

# **RAD**

Desarrollo Rapido de Aplicaciones Internet

# 1. Descripción

**RAD** es un entorno de desarrollo y ejecución de aplicaciones Internet desarrolladas en PHP.

Debido a estas dos características constituye un entorno de trabajo corporativo completo, muy potente y versátil.

Potente porque dispone de múltiples funciones integradas en su arquitectura básica:

- Control de Acceso (usuario y clave, certificado digital).
- Múltiples Niveles de Permisos o Perfiles, definibles por el administrador.
- Generador Automático de Aplicaciones de Gestión de Datos. Con gestor de base de datos integrado.
- Multidioma.
- Log de Operaciones (datos, paginas y tiempos), para auditoría, seguridad y recuperación de estado de datos a una fecha y hora determinada.
- Permite trabajar con múltiples copias de datos (históricos, pruebas, simulaciones).
- Contiene un gestor de contenidos para confeccionar una web:
  - . Páginas y Secciones con contenidos HTML para confeccionar un portal.
  - . Noticias, Foros de discusión.
- Incorpora un conjunto de utilidades de trabajo en grupo y comunicación:
  - . Mensajería Privada, Email y Chat.
  - . Agenda-Calendario por usuario.
  - . Libreta de Direcciones
  - . Recursos (ficheros, enlaces, anotaciones) propios y compartidos.
  - . Páginas Personales y Disco Virtual por usuario.

Versátil por los múltiples usos que permite:

- Portal Corporativo (con contenidos modificables desde navegador).
- Portal de Contenidos de grupos privados o abierta al público (periódico digital, weblog).
- Aplicación Internet o Intranet de gestión de datos, y herramienta de trabajo corporativo.

Además de otras ventajas:

- Puede ser utilizado para el desarrollo de aplicaciones Internet que utilicen datos de base de datos SQL de otras aplicaciones ya existentes (coexistiendo ambas).
- Permite una migración gradual desde otras aplicaciones. Manteniendo las mismas bases de datos.
- Muy útil en grandes desarrollos al permitir realizar un prototipado rápido y desarrollos por fases o graduales (desarrollo y mantenimiento ágil de aplicaciones).
- Visión de todos los datos relacionados con el dato que se visualiza, permitiendo que se pueda acceder a toda la información desde el dato principal.

Como colofón, decir que el propio gestor de contenidos que incorpora se ha creado con las herramientas que contiene. Se gestiona, publica, distribuye, desarrolla y mantiene con la propia herramienta.

## **1.1. Entorno de ejecución**

Como entorno de ejecución tiene una arquitectura similar a una aplicación de gestión de datos con control de grupos de permisos ilimitados o perfiles de usuario, de tal forma que un programa (módulo o bloque) solo puede ser ejecutado por aquellos usuarios pertenecientes al perfil que tiene acceso a dicha parte de la aplicación. Incluso un mismo programa puede tener distinto comportamiento para distintos perfiles y/o usuarios.

La definición de estos perfiles, y programas se realiza desde la parte de Administración, el usuario o usuarios que disponen del perfil de administrador del sistema.

Los usuarios se pueden dar de alta incluso desde la zona no identificada, aunque para asociarse a los perfiles se requiere la parte de Administración. En cualquier caso desde la parte de Administración se tiene acceso a todas las partes de la herramienta.

Una vez instalado, se crean las tablas mínimas para que funcione y se definen los programas y perfiles básicas, con los usuarios de pruebas. A partir de este momento se pueden cargar nuevas tablas de aplicaciones propias y crear los módulos que gestionen estas tablas propias. Este es el proceso de desarrollo habitual de cualquier nueva aplicación Internet o Intranet.

La arquitectura básica se compone de un conjunto mínimo de tablas y un conjunto mínimo de módulos que realizan todo el control.

Compruebe el capítulo de Administración para conocer a fondo todas sus posibilidades.

## **1.2. Entorno de desarrollo**

Como entorno de desarrollo, se dispone del Generador Automático de Aplicaciones de Gestión de Datos, denominado "phpRAD". Además se incorporan otras utilidades como un gestor de ficheros por red, un ejecutor de comandos remotos y un gestor de base de datos SQL (con control de perfiles, y que solo permite gestionar las bases de datos que pertenecen a la aplicación).

El generador de código phpRAD permite crear una aplicación (conjunto de módulos en código fuente PHP) de gestión de un conjunto de tablas correspondiente a cualquier aplicación que deseemos desarrollar. Para ello, se deben crear previamente las tablas de la nueva aplicación a gestionar. Es muy flexible y dispone de múltiples opciones de generación, permitiendo crear desde una gestión completa con las funciones básicas de listado, modificación, inserción, borrado. Hasta un simple listado, informe, una creación de un único registro, etc. Y permite integrar funciones propias (calendarios, gráficos, interfaces gráficas) o modificar las funciones que aporta.

Compruebe el capítulo de Desarrollo para comprobar sus posibilidades.

## **2. Instalación**

Guía para instalar RAD en un ordenador.

### **2.1. Requisitos**

Se requiere un servidor web con soporte PHP y un motor de base de datos SQL.

#### **2.1.1. Servidor WEB con PHP**

Para instalar este Sistema en un servidor se requiere tener un servidor Web con PHP instalado y funcionando.

Para su funcionamiento completo se requiere un servidor Linux con Apache y PHP, aunque puede funcionar las partes básicas en otros sistemas operativos y otros servidores web, siempre que tengan PHP instalado.

La instalación de PHP debe estar compilada con soporte de sesiones y alguna de las base de datos soportadas. Se sugiere MySQL.

#### **2.1.2. Base de Datos.**

El entorno de ejecución soporta las Bases de Datos siguientes:

- MySQL
- Oracle
- PostgreSQL
- mSQL
- ODBC
- Interbase
- Sybase

Se sugiere utilizar MySQL, por ser la base de datos con más amplia variedad de aplicaciones desarrolladas sobre RAD.

## **2.2. Procedimiento de Instalación**

Partiendo del paquete completo (*tgz*) de la aplicación (que contiene los programas y los datos) se puede instalar la aplicación en cualquier servidor que cumpla los requisitos anteriores. Esta instalación puede realizarse tantas veces se desee en el mismo o en distintos servidores. Esto permite tener distintas copias de la aplicación, incluso en el mismo servidor.

### 2.2.1. Descompresión del paquete

Primero se debe descomprimir el paquete "tgz" en un directorio accesible por el servidor Web. Y se deben poner los permisos del servidor web a todo el árbol de directorios y ficheros descomprimidos (chown -R webserveruser:webservergroup /webserver/directory/RAD).

