</div>

</div>

<div class="row">

<div class="section">

<asp:Button ID="btnRegisrar" CssClass="btn right" runat="server" Text="Registrar" OnClick="btnRegisrar\_Click" />

</div>

</div>

</div>

</asp:Content>

## Registro. CS

using SICAP.Modelos;

using System;

using System.Collections.Generic;

using System.Data;

using System.Linq;

using System.Web;

using System.Web.UI;

using System.Web.UI.WebControls;

namespace SICAP

{

public partial class registro : System.Web.UI.Page

{

SICAP.Modelos.Usuario usu;

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

{

usu = new SICAP.Modelos.Usuario();

if (ddlArea.Items.Count == 1)

{

foreach (DataRow row in usu.traerAreas().Rows)

{

ddlArea.Items.Add(row["area"].ToString());

}

}

}

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

{

usu.nombre = txtNombre.Text.Trim();

usu.paterno = txtPaterno.Text.Trim();

usu.materno = txtMaterno.Text.Trim();

usu.email = txtEmail.Text.Trim();

usu.especialidad = txtEspecialidad.Text.Trim();

usu.contrasena = txtContrasena.Text.Trim();

usu.telefono = txtTelefono.Text.Trim();

usu.ruta = imgPerfil.ImageUrl;

usu.area = usu.asignarArea(ddlArea.Text);

usu.rol = (ddlRol.Text == "Administrador") ? 1 : 2;

if (usu.correoExistente())

{

ScriptManager.RegisterClientScriptBlock(this, this.GetType(), "mensaje",

"alert('correo electronico ya registrado')", true);

}

else if (txtContrasena.Text.Trim() != txtConfirmarContrasena.Text.Trim())

{

ScriptManager.RegisterClientScriptBlock(this, this.GetType(), "mensaje",

"alert('las contraseñas no coinciden')", true);

}

else if (ddlRol.Text == "Elija rol de usuario" || ddlArea.Text == "Elija area")

{

ScriptManager.RegisterClientScriptBlock(this, this.GetType(), "mensaje",

"alert('Favor de completar todos los campos')", true);

}

else

{

usu.guardar();

ScriptManager.RegisterClientScriptBlock(this, this.GetType(), "mensaje",

"alert('usuario guardado correctamente'); location.href='./usuarios.aspx'", true);

}

}

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

{

if (fuPerfil.HasFile)

{

if (SICAP.Modelos.Usuario.validarExtensionImg(fuPerfil.FileName))

{

usu.ruta = "./Imagenes/" + fuPerfil.FileName;

fuPerfil.SaveAs(Server.MapPath(".") + usu.ruta);

imgPerfil.ImageUrl = usu.ruta;

}

else

{

ScriptManager.RegisterClientScriptBlock(this, this.GetType(), "mensaje",

"alert('La imagen no tiene un formato valido; PNG,JPG o GIF');", true);

}

}

else

{

ScriptManager.RegisterClientScriptBlock(this, this.GetType(), "mensaje",

"alert('No ha selecionado ningun archvio');", true);

}

}

}

}

## Usuario.aspx

<%@ Page Title="" Language="C#" MasterPageFile="~/admin.Master" AutoEventWireup="true" CodeBehind="usuario.aspx.cs" Inherits="SICAP.usuario" %>

<asp:Content ID="Content1" ContentPlaceHolderID="head" runat="server">

</asp:Content>

<asp:Content ID="Content2" ContentPlaceHolderID="ContentPlaceHolder1" runat="server">

<div class="contenido">

<div class="row">

<asp:HyperLink ID="hlDesUsu" NavigateUrl="#modalDeshabilitar" Text="Deshabilitar Usuario" CssClass="waves-effect waves-light btn-flat cyan white-text modal-trigger left" runat="server" />

<asp:HyperLink ID="hlHabUsu" Text="Habilitar Usuario" Visible = "false" NavigateUrl="#modalHabilitar" CssClass="waves-effect waves-light btn-flat green white-text modal-trigger right" runat="server" />

</div>

</div>

<div class="contenido card-panel animated fadeIn">

<div class="row">

<div class="col s12">

<asp:HyperLink Text="Volver Usuarios" NavigateUrl="~/usuarios.aspx" CssClass="left" runat="server" />

