

2010

2º CFGS Administración  
de Sistemas Informáticos  
I.E.S Gonzalo Nazareno

Daniel Ferrete Olarte



# [MEMORIA DEL PROYECTO INTEGRADO “TAIRAGESTION”]

## INDICE

1. [Presentación](#)
2. [Objetivos](#)
3. [Servicios Utilizados](#)
4. [TairaGestion Programa](#)
  - a. [Estructura del programa](#)
  - b. [Programas utilizados](#)
  - c. [Módulos y complementos](#)
  - d. [Soluciones a problemas encontrados](#)
5. [TairaGestion Web](#)
  - a. [Estructura Web](#)
  - b. [Programas utilizados](#)
  - c. [Soluciones a problemas encontrados](#)
6. [Conclusiones](#)

## 1. Presentación

**TairaGestión** nació como proyecto para subsanar la mayoría de los problemas que surgen a la hora de gestionar cualquier tipo de asociación (en este caso juveniles), ya sea a la hora de registrar nuevos socios, cobrar cuotas, ir a eventos y llevar una organización de voluntarios, etc...

Mucho de los problemas que hay a la hora de dirigir y administrar una asociación es que toda la información esté centralizada y de fácil acceso para todos los cargos de la junta, por ello TairaGestión se pensó para que fuera totalmente online o en su tal caso, un programa que se conectara a una base de datos desde internet.

## 2. Objetivos:

- Tener centralizado toda la información de la asociación (Socios, Cuotas, etc..)
- Programa que utilice recursos con menos costes posibles
- Programa de fácil acceso
- Programa muy intuitivo y de fácil aprendizaje
- Programa accesible desde cualquier dispositivo con internet
- Llevar un control de los socios
- Llevar un control de las cuotas
- Llevar un control sobre eventos
- Llevar un control sobre voluntarios a los eventos
- Llevar un control sobre los materiales de la asociación (Inventario)
- Llevar un control sobre los materiales prestados de la asociación
- Llevar un control sobre los patrocinadores que ayudan a la asociación

## 3. Servicios utilizados:

TairaGestión sobretodo utiliza 2 tipos de servicios:

- Base de datos
- Servicio Web



El servidor de base de datos está montado con MySQL, se ha montado así ya que a la hora de contratar cualquier hosting barato trae al menos 1 base de datos MySQL.

Dentro de lo que son servidores de MySQL he encontrado uno gratuito con posibilidad de conexión remota (cosa que los hosting normales tienen esa opción cortada), el servidor se llama "db4free.net" se utiliza sobretodo como servidor de base de datos de prueba ya que no es demasiado fiable, pero es el único gratuito con posibilidad de conexión remota.

Además también se ha necesitado un servicio web para montar lo que es la parte web del programa TairaGestion ya que la utilidad de este programa es que pueda ser consultado desde cualquier tipo de dispositivo con acceso a internet ya sea ordenador o móvil.



Hay servicios anexos al programa que se han utilizado para la mejor comunicación entre desarrollador y jefe de proyecto (Tutor de proyecto integrado) como es el caso del proyecto **Kenai**:



El Proyecto Kenai se lanzó en septiembre de 2008 por la empresa Sun, iniciativa para promover el desarrollo colaborativo de proyectos Open Source en la Red.

Uno de los aspectos que tiene el Proyecto Kenai es que la incorporación al proyecto es gratuita.

Kenai ofrece hospedaje de hasta 5 proyectos por miembro y una elaborada herramienta de participación y colaboración tanto en la parte de hospedaje, procesos de desarrollo, licencias, perfiles, redes sociales, búsqueda de intereses comunes, etc. Para ello dispone, entre otras herramientas, de Gestión del Código Fuente (Subversion, Mercurial, and Git), Seguimiento de Incidencias (Jira and Bugzilla), Wiki, Foros, Listas de Correo, Herramientas de Descarga,

Integración con NetBeans, etc.