Esta instalación se puede repetir en otros directorios del mismo servidor, y tener múltiples aplicaciones instaladas en el mismo ordenador (cada una con sus propios datos y programas).

Después de descomprimir el paquete, se procede a instalar las tablas de la arquitectura y a modificar el fichero de configuración con el módulo de instalación "install.php".

El paquete base viene con una base de datos (RAD.sql) que define las tablas mínimas y trae un conjunto de datos de pruebas (usuarios, perfiles, noticias).

### 2.2.2. Instalación automática

Para realizar de forma automática la instalación se debe solicitar desde un navegador la URL que apunte al servidor y directorio donde se ha descomprimido el paquete (p.e. <http://localhost/directorio/>).

En este momento aparece el formulario correspondiente al módulo "install.php" que solicita los datos de la configuración. Al pulsar "Instalar" carga las tablas mínimas y crea el fichero de configuración ("config.php").

### 2.2.3. Instalación manual

Si no se realiza la instalación descrita en el apartado anterior se puede realizar la instalación de forma manual, para ello se debe crear una base de datos y cargar el fichero con las tablas y datos "RAD.sql", por ejemplo para MySQL se deben ejecutar los comandos:

```
mysqladmin -u root -p create MIBASEDEDATOS  
mysql -u root -p MIBASEDEDATOS < privado/RAD.sql
```

En este ejemplo se utiliza el usuario *root*, suponiendo que éste tiene permiso para acceder a MySQL, y el fichero "privado/RAD.sql" que contiene las tablas y datos de nuestra aplicación base (hay un conjunto de tablas que son imprescindibles para el control y funcionamiento básico -bloques, módulos, perfiles, usuarios-).

Una vez cargadas las tablas se debe copiar el fichero config.default.php como "config.php" y modificar en éste último las constantes PHP a los valores correspondientes a nuestra instalación.

## 2.3. Comenzando a Usar la Herramienta

Una vez instalado se puede eliminar el programa install.php (o quitarle todos los permisos). Y desde un navegador poner la URL que corresponde a nuestra servidor y directorio.

La instalación básica define el usuario "admin" con clave "admin" que permite comenzar a usar la herramienta con todos los permisos. Se aconseja cambiar la clave de acceso de este usuario, una vez instalado.

## 3. Administración

Para crear el Portal o Aplicación de nuestras necesidades, y mantener el sistema en funcionamiento la arquitectura dispone de un conjunto de herramientas básicas de administración.

Para acceder a estas herramientas se debe entrar identificado con un usuario con perfil de "Administrador de Sistema". Al instalar el paquete se crea por defecto el usuario "admin" y con clave "admin" para comenzar a operar. Se puede utilizar este usuario para administrar la aplicación.

### 3.1. Usuarios

Para utilizar la aplicación se puede hacer de forma anónima accediendo a las partes de la aplicación que es pública, o de forma identificada con un usuario y una clave. Este usuario tendrá asociado un conjunto de perfiles o permisos de uso, lo cual nos da acceso a otras partes de la aplicación. Para definir estos usuarios, desde la opción del menú "Adm. RAD->Usuarios" podrás acceder a esta herramienta.

En la pantalla aparecerá un listado con la primera página de usuarios. Desde ella se podrá consultar los datos de un usuario pulsando sobre la línea correspondiente al usuario (vista de "Detalle"). Y desde esta pantalla de datos del usuario se podrá modificar pulsando sobre la opción "Editar", o borrar el registro con la opción "Borrar". También se podrá buscar un usuario por alguno de sus datos (con la opción "Buscar"), listar todos los usuarios para imprimir (con la opción "Imprimir"), y dar de alta un nuevo usuario con la opción "Nuevo". Incluso se podrá obtener un listado de usuarios que cumplan unas determinadas condiciones con la opción "Informe".

Para crear un nuevo usuario tomando como datos de partida los de otro usuario ya existente, es necesario ir a la vista de detalle del usuario ya existente y pulsar sobre la opción "Nuevo", de tal forma que tomará los datos por defecto del nuevo usuario como los del ya existente (este mecanismo permite duplicar registros).

El funcionamiento de las opciones de "Listar", "Editar", "Buscar", "Imprimir", "Borrar", "Informe", "Detalle" y "Nuevo" es común a todos los módulos de la aplicación a la hora de gestionar los datos de una tabla.

Cada persona que utilice este sistema debería tener un usuario propio. En caso de no tenerlo no se tendrá acceso a muchas partes de la aplicación, aunque si podrá utilizar como visitante las partes que el administrador de la aplicación (módulos y bloques) haya definido como de acceso público.

Un usuario se puede desactivar editando su registro y poniendo el campo "Activo" a "N". Con ello se mantiene reservado el código del usuario, pero no se le permite su acceso.

El código de Usuario debe ser único en todo el sistema, y consiste en un conjunto de letras y números que identifica a una persona (sin espacios).

Desde el registro de detalle de un usuario se pueden consultar sus perfiles, módulos autorizados, visitas y un informe de estadísticas. Además, se permite entrar a la aplicación identificado como el propio usuario, y también, desconectar las sesiones activas del usuario (echar al usuario de la aplicación si estuviese conectado en ese momento).

### **3.2. Perfiles**

Desde la opción "Adm. RAD->Perfiles" se gestiona la tabla de Perfiles. Que son los roles o nombres de permisos a los que se van a asociar los usuarios y los módulos (partes) de la aplicación.

Al igual que ocurre en la gestión de usuarios, aquí aparece una pantalla con todos los Perfiles de la tabla y desde aquí se puede modificar, crear o borrar dichos perfiles.

Este tipo de registro es una forma conceptual de clasificar los niveles de acceso a las partes de la aplicación (Editor, Contable, Compras, Facturación, ...), para posteriormente asignarlos a los usuarios, y determinar qué partes de la aplicación (módulos y bloques) tienen un determinado nivel.

Un Perfil autoriza a utilizar un conjunto de módulos o bloques. Un mismo Usuario puede tener uno o varios Perfiles. Y un mismo módulo o bloque puede ser utilizable por uno o varios Perfiles.

Se pueden definir tantos Perfiles como se deseen.

### **3.3. Bloques**

Un bloque es un pequeño programa en PHP (que genera un contenido HTML) o un fichero o un contenido HTML, que será incrustado en una determinada zona de la página (izquierda, derecha, centro, arriba, abajo).