<asp:LinkButton Text="Modificar" ID="btnHabilitar" runat="server" CssClass="btn right" OnClick="btnHabilitar\_Click" />

<h4 class="section">Datos del usuario</h4>

<div class="divider"></div>

</div>

</div>

<div class="row">

<asp:Image ID="imgPerfil" ImageUrl="zImagenes/perflDefault.png" CssClass="imagen responsive-img circle col s4 m1" runat="server"/>

<div class="file-field input-field col s12 m4">

<div id="fuArchivo" class="btn blue">

<span>Seleccionar Foto</span>

<asp:FileUpload ID="fuPerfil" runat="server" Enabled ="false"/>

</div>

<div class="file-path-wrapper">

<input class="file-path validate" type="text" placeholder="Favor de cargar una foto pata el perfil">

</div>

</div>

<asp:LinkButton ID="lbPerfil" CssClass="col s12 m2" runat="server" OnClick="lbPerfil\_Click">Cargar Foto</asp:LinkButton>

</div>

<div class="row">

<div class="input-field col s12 m4">

<i class="material-icons prefix">account\_circle</i>

<asp:TextBox ID="txtNombre" Enabled="false" required ="required" CssClass="validate disabled" runat="server" ></asp:TextBox>

<asp:Label ID="lblNombre" AssociatedControlID="txtNombre" runat="server" Text="Nombre"></asp:Label>

</div>

<div class="input-field col s12 m4">

<asp:TextBox ID="txtPaterno" Enabled="false" required ="required" CssClass="validate" runat="server" />

<asp:Label Text="Apellido Paterno" AssociatedControlID="txtPaterno" runat="server" />

</div>

<div class="input-field col s12 m4">

<asp:TextBox runat="server" ID="txtMaterno" Enabled="false" required ="required" CssClass="validate"/>

<asp:Label Text="Apellido Materno" AssociatedControlID="txtMaterno" runat="server" />

</div>

</div>

<div class="row">

<div class="input-field col s12 m4">

<i class="material-icons prefix">email</i>

<asp:TextBox ID="txtEmail" Enabled="false" TextMode="Email" CssClass="validate" required="required" runat="server"></asp:TextBox>

<asp:Label ID="lblEmail" AssociatedControlID="txtEmail" runat="server" Text="Correo Electronico"></asp:Label>

</div>

<div class="input-field col s12 m4">

<i class="material-icons prefix">phone</i>

<asp:TextBox ID="txtTelefono" Enabled="false" TextMode="Phone" runat="server" required ="required" CssClass="validate"></asp:TextBox>

<asp:Label Text="Numero telefonico" AssociatedControlID="txtTelefono" runat="server" />

</div>

</div>

<div class="row">

<div class="input-field col s12 m4">

<asp:DropDownList Enabled="false" ID="ddlRol" runat="server">

<asp:ListItem Text="Elija rol de usuario" />

<asp:ListItem Text="Administrador" />

<asp:ListItem Text="Usuario" />

</asp:DropDownList>

</div>

<div class="input-field col s12 m8">

<asp:HyperLink NavigateUrl="#modal" Text="Cambiar Contraseña" ID="hlCambiarPass" CssClass="waves-effect waves-light btn modal-trigger" runat="server" />

</div>

</div>

<div class="row">

<div class="input-field col s12 m4">

<i class="material-icons prefix">school</i>

<asp:TextBox ID="txtEspecialidad" Enabled="false" CssClass="validate" required="required" runat="server"></asp:TextBox>

<asp:Label Text="Especialidad" AssociatedControlID="txtEspecialidad" runat="server" />

</div>

<div class="input-field col s12 m4">

<i class="material-icons prefix">supervised\_user\_circle</i>

<asp:DropDownList ID="ddlArea" Enabled="false" runat="server">

<asp:ListItem Text="Elija area" disabled="disabled" Selected />

</asp:DropDownList>

</div>

</div>

<div class="row">

<div class="section">

<asp:Button ID="btnRegisrar" Enabled="false" CssClass="btn right" runat="server" Text="Guardar" OnClick="btnRegisrar\_Click" />

</div>

</div>

<div class="row">

<div class="col s12">

<ul class="collapsible">

<li>

<div class="collapsible-header"><i class="material-icons">work</i>Proyectos en los que participa</div>