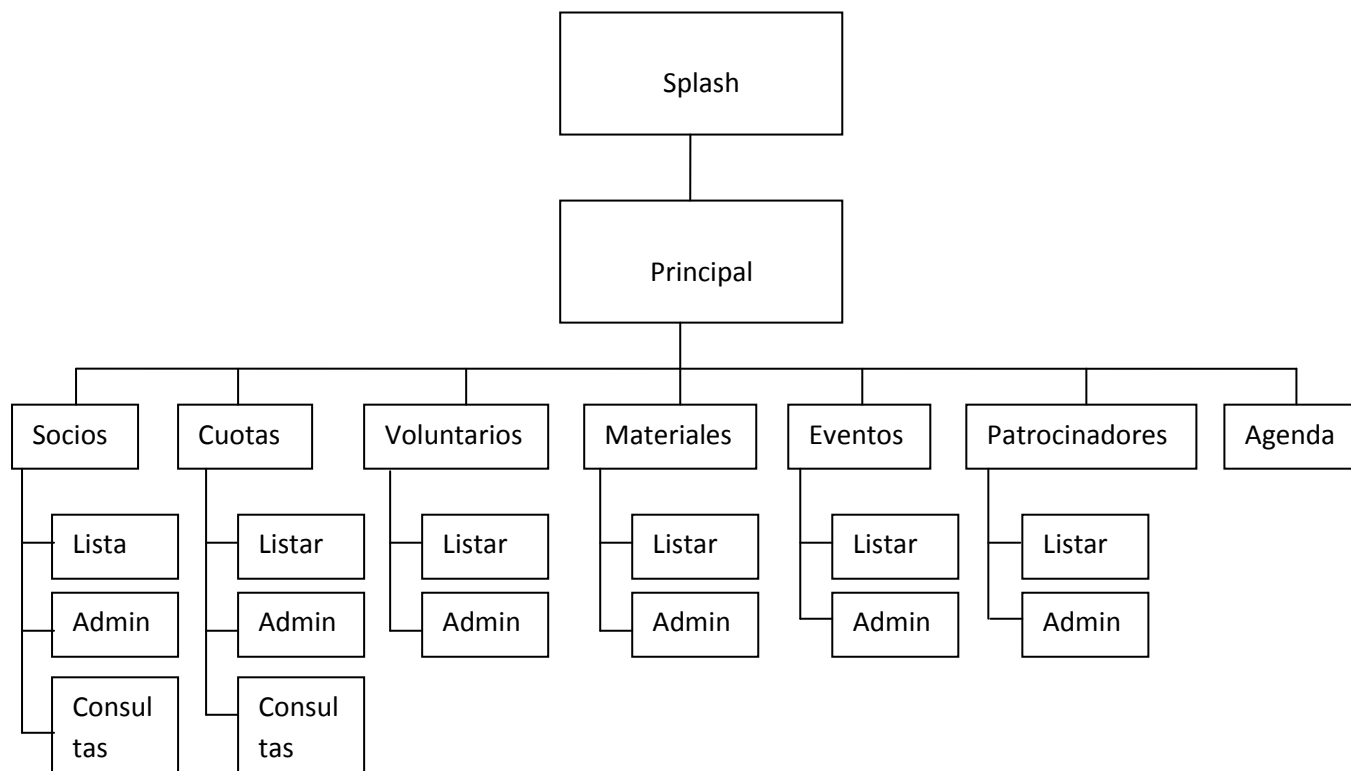
Y además todo ello con un portal muy interactivo e intuitivo, con un aire a redes sociales como Facebook, muy lejos de lo que podría esperarse en una utilidad para desarrolladores. Y con buenas ayudas y tutoriales multimedia.

Por ello ha sido una herramienta muy útil a la hora de desarrollar TairaGestion y muy recomendable para desarrollar cualquier proyecto colaborativo OpenSource.