Principalmente, se utilizan para mostrar enlaces (que se generan dinámicamente) a páginas del propio sistema u otras direcciones. De esta forma se construyen los menús, enlaces rápidos, cajas de búsqueda, miniagendas, ...

Los bloques (al igual que los módulos) poseen una autorización. Esta autorización está constituida por todos los "Perfiles" que lo pueden ejecutar (visualizar). Contienen también un campo "Orden" que indica el orden de ejecución y aparición dentro de la página. Y una "Posición" que indica en qué zona de la página aparecerá el texto que genera (izquierda, derecha, arriba, centro, abajo).

El campo "Fichero" contiene el nombre del programa PHP situado en el directorio "blocks" (no es necesario poner la extensión php).

El campo "Contenido" es el contenido HTML fijo que se mostrará, en caso de no existir el Fichero PHP.

Estos bloques pueden ser desarrollados por uno mismo o incluso utilizar los procedentes de otras aplicaciones tipo "nuke".

La herramienta aporta un conjunto de bloques básicos entre los que se encuentran: un bloque que no genera ningún contenido, pero guarda estadísticas del usuario (stat.php), otro para mostrar enlaces a los módulos a los que tiene acceso cada usuario (showmenumodules.php), otro para solicitar datos de identificación (getlogin.php), otro para solicitar datos de búsqueda (search.php) y ejecutar el módulo de búsqueda, etc. Otros bloques existentes son: un selector de idioma, selector de aspecto, anuncios publicitarios (banner), persiana de presentación de noticias y foros, entrada a chat, etc.

El único requisito que debe cumplir un fichero de bloque es ser un fichero php correcto y que lo que desee mostrar lo guarde en una variable global denominada "content".

### **3.4. Módulos**

Son los programas PHP que se utilizan para realizar una función completa. Generalmente gestionar una tabla.

Estos programas deben cumplir unos requisitos para ser utilizados en este sistema, y además deben ser definidos en la tabla de módulos con los perfiles correspondientes, para que un usuario pueda acceder a ellos (al igual que los bloques).

Pueden ser creados manualmente o generados con el Generador Automático de Aplicaciones de Gestión de Datos. Y es precisamente esta herramienta de generación automática la que aporta una gran potencia a phpRAD. Pues la propia parte de Administración de phpRAD ha sido generada de forma automática.

Al acceder a "Adm. RAD->Módulos" aparece el listado de módulos con sus características (nombre, directorio, fichero, parámetros, perfiles), y de la misma forma que en el resto de pantallas, desde ella se pueden crear, modificar o borrar los registros de módulos.

El campo "Grupo de Menú" del módulo es un literal que agrupa al conjunto de módulos, para que posteriormente se puedan mostrar sus enlaces agrupados (este campo lo utiliza el bloque "showmenumodules" para generar los menús de la aplicación). El campo "Ítem de Menú" del módulo es el literal que se utilizará para mostrar el enlace a este módulo dentro del grupo de menú anterior.

El campo "Directorio", es donde reside el módulo (debe ser un subdirectorio del directorio "modules").

El campo "Fichero" es el nombre del fichero PHP que corresponde al módulo (no es necesario poner la extensión php).

"Activo" indica si este módulo puede ser ejecutado por la arquitectura o no.

"Visible" es un indicador que utiliza el bloque de menús ("showmenumodules") para mostrar o no el enlace a este módulo.

Un fichero para ser un módulo debe ser un fichero PHP correcto. Y se recomienda que se compruebe que no debe ser invocado directamente, sino que debe ser ejecutado por el entorno de ejecución de la herramienta. De esta forma no es posible saltar el control de autorizaciones. Esto se explica en el capítulo de Desarrollo.

La arquitectura pone a disposición del módulo un acceso a la base de datos con el manejador ("handle") contenido en la variable RAD\_dbi, el conjunto de variables contenidas en el fichero "config.php", y un conjunto de variables de sesión contenidas en la matriz "HTTP\_SESSION\_VARS" (entre ellas se encuentran los datos del usuario conectado).

Revise el capítulo de Desarrollo para conocer mejor como se deben crear los módulos y la posibilidad de desarrollarlos automáticamente y/o manualmente.



### **3.5. Otras Tablas y funciones de Administración**

En el menú "Adm. RAD" del administrador del sistema aparecen los enlaces a la gestión de las restantes tablas de la arquitectura básica del sistema (aunque no son estrictamente necesarias). Estas tablas son:

- Estadísticas: es cada una de las visitas a nuestro portal. Ya sea por un usuario identificado, o por un visitante. Y contiene información de todas las páginas que ha visitado, tiempos, y la IP desde la que procede (e información de RIPE de esta IP). Desde una visita de un usuario se puede acceder a la ficha de este usuario para ver todas las demás visitas que hubiera hecho.
- Páginas y Secciones: las páginas dinámicas o artículos son contenidos HTML modificables, y que permiten elaborar los contenidos del portal (información corporativa, periódicos digitales, catálogos de productos, ...). Las secciones son las clasificaciones en grupos de estas páginas dinámicas, para ser mostradas en los menús del portal.
- Tipos Categorías: permite definir los distintos grupos o categorías de contenidos de noticias, foros, ayudas, ...
- Contenidos: contienen los registros de noticias, foros, mensajes, ayudas, definidos en las Categorías anteriores.
- Banners: permite definir contenidos publicitarios que se pueden publicar en distintas zonas de la página.
- Encuestas (poll): para definir encuestas con sus preguntas. Estas encuestas pueden ser mostradas en la página, para que puedan ser rellenadas por los visitantes y usuarios.
- Salas de Chat: permite definir las distintas salas de chat que pueden existir para ser usadas por los visitantes o por los distintos tipos de perfil de usuario.

También, dentro del menú "Adm.RAD" se acceden a funciones de administración del servidor y la aplicación:

- Base de Datos: permite gestionar la base de datos de esta aplicación. Realizar copias de respaldo, modificar la estructura, etc. Es una herramienta muy útil en la fase de desarrollo.
- Tabla de Datos: permite rellenar (o volcar a texto) el contenido de una tabla desde un texto que contiene la información de sus registros (por ejemplo un fichero CSV). Permite seleccionar los campos de la tabla y su orden. Es muy útil para obtener un volcado de una tabla en formato CSV. Y viceversa, para rellenar el contenido de una tabla desde CSV. Este modulo también permite crear una copia de la base de datos completa, y consultar una operación en el Log de acceso a la base de datos.
- Gestor de Ficheros: permite gestionar todo el árbol de directorios de esta aplicación y sus ficheros. Para crear, borrar o modificar directorios y ficheros. También es una herramienta muy útil en desarrollo.
- Adm Disco Virtual: permite gestionar los directorios de los discos virtuales y las páginas personales de los usuarios. Permite buscar ficheros muy grandes, y los ficheros que se han modificado desde una fecha determinada. Muy útil para comprobar los contenidos que están introduciendo los usuarios en sus discos virtuales y sus páginas personales.
- Shell: permite ejecutar comandos en modo terminal en el servidor. Solo debe permitirse su uso en equipos de desarrollo.
- Terminal SSH: permite acceder en modo terminal al servidor. Similar al anterior.
- Inf. Sistema: muestra información del servidor (estado de discos, CPU, red, memoria, versión de PHP, ).