<div class="collapsible-body">

<div class="row">

<asp:GridView runat="server" AutoGenerateColumns ="false" CssClass="responsive-table" ID="gvProyectosUsu">

<Columns>

<asp:BoundField DataField="id\_proyecto" HeaderText ="ID" />

<asp:BoundField DataField="proyecto" HeaderText="Proyecto" />

<asp:BoundField DataField ="fecha\_registro" HeaderText="Registro" />

<asp:BoundField DataField ="fecha\_inicio" HeaderText="Inicio" />

<asp:BoundField DataField ="fecha\_final" HeaderText="Termino" />

<asp:HyperLinkField DataNavigateUrlFields="id\_proyecto" DataNavigateUrlFormatString="perfilProyecto.aspx?id\_proyecto={0}" Text="ver datos" />

</Columns>

</asp:GridView>

</div>

</div>

</li>

</ul>

</div>

</div>

</div>

<div id="modal" class="modal">

<div class="modal-content">

<%-- toda la estructura va aqui --%>

<h5 class="section">Cambiar Contraseña</h5>

<div class="divider"></div>

<p class="section">

Favor de ingresar tu actual contraseña y confirmar la nueva contraseña

</p>

<div class="row">

<div class="input-field col s12 m4">

<asp:Label Text="Contraseña Actual" AssociatedControlID="txtOldContrasena" runat="server" />

<asp:TextBox runat="server" TextMode="Password" ID="txtOldContrasena" />

</div>

<div class="input-field col s12 m4">

<asp:Label Text="Nueva Contraseña" AssociatedControlID="txtNewContrasena" runat="server" />

<asp:TextBox runat="server" TextMode="Password" ID="txtNewContrasena" />

</div>

<div class="input-field col s12 m4">

<asp:Label Text="Confirmar Contraseña" AssociatedControlID="txtConfirmarContrasena" runat="server" />

<asp:TextBox runat="server" ID="txtConfirmarContrasena" TextMode="Password" />

</div>

</div>

</div>

<div class="modal-footer">

<asp:LinkButton Text="Cambiar Contraseña" class="modal-close waves-effect waves-green btn-flat" OnClick="btnCambiarContrasena\_Click" ID="btnCambiarContrasena" runat="server" />

<%--<a href="#!" >Agree</a>--%>

</div>

</div>

<div id="modalDeshabilitar" class="modal">

<div class="modal-content">

<%-- toda la estructura va aqui --%>

<h5 class="section">¿ Seguro de deshabilitar el Usuario?</h5>

<div class="divider"></div>

<div class="row">

<h6>

Recuerde que podra habilitar el usuario en cualquier momento

</h6>

</div>

</div>

<div class="modal-footer">

<asp:LinkButton Text="Deshabilitar Usuario" class="modal-close waves-effect waves-green btn-flat" OnClick="btnDeshabilitarUusuario\_Click1" ID="btnDeshabilitarUusuario" runat="server" />

<%--<a href="#!" >Agree</a>--%>

</div>

</div>

<div id="modalHabilitar" class="modal">

<div class="modal-content">

<%-- toda la estructura va aqui --%>

<h5 class="section">¿Desea habilitar el usuario?</h5>

<div class="divider"></div>

<div class="row">

<h6>

El usuario podra volver iniciar sesión.

</h6>

</div>

</div>

<div class="modal-footer">

<asp:LinkButton Text="Habilitar Usuario" class="modal-close waves-effect waves-green btn-flat" OnClick="btnHabilitarUsuario\_Click" ID="btnHabilitarUsuario" runat="server" />

<%--<a href="#!" >Agree</a>--%>

</div>

</div>

</asp:Content>

## Usuario CS

using System;

using System.Collections.Generic;

using System.Data;

using System.Linq;

using System.Web;

using System.Web.UI;

using System.Web.UI.WebControls;

namespace SICAP

{

public partial class usuario : System.Web.UI.Page

{

SICAP.Modelos.Usuario usu;

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

{

if(Request.Params["id\_usuario"] == null)

{

Response.Redirect("usuarios.aspx");