#### 4. TairaGestion Programa:

##### a. Estructura del programa:

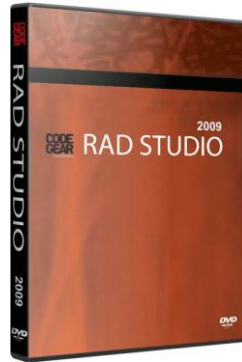
El programa de TairaGestion se estructura de la siguiente forma:



##### b. Programas utilizados:

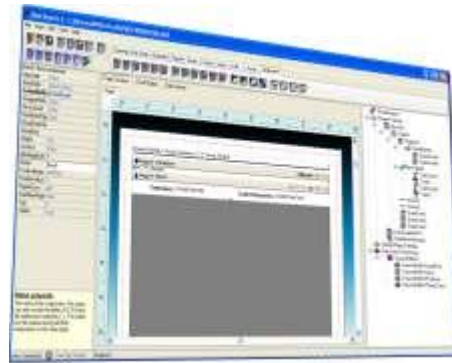
Los programas utilizados para desarrollar el programa TairaGestion son:

- Embarcadero RAD Studio 2010:  
es la suite de desarrollo rápido de aplicaciones más potente del sector para crear visualmente aplicaciones para usuario final que usen intensivamente el GUI y dirigidas a datos, tanto nativas Windows como .NET.



RAD Studio incluye Delphi, C++Builder y Delphi Prism, permitiéndote entregar aplicaciones hasta 5 veces más rápido para múltiples plataformas Windows y de bases de datos.

- Rave Reports: Programa de creación de reportes de informes que pueden ser compilados en Delphi y C++



- MySQL WorkBench: es una herramienta de modelado de bases de datos visual multiplataforma, desarrollada por MySQL. Es la aplicación sucesora del proyecto DBDesigner4. MySQL Workbench está disponible para Linux, Mac OS X, y Windows. Está actualmente en su versión 5.1.16 para cualquiera de las plataformas mencionadas.



- HM NIS Edit:

GUI para el que es considerado por muchos mejor sistema de creación de ejecutables: NSIS.

Cuenta con asistente (wizard) para crear los instaladores cómoda y ágilmente, aunque si prefieres hacerlo manualmente cuenta con la ayuda de plantillas de código, syntax highlighting y más opciones cómodamente accesibles.



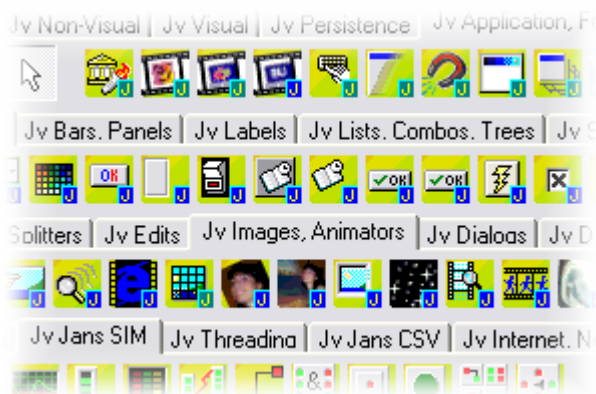
c. *Módulos y Complementos:*

- JVCL (JEDI Visual Component Library):

La Librería Visual de Componentes JEDI, consiste en una gran colección (en la actualidad aproximadamente 500) componentes visuales y no visuales que pueden ser instantáneamente reutilizados en tus proyectos Delphi, Kylix y C++ Builder.

La biblioteca se basa en el código donado de la comunidad JEDI. Se cambia de formato para lograr una mirada común - y - sentimiento, probado y documentado se fusionaron dentro de la biblioteca.

La biblioteca esta agrupada en varias categorías, las cuales han mejorado estandarizadamente, Visual, no visual, Consciente de datos y muchos, muchos más. La biblioteca está a disposición del público "en virtud de los términos" de la Licencia Pública de Mozilla (MPL) y puede utilizarse libremente en ambos, software libre (free software) y software para compartir (shareware), y proyectos comerciales.



- ZeosLib: El ZeosLib es un sistema de los componentes de la base de datos para MySQL, PostgreSQL, Interbase, Firebird, MS SQL, Sybase, Oracle, DB/2 para el constructor de Delphi, de Kylix y de C++.  
He utilizado este sistema de componentes porque es el único gratuito de los que existen.