Y como herramientas de desarrollo:

- Generador de Módulos (phpRAD): es la herramienta que permite crear módulos de forma automática, para gestionar una tabla de la base de datos. Es la herramienta que da mayor potencia a esta aplicación, pues permite crear aplicaciones grandes, muy rápidamente. Para conocer en profundidad esta herramienta se debe consultar el capítulo de Desarrollo.
- Activa Módulos: permite dar de alta un grupo módulos de un directorio. Muy útil cuando se han desarrollado muchos módulos y se desean dar de alta para poder ser utilizados por los

usuarios.

- Idiomas: permite definir las constantes de idiomas que utiliza la aplicación. Se pueden crear ficheros para nuevos idiomas, y copiar todas las definiciones de otro idioma.

El funcionamiento de los módulos de gestión de tablas es similar en todos los casos: listado, edición, nuevo, borrado y modificación de registros de la tabla correspondiente. Todos estos módulos han sido desarrollados con el Generador Automático.

### **3.6. Aspectos (themes)**

Los aspectos son rutinas PHP e imágenes contenidas en subdirectorios del directorio "themes", que permiten cambiar por completo el aspecto de presentación de las páginas, cuyos contenidos HTML son generados por los bloques y los módulos.

El aspecto se selecciona al instalar, y se define en el fichero "config.php". Adicionalmente cada usuario puede elegir su propio aspecto.

Estos aspectos son similares a los de las aplicaciones tipo "nuke". Aunque en el caso de RAD los ficheros de aspecto son más cortos, y sencillos (no se requieren tantas rutinas).

En resumen un Aspecto es un subdirectorio del directorio "themes", y debe contener un fichero PHP de nombre "theme.php".

Opcionalmente puede contener un fichero de estilo (p.e. style.css). Los estilos que deberían definirse en la pagina de estilo, y que la aplicación RAD utiliza son:

- tiny : para textos pequeños (elemento FONT)
- block : para las tablas de menús dinámicos (elementos TABLE, TR, TD, A)
- blockcab : para los grupos de menú (elementos TABLE, TR, TD, A)
- blockcabsel : para el grupo de menú en el que se encuentra la pagina visitada (elementos TABLE, TR, TD, A)
- blockitem : para cada elementos de menú (elementos TABLE, TR, TD, A)
- blockitemsel : para el elemento de menú en el que se encuentra la pagina visitada (elementos TABLE, TR, TD, A)
- blocksubitem : para cada elementos de menú de segundo nivel (elementos TABLE, TR, TD, A)
- blocksubitemsel : para el elemento de menú de segundo nivel en el que se encuentra la pagina visitada (elementos TABLE, TR, TD, A)
- detail : para las tablas (elemento TABLE) donde se muestra el detalle de un registro, y para el valor de un campo del registro (elemento TD)
- detailtit : para el titulo de un campo de un registro cuando se muestra su detalle (elemento TD)
- optlevelNN : para mostrar el nivel NN de una caja de selección (elemento OPTION)
- button : aspecto de los botones (elemento BUTTON)
- inputdisabled : aspecto de las cajas de entrada de texto deshabilitadas (elemento INPUT)
- inputreadonly : aspecto de las cajas de entrada de texto de solo lectura (elemento INPUT)
- row1 : filas impares al mostrar una lista de registros (elemento TR, TD)
- row2 : filas pares al mostrar una lista de registros (elemento TR, TD)
- browse : tabla de lista de registros (elemento TABLE)
- menu : tabla de menú de opciones (elemento TABLE, TR), equivalente a las pestañas
- menuon : elemento de tabla de menú seleccionado (elemento TD)
- menuoff : elemento de tabla de menú no seleccionado (elemento TD)
- submenu : tabla de submenú de opciones (elemento TABLE, TR), equivalente a pestañas
- submenuon : elemento de tabla de submenú seleccionado (elemento TD)
- submenuoff : elemento de tabla de submenú no seleccionado (elemento TD)
- borde : tabla que engloba todo el contenido que devuelve un modulo (elementos TABLE, TR, TD)
- content : para la tablas/filas/columnas con contenidos dinámicos (noticias, páginas, foros) (elementos TABLE, TR, TD, A)

El fichero "theme.php" debe tener definir al menos las siguientes funciones:

```
<?php
/*****/
function themeheader() {
// Funcion que construye el cuerpo HTML de la pagina, invocando las distintas zonas de bloques y
el modulo
// en el lugar que desee
.....
}
/*****/
function themefooter() {
// Funcion que construye el pie HTML una vez se ha ejecutado el modulo
.....
}
/*****/
function themesidebox($TMP_title, $TMP_content) {
// Funcion que muestra el contenido de un bloque, recibe el titulo y el contenido html que devuelve
el bloque
.....
}
/*****/
function OpenTable() { // Funcion que genera una apertura de tabla y que contendra mostrar el
contenido que devuelve el modulo
.....
}
function CloseTable() { // Cierre de la tabla anterior
.....
}
?>
```

Existe un mecanismo más completo que permite definir el HTML de la página de portada y de las páginas de interior del portal. Para ello se deben crear, además, en el mismo subdirectorio del aspecto la página index.php y modules.php con el contenido HTML y las variables : \$CONTENT\_BLOCKS\_LEFT, \$CONTENT\_BLOCKS\_RIGHT y \$CONTENT\_MODULE que será sustituidas por el contenido HTML generado por la aplicación para el contenido de los bloques de la izquierda, la derecha y el contenido del módulo correspondiente.