}

usu = new Modelos.Usuario();

usu.id\_usuario = int.Parse(Request.Params["id\_usuario"]);

if ((int)(Session["rol"]) != 1)

{

hlDesUsu.Visible = false;

}

else if ((int)(Session["id\_usuario"]) == int.Parse(Request.Params["id\_usuario"]) || usu.isAdmin())

{

if ((int)(Session["id\_usuario"]) != 1)

{

hlDesUsu.Visible = false;

}

else if ((int)(Session["id\_usuario"]) == int.Parse(Request.Params["id\_usuario"]))

{

hlDesUsu.Visible = false;

}

}

usu.id\_usuario = int.Parse(Request.Params["id\_usuario"]);

if (!usu.IsHabiltado())

{

hlDesUsu.Visible = false;

hlHabUsu.Visible = true;

btnHabilitar.CssClass = "btn right disabled";

hlCambiarPass.CssClass = "btn disabled";

}

if (!IsPostBack)

{

usu.id\_usuario = int.Parse(Request.Params["id\_usuario"]);

DataTable usuario = usu.traerUsuario();

imgPerfil.ImageUrl = (string) (usuario.Rows[0]["ruta"]);

txtNombre.Text = (string)(usuario.Rows[0]["nombre"]);

txtPaterno.Text = (string)(usuario.Rows[0]["paterno"]);

txtMaterno.Text = (string)(usuario.Rows[0]["materno"]);

txtEmail.Text = (string)(usuario.Rows[0]["email"]);

txtEspecialidad.Text = (string)(usuario.Rows[0]["especialidad"]);

txtTelefono.Text = (string) (usuario.Rows[0]["telefono"]);

ddlArea.SelectedItem.Text = (string)(usuario.Rows[0]["area"]);

ddlRol.SelectedItem.Text = (string)(usuario.Rows[0]["rol"]);

gvProyectosUsu.DataSource = usu.traerMisProyectos();

gvProyectosUsu.DataBind();

}

if (ddlArea.Items.Count == 1)

{

foreach (DataRow row in usu.traerAreas().Rows)

{

ddlArea.Items.Add(row["area"].ToString());

}

}

}

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

{

usu.id\_usuario = int.Parse(Request.Params["id\_usuario"]);

usu.nombre = txtNombre.Text.Trim();

usu.paterno = txtPaterno.Text.Trim();

usu.materno = txtMaterno.Text.Trim();

usu.email = txtEmail.Text.Trim();

usu.especialidad = txtEspecialidad.Text.Trim();

usu.telefono = txtTelefono.Text.Trim();

usu.ruta = imgPerfil.ImageUrl;

usu.area = usu.asignarArea(ddlArea.Text);

usu.rol = (ddlRol.Text == "Administrador") ? 1 : 2;

if (!usu.correoExistente())

{

ScriptManager.RegisterClientScriptBlock(this, this.GetType(), "mensaje",

"alert('correo electronico ya registrado')", true);

}

else if (ddlRol.Text == "Elija rol de usuario" || ddlArea.Text == "Elija area")

{

ScriptManager.RegisterClientScriptBlock(this, this.GetType(), "mensaje",

"alert('Favor de completar todos los campos')", true);

}

else

{

usu.modificarUsuario();

ScriptManager.RegisterClientScriptBlock(this, this.GetType(), "mensaje",

"alert('Datos modificados correctamente'); location.href= './usuarios.aspx'", true);

}

}

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

{

if (fuPerfil.HasFile)

{

if (SICAP.Modelos.Usuario.validarExtensionImg(fuPerfil.FileName))

{

usu.ruta = "./Imagenes/" + fuPerfil.FileName;

fuPerfil.SaveAs(Server.MapPath(".") + usu.ruta);

imgPerfil.ImageUrl = usu.ruta;

}

else

{

ScriptManager.RegisterClientScriptBlock(this, this.GetType(), "mensaje",

"alert('La imagen no tiene un formato valido; PNG,JPG o GIF');", true);

}

}

else

{

ScriptManager.RegisterClientScriptBlock(this, this.GetType(), "mensaje",

"alert('No ha selecionado ningun archvio');", true);

}

}

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

{

if(btnHabilitar.Text == "Modificar")