*d. Soluciones a problemas encontrados:*

- Creación de splash para pedir datos de la base de datos:  
Se necesita tener creado un Form (que será la ventana Splash) y tener abierto el código fuente del proyecto.

En el splash tendremos este código a la hora de pulsar el botón de entrar:

```
procedure Tfsplash.Button2Click(Sender: TObject);

begin
  if passwordEdit.Text = 'tairagestion' then
    begin
      MD.com1.User := euser.Text;
      MD.com1.Database := edb.Text;
      MD.com1.Password := epassword.Text;
      MD.com1.Hostname := ehostname.Text;
      MD.com1.Connected := True;
      ModalResult := mrOK
    end
  else
    begin
      MD.Destroy;
      Application.MessageBox('No has introducido la contraseña válida','Acceso no Autorizado');
      ModalResult := mrAbort;
    end;
end;

class function Tfsplash.Execute: boolean;
begin
  with Tfsplash.Create(nil) do
    try
      Result := ShowModal = mrOk;
    finally
      Free;
    end;
end;

procedure Tfsplash.FormCreate(Sender: TObject);
begin
  passwordEdit.Text:='';
end;
```

Y en el código fuente del proyecto este otro:

```
if Tfsplash.Execute then
begin
Application.Initialize;
Application.MainFormOnTaskbar := True;
Application.Title := 'Taira Gestion V.2';
Application.CreateForm(TF_principal, F_principal);
Application.CreateForm(Tso_listar, so_listar);
Application.CreateForm(Tcuo_listar, cuo_listar);
Application.CreateForm(Tcuo_consultas, cuo_consultas);
Application.CreateForm(Tmat_listar, mat_listar);
Application.CreateForm(Tmat_insertar, mat_insertar);
Application.CreateForm(TCuo_administrar, Cuo_administrar);
Application.CreateForm(Tmat_prestamos, mat_prestamos);
Application.CreateForm(Tmat_prestamos_listar, mat_prestamos_listar);
Application.CreateForm(Teven_listar, even_listar);
Application.CreateForm(Teven_insertar, even_insertar);
Application.CreateForm(Tvolun_listar, volun_listar);
Application.CreateForm(Tpatroc_listar, patroc_listar);
Application.CreateForm(Tvolun_administrar, volun_administrar);
Application.CreateForm(Tpatroc_administrar, patroc_administrar);
Application.CreateForm(TF_agenda, F_agenda);
Application.CreateForm(Tso_insertar, so_insertar);
Application.CreateForm(Tso_consultas, so_consultas);
Application.CreateForm(TF_acercade, F_acercade);
Application.Run;
end
else
begin
end;
```

- Botón para cargar Imágenes con imagen por defecto:

```
procedure Tso_insertar.Cargar1Click(Sender: TObject);
begin
dir0:=ExtractFilePath(Application.ExeName)+'imagenes\';

OPDialog1.Execute();
if OPDialog1.FileName <> '' then
begin
Nolmagen.Picture.LoadFromFile(OPDialog1.FileName);
Clipboard.Assign(Nolmagen.Picture);
with DBISocios.Picture.Bitmap do
DBISocios.PasteFromClipboard;
Nolmagen.Picture.LoadFromFile(dir0 + 'noimagen.bmp');
end;
end;
```

- Botón para limpiar imagen y colocar una por defecto:

```
procedure Tso_insertar.Limpiar1Click(Sender: TObject);  
begin  
  Clipboard.Assign(Nolmagen.Picture);  
  with DBISocios.Picture.Bitmap do  
    DBISocios.PasteFromClipboard;  
  end;
```

- Botón para pegar imagen desde portapapeles:

```
procedure Tso_insertar.Pegar1Click(Sender: TObject);  
begin  
  if Clipboard.HasFormat(CF_BITMAP) then  
    with DBISocios.Picture.Bitmap do  
      DBISocios.PasteFromClipboard  
    else  
      ShowMessage('No hay un Bitmap en el Portapapeles...');  
  end;
```

- Pasos para crear un reporte básico con Rave Reports:

- a. Teniendo creado un datasource de la tabla que queremos crear el reporte debemos crearnos 2 componentes: TRvDataSetConnection y TRvProyect
- b. A continuación entramos en TRvDataSetConnection y en DataSet le colocamos el datasource de la tabla.
- c. Entramos en Rave Reports clickeamos sobre New Data Object, pulsamos sobre Direct Data View y escogemos el que hemos creado.
- d. Creamos un informe simple y lo guardamos.
- e. Por último en TRvProyect solo debemos enlazarlo con el informe que hemos creado.