En este caso el fichero theme.php que debe ponerse en el subdirectorio del aspecto es el siguiente:

```
<?php
/*****/
function themeheader() {
global $SESSION_blocks_left, $SESSION_blocks_right, $CONTENT_BLOCKS_RIGHT,
$CONTENT_BLOCKS_LEFT, $headeroff, $blocksoff, $footeroff;
$headeroff=""; $blocksoff=""; $footeroff=""; $SESSION_blocks_left = 1;
$SESSION_blocks_right = 1;
ob_end_clean(); ob_start();
blocks("left"); $CONTENT_BLOCKS_LEFT=ob_get_contents();
ob_end_clean(); ob_start();
blocks("right"); $CONTENT_BLOCKS_RIGHT=ob_get_contents();
ob_end_clean(); ob_start();
$blocksoff="x"; $SESSION_blocks_left = 0; $SESSION_blocks_right = 0;
$CONTENT_BLOCKS_LEFT=str_replace("\n", "", $CONTENT_BLOCKS_LEFT);
$CONTENT_BLOCKS_LEFT=str_replace("<br><br>", "<br>", $CONTENT_BLOCKS_LEFT);
$CONTENT_BLOCKS_RIGHT=str_replace("\n", "", $CONTENT_BLOCKS_RIGHT);
$CONTENT_BLOCKS_RIGHT=str_replace("<br><br>", "<br>", $CONTENT_BLOCKS_RIGHT);
```

```

    echo "\n<script type='text/javascript' src='RAD.js'></script>\n";
}
/*****/
function themefooter() {
global $CONTENT_FOOT, $V_index, $CONTENT_MODULE, $CONTENT_BLOCKS_RIGHT,
$CONTENT_BLOCKS_LEFT;
    $CONTENT_MODULE=ob_get_contents();
    ob_end_clean(); ob_start();
    footmsg(); $CONTENT_FOOT=ob_get_contents();
    ob_end_clean();
    chdir("themes/".$TMP_dirtheme);
    if ($V_index == 1) include("index.php");
    else include("modules.php");
}
/*****/
function themesidebox($TMP_title, $TMP_content) {
// if (trim($TMP_title)!="") $TMP_title.="<br>\n";
// $TMP_content=str_replace("\n\n","\n",$TMP_content);
// $TMP_content=str_replace("\n","<br>",$TMP_content);
// $TMP_content=str_replace("<br><br>","<br>",$TMP_content);
    echo $TMP_title.$TMP_content;
}
/*****/
function OpenTable() { echo "<table width=100% class=borde><tr><td class=borde>"; }
function CloseTable() { echo "</td></tr></table>"; }
?>

```

El mecanismo más rápido para construir un aspecto nuevo, es buscar el más parecido al que deseamos, dentro de los aspectos existentes, y posteriormente copiar su directorio con el nuevo nombre del aspecto buscado.

### 3.7. Portada

Al instalar RAD, se crea por defecto una configuración donde se definen los Bloques y Módulos que están en la Portada, "Home Page" o index.

Esta configuración de la Portada (contenido de los Bloques y Módulos en "Home Page") se puede modificar, para crear el aspecto deseado de la portada de nuestro portal o aplicación.

De la misma forma, a cada Perfil de Usuario se le puede definir un módulo (o cualquier otra URL) como página de inicio. De tal forma, que cuando un usuario se identifica se le muestra ese módulo como página inicial. Si un Usuario tiene varios Perfiles, se le muestra la página inicial del primer perfil con este valor configurado.

## 4. Usuario

Un Usuario debe utilizar este sistema como una aplicación Internet, es decir utilizando un navegador.

Se puede utilizar de forma no identificada, accediendo a las partes (bloques y módulos) que el administrador le permite (públicos).

En la instalación inicial, RAD dispone de grupos de "noticias" y un "foro de discusión" públicos, donde los usuarios no identificados pueden ver todos los mensajes pero no pueden enviar mensajes al foro, ni comentar noticias.

Si se utiliza de forma identificada (después de introducir usuario y clave en la parte de "Usuario"), se permite acceder a más funciones.

Una vez identificado aparecerán las nuevas funciones a las que el usuario tiene perfil. Para abandonar la sesión se debe pulsar sobre el enlace "Salir" (esto invalida la sesión con el servidor).

Dentro de las opciones propias del usuario existe la opción de "Datos de Usuario" donde se le muestran sus datos, y desde la cual puede cambiar su Clave, y cambiar el aspecto que desee utilizar cuando entre o el idioma.

Las funciones de usuario, incluidas en la arquitectura básica de RAD son:

- Datos de Usuario (cambio de clave, aspecto, idioma)
- Mensajería Privada y opcionalmente Correo Electrónico de Internet
- Artículos (páginas estáticas HTML, visibles en el Portal/Aplicación, y agrupados en Secciones).
- Recursos propios y comunes (enlaces, imágenes, ficheros)
- Noticias. Contenidos de grupos de noticias (públicos o privados).
- Opiniones. Mensajes en Foros de Discusión (públicos o privados) y sus réplicas.
- Agenda-Calendarario de eventos y tareas.
- Libreta de Direcciones, Teléfonos y Email.

Todos los contenidos y mensajes permiten anexar ficheros (multimedia, ...).

Adicionalmente RAD dispone otras funciones de usuario:

- Encuestas
- Banner (textos o imágenes publicitarias)
- Estadísticas de uso
- Chat ...

## **4.1. Datos de Usuario**

Este módulo permite cambiar la clave, el idioma y el aspecto del usuario.

## **4.2. Mis Mensajes, Mi Agenda, Mi ...**

Cada una de estas opciones de usuario permite el acceso a un tipo de contenido propio de cada usuario.

Estas opciones además de las mencionadas son:

- Mensajería Privada. Opción que permite gestionar los mensajes propios. Leer los recibidos, enviar a otros, borrar... Y Correo Electrónico de Internet
- Artículos (páginas estáticas HTML, visibles en el Portal, y agrupados en Secciones).
- Recursos propios y comunes (enlaces, imágenes, ficheros)
- Noticias. Contenidos de grupos de noticias (públicos o privados).
- Opiniones. Mensajes en Foros de Discusión (públicos o privados) y sus réplicas. Presenta los mensajes a foros enviados por este usuario, cambiarlos, borrarlos, enviar nuevos...
- Agenda-Calendario
- Libreta de Direcciones, Teléfonos y Email.

Con cada tipo de contenido el usuario puede crear, modificar, borrar y listar los registros.

## **5. Desarrollo**

En este capítulo se describe como crear aplicaciones (conjunto de módulos) PHP para gestionar tablas de datos (SQL), y realizando las funciones deseadas.

Para ello, se deben utilizar las herramientas de desarrollo que aparecen en el menú "Adm. RAD", mencionadas en el capítulo de Administración.

### **5.1. Gestión de Base de Datos SQL**

Permite gestionar la base de datos de la aplicación RAD, y sus bases de datos copia o respaldo (aquellas que tienen el mismo nombre raíz).