{

txtEmail.Enabled = true;

txtEspecialidad.Enabled = true;

txtMaterno.Enabled = true;

txtNombre.Enabled = true;

txtPaterno.Enabled = true;

txtTelefono.Enabled = true;

ddlArea.Enabled = true;

btnRegisrar.Enabled = true;

fuPerfil.Enabled = true;

btnHabilitar.Text = "Cancelar Modificacion";

btnHabilitar.CssClass = " btn red white-text right";

}

else

{

Response.Redirect("usuarios.aspx");

}

}

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

{

usu.contrasena = txtOldContrasena.Text;

usu.id\_usuario = int.Parse(Request.Params["id\_usuario"]);

if (usu.validarPass())

{

if(txtNewContrasena.Text == txtConfirmarContrasena.Text)

{

usu.CambiarPass(txtNewContrasena.Text);

Session["id\_usuario"] = null;

ScriptManager.RegisterClientScriptBlock(this, this.GetType(), "mensaje",

"alert('Contraseña cambiada correctamente, favor de iniciar sesión con tu nueva contrasena'); location.href= './default.aspx'", true);

}

else

{

ScriptManager.RegisterClientScriptBlock(this, this.GetType(), "mensaje",

"alert('Las contraseñas no coinciden');", true);

}

}

else

{

ScriptManager.RegisterClientScriptBlock(this, this.GetType(), "mensaje",

"alert('Contraseña actual incorrecta');", true);

}

}

protected void btnDeshabilitarUusuario\_Click1(object sender, EventArgs e)

{

var usuario = new SICAP.Modelos.Usuario();

usuario.id\_usuario = int.Parse(Request.Params["id\_usuario"]);

usuario.deshabilitar();

ScriptManager.RegisterClientScriptBlock(this, this.GetType(), "mensaje",

string.Format("location.href='./usuario.aspx?id\_usuario={0}'",Request.Params["id\_usuario"]), true);

}

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

{

var usuario = new SICAP.Modelos.Usuario();

usuario.id\_usuario = int.Parse(Request.Params["id\_usuario"]);

usuario.habilitar();

ScriptManager.RegisterClientScriptBlock(this, this.GetType(), "mensaje",

string.Format("location.href='./usuario.aspx?id\_usuario={0}'", Request.Params["id\_usuario"]), true);

}

}

}

## Usuarios.aspx

<%@ Page Title="" Language="C#" MasterPageFile="~/admin.Master" AutoEventWireup="true" CodeBehind="usuarios.aspx.cs" Inherits="SICAP.usuarios" %>

<asp:Content ID="Content1" ContentPlaceHolderID="head" runat="server">

</asp:Content>

<asp:Content ID="Content2" ContentPlaceHolderID="ContentPlaceHolder1" runat="server">

<div class="contenido card-panel animated fadeIn">

<h5 class="section">Usuarios</h5>

<div class="divider"></div>

<asp:GridView ID="gvUsurios" CssClass="responsive-table" runat="server"

DataKeyNames="id\_usuario" AutoGenerateColumns="false" >

<Columns>

<asp:BoundField DataField="id\_usuario" HeaderText="ID" />

<asp:ImageField DataImageUrlField="ruta" HeaderText="Img" ControlStyle-Width="80px" ControlStyle-CssClass="responsive-img circle" ></asp:ImageField>

<asp:BoundField DataField="nombre" HeaderText="Nombre" />

<asp:BoundField DataField="paterno" HeaderText="Apellido Paterno" />

<asp:BoundField DataField="materno" HeaderText="Apellido Materno" />

<asp:BoundField DataField="area" HeaderText="Área" />

<asp:HyperLinkField DataNavigateUrlFields="id\_usuario" DataNavigateUrlFormatString="usuario.aspx?id\_usuario={0}" Text="Ver perfil" />

</Columns>

</asp:GridView>

</div>

</asp:Content>

## Usuarios CS

using System;

using System.Collections.Generic;

using System.Linq;

using System.Web;

using System.Web.UI;

using System.Web.UI.WebControls;

namespace SICAP

{

public partial class usuarios : System.Web.UI.Page

{

SICAP.Modelos.Usuario usu;

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

{

usu = new SICAP.Modelos.Usuario();

gvUsurios.DataSource = usu.traerUsuarios();

gvUsurios.DataBind();

}

}

}