## 5. TairaGestion Web:

### a. Estructura de la Web:

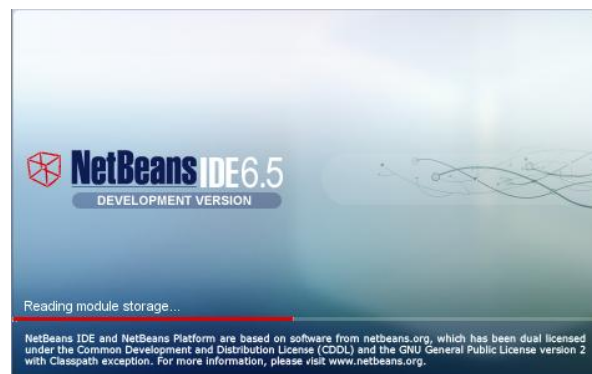
- images/
  - images/eliminar.png
  - images/img01.gif
  - images/img02.gif
  - images/img03.gif
  - images/img04.gif
  - images/img05.png
  - images/img06.gif
  - images/img07.png
  - images/img08.gif
  - images/modificar.png
  - images/mostrarevento.png
  - images/mostrarsocio.png
- conexión.php
- erroracceso.php
- logout.php
- funciones\_fecha.php
- paginator.inc.php
- screen.css
- index.php
- ingresar.php
- socios.php
  - socios\_consultas.php
  - socios\_eliminar.php
  - socios\_consultas.php
  - socios\_modificar.php
  - socios\_mostrar.php
  - socios\_nuevo.php
  - socios\_proceso\_consultas.php
  - socios\_proceso\_insertar.php
- cuotas.php
  - cuotas\_consultas.php
  - cuotas\_eliminar.php
  - cuotas\_consultas.php
  - cuotas\_modificar.php
  - cuotas\_nuevo.php
  - cuotas\_proceso\_consultas.php
  - cuotas\_proceso\_insertar.php

eventos.php  
eventos\_consultas.php  
eventos\_eliminar.php  
eventos\_consultas.php  
eventos\_modificar.php  
eventos\_nuevo.php  
eventos\_mostrar.php  
eventos\_proceso\_consultas.php  
eventos\_proceso\_insertar.php  
voluntarios.php  
voluntarios\_consultas.php  
voluntarios\_eliminar.php  
voluntarios\_consultas.php  
voluntarios\_modificar.php  
voluntarios\_nuevo.php  
voluntarios\_proceso\_consultas.php  
voluntarios\_proceso\_insertar.php  
materiales.php  
materiales\_consultas.php  
materiales\_eliminar.php  
materiales\_consultas.php  
materiales\_modificar.php  
materiales\_nuevo.php  
materiales\_proceso\_consultas.php  
materiales\_proceso\_insertar.php  
prestamos.php  
prestamos\_consultas.php  
prestamos\_eliminar.php  
prestamos\_consultas.php  
prestamos\_modificar.php  
prestamos\_nuevo.php  
prestamos\_proceso\_consultas.php  
prestamos\_proceso\_insertar.php  
patrocinadores.php  
patrocinadores\_consultas.php  
patrocinadores\_eliminar.php  
patrocinadores\_consultas.php  
patrocinadores\_modificar.php  
patrocinadores\_nuevo.php  
patrocinadores\_mostrar.php

patrocinadores\_proceso\_consultas.php  
patrocinadores\_proceso\_insertar.php  
agenda.php  
agenda\_consultas.php  
agenda\_eliminar.php  
agenda\_consultas.php  
agenda\_modificar.php  
agenda\_nuevo.php  
agenda\_proceso\_consultas.php  
agenda\_proceso\_insertar.php