Esta herramienta permite crear la base de datos acorde al diseño de datos de la aplicación. Es el primer paso del desarrollo de la aplicación, después de haber realizado el Análisis y el Diseño de Datos.

También permite modificar la estructura durante la fase de afinamiento o mantenimiento de la aplicación, crear volcados de la base de datos completa o de una tabla, etc.

Es una versión modificada de phpMyAdmin, convertida a módulos de RAD para que funcione el control de perfiles, y los aspectos de RAD.

### **5.2. Gestor de Ficheros Remoto (filemanager)**

Es un gestor de ficheros del servidor desde un navegador remoto. Permite enviar al servidor (de forma remota) ficheros, modificarlos, borrarlos y visualizarlos. También crear, renombrar o borrar directorios.

Muy util en la fase de desarrollo.

### **5.3. Ejecución Remota de Comandos (shell)**

Es una aplicación que permite ejecutar comandos en el servidor, desde un navegador, de forma remota.

Es una versión modificada del paquete PHPSHELL. Con esta herramienta podemos crear paquetes comprimidos en el servidor para descargarlos de forma remota, cambiar perfiles a ficheros o directorios, etc.

## **5.4. Generación Automática de Módulos phpRAD**

Es la aplicación que permite generar módulos php de forma automática, para gestionar una tabla de SQL. Y es la parte que aporta mayor potencia a la herramienta. Permitiendo incluso que se construya a sí misma (las partes de administración de datos).

Un módulo es un programa que gestiona una tabla de datos y realiza una función determinada. Es la unidad mínima de ejecución independiente.

Una aplicación se compone de un conjunto de módulos. Y son estos módulos los que se les asignan permisos de ejecución para los usuarios. Todo proceso mínimo e indivisible de la aplicación debe ser realizado por un módulo.

Un módulo puede ser desarrollado de forma manual, automática (por este generador) o de forma combinada, es decir generado de forma automática y posteriormente retocado de forma manual. La herramienta permite que un módulo generado de forma automática pueda ser retocado su comportamiento, mediante el uso de funciones externas.

El módulo puede ser generado para ser utilizado dentro de RAD o como una aplicación Internet independiente con sus propios controles de acceso, etc.

Esta aplicación es muy flexible, y con ella se ha generado toda la parte de administración de las tablas básicas de RAD.

Las aplicaciones creadas pueden ser muy complejas (permite tablas relacionadas, funciones propias utilizando el concepto de "función" o seudocampo de una tabla). Todo el desarrollo se realiza desde un navegador de forma local o remota.

El desarrollador solo necesitaría crear manualmente las funciones propias (campos complejos cuyas funciones no tiene RAD).

## **5.5. Como crear una Aplicación**

Para gestionar todas las tablas de una base de datos se debe invocar el modulo de Generador Automático de Módulos con la opción de "Crear Aplicación Base de Datos", una vez seleccionada esta opción nos aparecerá una pantalla donde se nos muestran todas las tablas de la base de datos, y se nos solicita el nombre del subdirectorío de "modules" donde se van a crear los módulos que corresponden a cada tabla. Elegido el nombre del directorío, y después de pulsar el botón Aceptar, aparecerá una nueva pantalla donde se muestran los campos de cada tabla, y se nos muestra el botón "Crear Aplicación" que nos solicita la confirmación para crear todos los módulos.

Se creará un módulo para gestionar cada tabla. Posteriormente se deben editar aquellos módulos que se deseen modificar para completar el comportamiento final de la aplicación. Esta parte compone la fase de desarrollo de la aplicación. Para completar la aplicación y que un usuario pueda utilizarla, se deben definir cada uno de los módulos con los permisos pertinentes, tal y como se describe en el capítulo de Administración.

Realizados estos pasos se puede decir que la aplicación ya está finalizada para su uso. Aunque es previsible que surjan peticiones de cambios y mejoras, lo cual pasa a ser la fase de refinamiento progresivo o mantenimiento de la aplicación. Esta es la base de la programación extrema y de las técnicas de prototipado rápido.



Por todo ello, se concluye que ésta es una herramienta muy adecuada para las técnicas de desarrollo y mantenimiento rápido de aplicaciones informáticas.

### **5.5.1. Como Gestionar una Tabla de Datos**

Para gestionar una única tabla de datos el Generador Automático de Código dispone de la opción "Crear Módulo Tabla". Con esta opción se creará un único módulo para gestionar una tabla. Una vez pulsada la opción se muestra una pantalla donde se debe seleccionar la tabla a gestionar y el directorio donde se creará el módulo. Una vez pulsado el botón "Aceptar" aparecerá una nueva pantalla donde se muestran los campos de la tabla y se pide confirmar la creación con el botón "Crear Módulo".

Una vez pulsado el botón se creará el módulo deseado. A partir de este momento el módulo puede ser usado o modificado acorde a nuestras necesidades.

### **5.5.2. Como Adaptar el Comportamiento de un Módulo Automático**

Los módulos generados de forma automática, tienen un comportamiento muy similar, y no interpretan el significado de los campos de las tablas, ni las relaciones de estos campos con otras tablas, por ello es habitual el que sea necesario adaptar su comportamiento.

Para modificar el comportamiento de los módulos creados por el Generador Automático se dispone de un conjunto de métodos:

- Funciones Externas: permiten agregar o cambiar el comportamiento del módulo. Son funciones PHP desarrolladas manualmente por el desarrollador.
- Ocultar o agregar opciones en los menús y submenús.
- Evitar que un Registro se muestre en el Listado. Cambios de filtro.
- Ocultar o Agregar Información a un Registro en el Listado.
- Modificar la Ventana de Selección de Opciones (popup).

## **5.6. Aviso de Seguridad**

Al igual que se comentó que el programa `install.php` debe ser eliminado o anulados todos sus permisos, se debe tener en cuenta que las herramientas de desarrollo no se deben tener disponibles en el servidor accesible desde Internet (por motivos de seguridad).

Si alguien descubriese nuestra clave de usuario de desarrollo, podría cambiar nuestro sistema o crear nuevas aplicaciones, borrar datos, etc. Se recomienda tener estas herramientas en un servidor de desarrollo inaccesible desde Internet, y una vez creadas las aplicaciones copiarlas con los datos al servidor de Internet.