**b. Programas utilizados:**

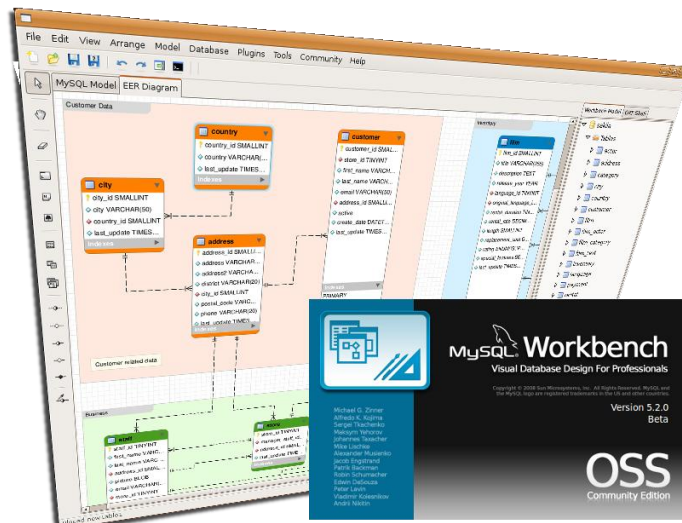
- NetBeans: La versión de netbeans que he utilizado es la PHP. NetBeans permite crear aplicaciones Web con PHP 5, un potente debugger integrado y además viene con soporte para Symfony un gran framework MVC escrito en php. Al tener también soporte para AJAX, cada vez más desarrolladores de aplicaciones LAMP o WAMP, están utilizando NetBeans como IDE.



- Adobe Photoshop: Programa de tratamiento de imágenes, lo he utilizado sobre todo para modificar imágenes del template.



- MySQL WorkBench: como he comentado antes, es una herramienta de modelado de bases de datos visual multiplataforma, desarrollada por MySQL. Es la aplicación sucesora del proyecto DBDesigner4. MySQL Workbench está disponible para Linux, Mac OS X, y Windows. Está actualmente en su versión 5.1.16 para cualquiera de las plataformas mencionadas.



**c. Soluciones a problemas encontrados:**

- Crear conexión sin tener que repetirlas una y otra vez:

- Creamos un archivo llamado conexión.php
- Dentro colocamos estos datos:

```
<?
$dbhost="localhost"; // host del MySQL
                        (generalmente localhost)
$dbusuario="mysql"; // aqui debes ingresar
                        el nombre de usuario
                        // para acceder a la
base
$dbpassword="mysql123"; // password de
acceso para el usuario de la
                        // linea anterior
$db="taira";           // Seleccionamos la
base con la cual trabajar
$conexion = mysql_connect($dbhost,
$dbusuario, $dbpassword);
mysql_select_db($db, $conexion);
?>
```

- Luego solo debemos llamar a esta conexión con un [ include ("conexion.php"); ]

- Control de sesiones:

- Al inicio de cada archivo que queremos controlar las sesiones de los usuarios pondremos esto:

```
session_start();
//antes de hacer los cálculos, compruebo que el usuario está logueado
//utilizamos el mismo script que antes
if(isset($_SESSION['autenticado'])){
//sino, calculamos el tiempo transcurrido
$fechaGuardada = $_SESSION["ultimoAcceso"];
$ahora = date("Y-n-j H:i:s");
$tiempo_transcurrido = (strtotime($ahora)-strtotime($fechaGuardada));
//comparamos el tiempo transcurrido
if($tiempo_transcurrido >= 600) {
//si pasaron 10 minutos o más
session_destroy(); // destruyo la sesión
header("Location: index.php");
//envío al usuario a la pag. de autenticación
//sino, actualizo la fecha de la sesión
}else {
$_SESSION["ultimoAcceso"] = $ahora;
}
} else {
header("location: index.php");
}
/* Cerramos la parte de código PHP porque vamos a escribir bastante HTML y nos será
más cómodo así que metiendo echo's */
```



- Validar usuarios con contraseña de SMF:
  - Creamos una clase llamada ingresar.php
  - En dicha clase añadiremos esto:

```
include("conexion.php");
function protege($texto)
{
    $texto = str_replace("'", "\'", $texto);
    $texto = htmlspecialchars($texto);
    $texto = htmlentities($texto);
    $texto = trim($texto);
    return $texto;
}
//llamada
$user=protege($_REQUEST['usuario']);
$pass=protege($_REQUEST['contrasena']);
//Encriptacion
$contrasena=sha1($user.$pass);
//Sentencia SQL para buscar un usuario con esos datos
$sql = "SELECT * FROM smf_members WHERE
memberName='$user' and passwd='$contrasena' AND
ID_GROUP='1'";
//Ejecuto la sentencia
$rs = mysql_query($sql,$conexion);
//vemos si el usuario y contraseña es válido
//si la ejecución de la sentencia SQL nos da algún resultado
//es que si que existe esa combinación usuario/contraseña
if ($ol=mysql_fetch_array($rs)){
    //usuario y contraseña válidos
    //defino una sesion y guardo datos
    $_SESSION['autenticado']=$user;
    $_SESSION["ultimoAcceso"]= date("Y-n-j H:i:s");
    header ("Location: index.php");
}else {
    //si no existe le mando a la pantalla de error
    header("Location: erroracceso.php?errorusuario=si");
}
mysql_free_result($rs);
mysql_close($conn);
```

- Y desde index.php crearemos un formulario donde envíe el usuario y la contraseña a la clase ingresar.php

```
if(isset($_SESSION['autenticado']))}{else{ echo "

    <form align=\"right\" action=\"ingresar.php\"
    method=\"POST\">

        Login: <input type=\"text\" size=5
        name=\"usuario\"/>

        Password: <input type=\"password\" size=5
        name=\"contrasena\"/>

        <input type=\"submit\" value=\"Entrar\" /></form>;}
```

- Subir una foto al servidor:

- Debemos empezar el formulario del archivo de registro de esta manera para que recoja subida de archivos:

```
<FORM ACTION="socios_proceso_insertar.php"
METHOD="POST" enctype="multipart/form-data">
```

- Dentro del formulario de alta deberemos de poner este código:

```
<TR>
<TD>Foto:</TD>
<TD><INPUT TYPE="file" ACCEPT="jpeg, png, gif"
NAME="foto"></TD>
</TR>
```

- Dentro del archivo del proceso de insertar un nuevo socio añadimos estas 2 líneas:

```
$nom_imagen = $_HTTP_POST_FILES['foto']['name'];
    move_uploaded_file ( $_FILES [ 'foto' ][ 'tmp_name' ] ,
    $carpeta . '/' . $nom_imagen);
```

- Si nos fijamos tenemos que crearnos una variable "carpeta" donde contendrá la ruta en donde queremos guardar las fotos.
- Y por último en la línea de inserción a la base de datos, debemos especificar en el campo foto esto:

'\$carpeta\$nom\_imagen'

Lo cual le dirá en que ruta y con qué nombre está la imagen.

## 6. Conclusiones:

Gracias a la realización de este proyecto he podido llevar a cabo los siguientes objetivos:

- Reforzar el aprendizaje del lenguaje de Delphi
- Reforzar el aprendizaje del debug de Delphi
- Aprender a utilizar correctamente Rave Reports para integrarlo con Delphi
- Aprender a utilizar correctamente Netbeans
- Aprender a utilizar proyecto kenai con Netbeans
- Aprender a utilizar el debug de Netbeans
- Aprender a crear instaladores con HM NIS EDIT
- Reforzar el aprendizaje de la creación de diseños de bases de datos
- Reforzar el aprendizaje del lenguaje SQL
- Reforzar el aprendizaje del lenguaje PHP
- Reforzar el aprendizaje de CSS
- Poder hacer un programa bastante valorado dentro del sector de las asociaciones
- Por último y más importante, aprender a buscar información y solucionar problemas por mi mismo ante algo que no he realizado antes.