## **5.7. Desarrollo de una Aplicación Tipo**

El desarrollo de una aplicación tipo utilizando "phpRAD", se debe realizar siguiendo estos pasos:

- 1.- Análisis de funciones básicas de la aplicación (facturación, clientes, contabilidad, ...).
- 2.- Diseño de modelo de datos (entidades: clientes, proveedores, ventas, compras, asientos, cuentas, ... relaciones) y definición de las tablas del modelo. Crear este diseño con el Gestor de Base de Datos.
- 3.- Generación automática de módulos de gestión de todas las tablas definidas en el punto

anterior, y modificación de dichos módulos para incluir el comportamiento deseado de la aplicación, incluyendo las relaciones de las tablas.

4.- Desarrollo manual de funciones propias (funciones de totales, gráficas, informes especiales), e inclusión en los correspondientes módulos.

5.- Generación de un Paquete para instalar. Una vez desarrollada una aplicación se debe volcar su base de datos (mysqldump BASEDEDATOS > BASEDEDATOS.sql) y comprimir todo desde el directorio raíz (tar cvfz paquete.tgz \* ) para generar un paquete que se puede distribuir para instalar, tal y como se comentó en el capítulo de Instalación.

Antes de generar el paquete debe ponerse el programa install.php correctamente, para que se pueda hacer una instalación automática. Después de generar el paquete debe eliminarse (o eliminar sus permisos) el programa install.php del directorio original.

La forma más cómoda y segura de generar un paquete es copiar todo a un nuevo directorio de trabajo y en él poner el programa install.php, y eliminar todos los ficheros que no correspondan a la aplicación (files, privado, ...), hacer el volcado de la base de datos (mysqldump ...) , y por último generar el paquete comprimido (tar ...).

Un ejemplo de aplicación compleja es **GIE** (<http://gie.edisaopensource.es/>) que permite gestionar un conjunto de tablas (GIE\_\*) y realiza las funciones de una aplicación de Gestión Integral de Empresa (clientes, facturación, contabilidad, ...) desde un navegador. Se compone de casi 200 tablas con sus correspondientes módulos de gestión.

## **Anexo 1.1 Funciones Disponibles en Módulos dentro de RAD .**

### **Funciones PHP de Módulo language/lang-XXXXXX.php**

- translate(\$texto)

Traduce un texto al idioma utilizado por el usuario. Recibe como entrada el texto a traducir y devuelve el texto traducido.

### **Funciones PHP de Módulo ASPECTO/theme.php**

- themeheader()

- themefooter()

- themesidebox(\$TMP\_title, \$TMP\_content)

- OpenTable()

- CloseTable()

### **Funciones PHP de Módulo sqlDB.php**

- sql\_connect(\$host, \$user, \$password, \$db)

- sql\_logout(\$id)

- sql\_query(\$query, \$id)

- sql\_error(\$res)

- sql\_errno(\$res)

- sql\_insert\_id(\$link)

- sql\_affected\_rows(\$res)

- sql\_num\_rows(\$res)

- sql\_num\_fields(\$res)

- sql\_fetch\_row(&\$res, \$nr)

- sql\_fetch\_array(&\$res, \$nr)

- sql\_fetch\_object(&\$res, \$nr)

- `sql_free_result($res)`
- `converttosql($value, $type="Text")`

### **Funciones PHP de Módulo `functions.php`**

- `RAD_unlink($fich)`
- `RAD_delParamURL($URL, $param)`
- `RAD_delParamFORM($FORMHIDDEN, $param)`
- `RAD_delField($namefield)`
- `RAD_addField($index,$name,$props)`
- `RAD_userName($TMP_user)`
- `RAD_groupName($TMP_group)`
- `RAD_getUserProfile($profile,$uname)`
- `RAD_existTable($TMP_table)`
- `RAD_showText($value)`
- `RAD_showDateInt($value)`
- `RAD_showShortDateInt($value)`
- `RAD_showDateTimeInt($value)`
- `RAD_showShortDateTimeInt($value)`
- `RAD_lookup($TMP_table,$TMP_field,$TMP_key,$TMP_val)`
- `RAD_showDateTime($value)`
- `RAD_showShortDateTime($value)`
- `RAD_showTimeInt($value)`
- `RAD_showTime($value)`
- `RAD_showDate($value)`
- `RAD_showShortDate($value)`
- `RAD_gregorianDate($fecha)`

- RAD\_julianDate(\$fecha)
- RAD\_daysinmonth(\$month, \$year)
- RAD\_weekMonthYear2Date(\$week,\$month,\$year)
- RAD\_week\_to\_dates(\$weekyear, \$year)
- RAD\_showcalendar(\$year,\$month,\$week,\$day,\$URLROI)
- RAD\_showcalyear(\$ano,\$ROI)
- RAD\_showcalmonth(\$ano,\$mes,\$week,\$ROI)
- RAD\_showcalweek(\$ano,\$mes,\$week,\$weekyear,\$ROI)
- RAD\_showcalweekmonths(\$ano,\$mes,\$week,\$weekyear,\$ROI)
- RAD\_showcalday(\$ano,\$mes,\$dia,\$ROI)
- RAD\_showfield(\$fdtype, \$fextra="", \$value)
- RAD\_inputDateText(\$TMP\_name,\$value,\$type,\$onChange,\$formName="F")
- RAD\_inputDate(\$TMP\_name,\$value,\$type,\$onChange,\$formName="F")
- RAD\_inputTime(\$TMP\_name,\$value, \$type,\$onChange,\$formName="F")
- RAD\_inputTimeText(\$TMP\_name,\$value, \$type,\$onChange,\$formName="F")
- RAD\_inputDateInt(\$TMP\_name,\$value,\$type, \$onChange,\$formName="F")
- RAD\_inputDateIntText(\$TMP\_name,\$value,\$type, \$onChange,\$formName="F")
- RAD\_inputDateTimeInt (\$TMP\_name,\$value,\$type, \$onChange,\$formName="F")
- RAD\_inputDateTimeIntText(\$TMP\_name,\$value,\$type, \$onChange,\$formName="F")
- RAD\_inputTimeInt(\$TMP\_name,\$value,\$type, \$onChange,\$formName="F")
- RAD\_inputTimeIntText(\$TMP\_name,\$value,\$type, \$onChange,\$formName="F")
- RAD\_inputDateTime (\$TMP\_name,\$value,\$type, \$onChange,\$formName="F")
- RAD\_inputDateTimeText(\$TMP\_name,\$value,\$type, \$onChange,\$formName="F")
- RAD\_array2select(\$arr, \$sel = "")
- RAD\_array2selectMultiplecolor(\$arr, \$arrlevel, \$sel = "")
- RAD\_array2selectcolor(\$arr, \$arrlevel, \$sel = "")

- RAD\_array2selectMultiple(\$arr, \$sel = "")
- RAD\_array2radio(\$arr,\$TMP\_name,\$sel = "",\$onChange)
- RAD\_ampliatree(\$level,\$cmdSQLtree, \$pfieldparent, \$value, \$pfilterBase)
- RAD\_edfdplistdbtree (\$fieldname, \$fdtype, \$fextra="", \$onChange="", \$fcanbenull=true, \$value = "", \$fname = "", \$formName="F")
- RAD\_editfield (\$fieldname, \$fdtype, \$flength, \$fmaxlength, \$fextra="", \$TMP\_onChange="", \$fcanbenull=true, \$value = "", \$fname = "", \$formName="F")
- RAD\_die(\$msg)
- RAD\_logError(\$TMP\_err)
- RAD\_printLog(\$cmdSQL)
- RAD\_grabaBDHistorica (\$TMP\_user,\$TMP\_idmod,\$TMP\_time,\$TMP\_table,\$TMP\_operacion, \$TMP\_valores,\$TMP\_where)
- RAD\_creaBDHistorica(\$TMP\_dbname)
- RAD\_creaTBHistorica(\$TMP\_dbname,\$TMP\_table)
- RAD\_grabaLog (\$TMP\_user,\$TMP\_idmod,\$TMP\_time,\$TMP\_table,\$TMP\_operacion, \$TMP\_valores,\$TMP\_where)
- RAD\_nameFile(\$TMP\_fich)
- RAD\_nameSecure(\$TMP\_fich)
- RAD\_nameDownload(\$TMP\_fich)
- RAD\_microtime()
- RAD\_redondeaEuros(\$total)
- RAD\_primerFich(\$TMP)
- RAD\_resizeImg(\$TMP\_img, \$ancho, \$salto)
- RAD\_hideurlencodeFich(\$TMP)
- RAD\_urlencodeFich(\$TMP)
- RAD\_barraHTML (\$longtotal,\$desplazamiento,\$longbarra,\$colorborde,\$colorfondo,\$colorbarra, \$altura)
- RAD\_checkSQLDelete(\$cmdSQL)
- RAD\_checkPars()

- RAD\_numero (\$val,\$decimales)
- RAD\_str2num(\$TMP\_num)
- RAD\_GregorianToJD (\$month,\$day,\$year)
- RAD\_JDToGregorian(\$julian)
- RAD\_inDate(\$TMP\_date,\$day,\$week,\$month,\$year)
- RAD\_IMGShortCut(\$TMP\_idmod,\$TMP\_par0,\$TMP\_func)
- RAD\_CodigoBarras(\$code, \$width, \$height)
- RAD\_CodigoBarrasEstilo(\$code, \$width, \$height, \$style, \$type, \$xres)
- RAD\_CodigoBarrasSimple(\$valor, \$ancho, \$salto)
- RAD\_openURL(\$TMP\_URLfile)
- RAD\_sendMail(\$V\_to,\$V\_cc,\$V\_bcc,\$V\_from,\$V\_subject,\$V\_body,\$V\_file)
- RAD\_convertHTML2CSV(\$TMP)
- RAD\_convertHTML2TXT(\$TMP)
- RAD\_backupDB(\$sourcedbname,\$targetdbname)
- RAD\_dup\_table\_def(\$sourcedb, \$targetdb, \$sourcetable, \$targettable)
- RAD\_dup\_table\_content(\$sourcedb, \$targetdb, \$sourcetable, \$targettable)
- RAD\_validUploadFile(\$file)
- RAD\_saveHTML(\$table,\$field,\$where)
- RAD\_copyFile(\$namefile, \$file)
- RAD\_saveFileContent(\$table,\$field,\$where,\$namefile,\$content)
- RAD\_sustituyeVarPHP(\$literal)
- RAD\_setFieldProperty(\$fieldList,\$propertyList)
- RAD\_debug(\$texto)
- RAD\_debugClear()
- RAD\_tree (\$TMP\_URL, \$TMP\_table, \$TMP\_defaultfilter, \$TMP\_orderby, \$TMP\_showfields, \$TMP\_field, \$TMP\_fieldparent, \$TMP\_idreg)

- RAD\_showtree (\$TMP\_URL, \$TMP\_table, \$TMP\_defaultfilter, \$TMP\_orderby, \$TMP\_showfields, \$TMP\_field, \$TMP\_fieldparent, \$TMP\_idreg, \$TMP\_level, \$TMP\_par)
- RAD\_dirsize(\$directory)
- RAD\_HTTP\_Post(\$URL,\$data, \$referrer="", \$timeout="")
- RAD\_reloadURL(\$params)
- RAD\_doZIP(\$value)
- RAD\_mailSocket(\$smtpserver,\$to,\$subject,\$message,\$headers,\$auth)
- RAD\_sqlUniqueResult(\$query)
- RAD\_sqlUniqueRow(\$query)
- RAD\_print\_var(\$var)
- RAD\_trace(\$msg)
- RAD\_moveField(\$fieldname,\$offset)

### **Funciones de Módulo mainfile.php**

- is\_admin()
- is\_user()
- is\_modulepermitted(\$TMP\_idmod, \$TMP\_name, \$TMP\_file)
- getSessionVar(\$ParamName)
- setSessionVar(\$ParamName, \$ParamValue, \$lifetime=0)
- getTheme()
- head()
- footmsg()
- foot()

### **Funciones Javascript de Módulo mainfile.php**

- RAD\_OpenW(pagina,x,y)
- RAD\_showL(url,x,y)



## Anexo 1.2 Funciones PHP Disponibles en Módulos Automáticos

Además de las funciones anteriores, y dado que todos los módulos automáticos incluyen el fichero `modules/phpRAD/RAD_common.php`, las funciones de este módulo también están disponibles para los módulos desarrollados de forma automática por RAD. Estas funciones son:

– `RAD_setField($field,$val)`

Almacena en el campo `$field` el valor `$val`. Este campo es uno de los campos definidos dentro del módulo automático.

– `RAD_getField($field)`

Devuelve el valor del campo `$field`. Este campo es uno de los campos definidos dentro del módulo automático.

– `RAD_menu_on($text)`

Pone en una pestaña de menú resaltada el texto `$text`

– `RAD_menu_off($text)`

Pone en una pestaña de menú no resaltada el texto `$text`

– `RAD_submenu_on($text)`

Pone en una pestaña de submenú resaltada el texto `$text`

– `RAD_submenu_off($text)`

Pone en una pestaña de submenú no resaltada el texto `$text`