

# Proyecto final de curso

Sergi Ruiz Sánchez

15 de Junio 2020

Este proyecto ha sido realizado de forma individual por el alumno Sergi Ruiz Sánchez.

## 1 Usuarios que componen la BD original

Al acceder por primera vez a index.php, se creará de forma automática toda la base de datos, estableciendo los campos oportunos para cada tabla, y en concreto , para la tabla de usuarios, los siguientes valores:

- **Usuario:** "Anónimo"      **Contraseña:** null      **Rol:** no registrado  
(Usuario que creamos para los que no se encuentran registrados; si intenta registrarse en el sistema el acceso le será denegado)
- **Usuario:** "admin"      **Contraseña:** "admin"      **Rol:** admin  
Administrador por defecto del sistema. Al originarse la BD, las 5 recetas que hay son suyas
- **Usuario:** "El Cocinillas"      **Contraseña:** "usuario"      **Rol:** usuario  
Único usuario por defecto del sistema

## 2 Ficheros de restauración

Dentro del directorio *backup/* encontraremos dos ficheros:

- **originalDB.txt:** Es la base de datos original, la cual va a cargarse e instalarse en el servidor de forma automática cuando hagamos nuestra primera visita a index.php
- **backupDB.txt:** Este fichero también almacena en primera instancia lo mismo que *original.txt*, con la diferencia de que cuando vayamos a realizar un backup del sistema, los datos del fichero son machacados por toda la información de la BD en ese momento, de forma que si por algún motivo no se nos desplegase la ventana que habilita la descarga del fichero (error que no ocurre, pero por prevención) sabemos donde poder encontrarlo.

Del mismo modo, podemos encontrar en cualquier momento todos los ficheros que hemos subido para restaurar el sistema en el directorio *uploadedFiles/*

### 3 Descripción de la base de datos

La base de datos empleada ha sido prácticamente la proporcionada por el profesor de la asignatura, solo ha cambiado el nombre de un campo ("password" por "contraseña" en la tabla de usuarios) y el tipo de otro("blob" por "varchar(200)" en el campo "foto" de la tabla de usuarios), por lo que adjunto el modelo relacional ya dado para el desarrollo de esta práctica con los cambios pertinentes.

Modelo Entidad-Relacion:

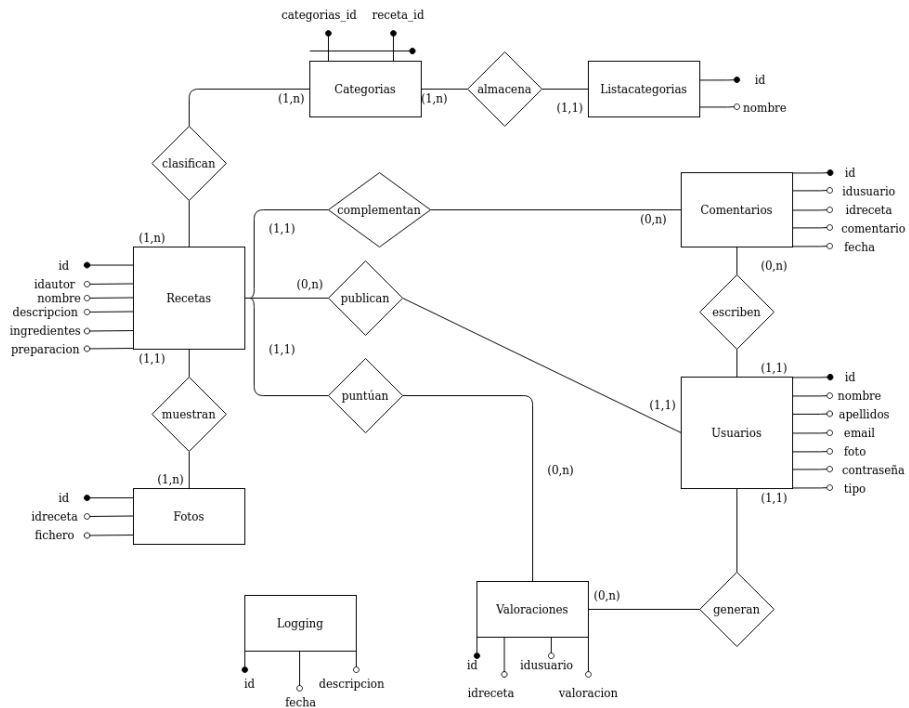


Figure 1: Modelo Entidad-Relación

Modelo relacional asociado:

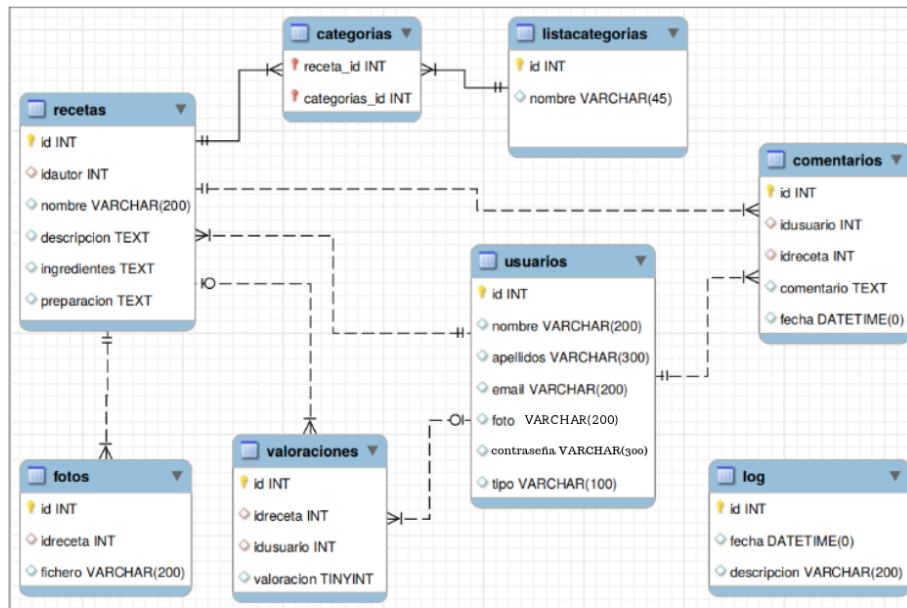


Figure 2: Modelo Relacional

## 4 Apartados opcionales realizados y explicaciones a tener en cuenta

- **Opcional 1:** Realizado, pero a la hora de crear/editar recetas, en vez de dispuestos en checkboxes, el usuario introduce las categorías a través de un campo de texto.
- **Opcional 2**
- **Opcional 3:** He optado por la segunda vertiente. Cuando alguien se crea su propio usuario, se almacena en la tabla de usuarios con tipo "pendiente". Cuando un usuario administrador se loguea, dispone del menú "Validar usuarios", donde se despliega una lista de los usuarios pendientes de ser validados. Ahí el admin puede decidir si eliminar el usuario (desaparece de la BD), o le da un rol distinto de "pendiente", momento a partir del cual pasa a formar parte del sistema como tal y le será permitido loguearse. Del mismo modo, en el menú "Gestión de usuarios", los administradores pueden poner el campo "tipo" de cualquier usuario (excepto del suyo, para evitar que podamos quedarnos sin administradores en el sistema) a "pendiente", lo cual hará que desde ese momento dicho usuario ya no pueda loguearse hasta que se le vuelva a cambiar el tipo, y será movido a la lista de "pendientes de validar" mencionada anteriormente. Esto es útil si, por ejemplo, queremos hacer una baja temporal de algún usuario sin que este pierda sus datos ni las recetas asociadas que

tiene. Otra cosa a destacar es, que si se elimina un usuario que había publicado recetas, estas se mantienen en el sistema, pero pasan a asociarse a la "id" del usuario "Anónimo", por lo que al verlas pondrá que este es su autor.

- **Opcional 4-11:** En esta mejora he usado las tecnologías de JavaScript y AJAX, completando también de este modo el apartado opcional número 11. La solución que he aportado es la siguiente: Cuando pulsamos en el botón para añadir las imagenes (tiene el campo "multiple", por lo que pueden añadirse varias imagenes a la vez) JavaScript captura esas imagenes, y mediante AJAX las subimos al fichero *tmp/*, el cual almacena todas las imagenes relativas a los eventos antes de subirlos al fichero definitivo(*Images*). Posteriormente, devuelve todas las fotos de ese directorio a JavaScript, el cual las introduce en el HTML, por lo que de forma totalmente dinámica, al pulsar en una imagen se nos va a mostrar por pantalla dicha imagen junto con el resto de imagenes del fichero *tmp/* (en el caso de modificar la receta tenemos dos contenedores, uno de las imagenes ya incluidas, en el directorio *Images/*, y otro con las imagenes pendientes de incluir, en *tmp/*). Esta imagenes poseen un botón "Eliminar", el cual, al pulsarlo, hace que eliminemos las imagenes del directorio correspondiente.

Al añadir las imagenes podemos ver que el botón "Eliminar" aparece a disabled para las imagenes procedentes del fichero *tmp/*, pero eso no es ningún problema, ya que, aunque todos los campos estén correctos, se le concede al usuario "una vuelta más" al formulario, donde los botones ya si están habilitados, de forma que pueda eliminar las imagenes que estime oportunas.

- **Opcional 6:** He optado por la segunda opción, de forma que al pulsar en cualquier imagen se nos despliega una nueva página, la cual contiene un botón para volver a la receta.
- **Opcional 7:** Esta opción está implementada en el menú "Resataurar base de datos", y los ficheros subidos se almacenan en el directorio *uploaded-Files/*, como hemos mencionado previamente. Solo se permiten ficheros con extensión .txt, y así se lo hará saber la web si intenta introducir otro tipo de fichero.
- **Opcional 8:** Disponible en el menú "Borrar todos los datos". Se le preguntará si está seguro de borrar los datos, y en el caso de pulsar sí, todo lo que contenga la base de datos será eliminado, con la excepción de los usuarios "admin" y "Anónimo".
- **Opcional 10:** El mecanismo de autoinstalación consiste en la carga en la base de datos del fichero *originalDB.txt* mencionado al inicio del documento.

Otras cuestiones a tener en cuenta:

- Cuando borramos una receta a mano, sus imágenes van a parar al directorio *deletedImages/* en vez de ser eliminadas directamente. Esto está pensado para que en el caso de querer restaurar las recetas y entre ellas hay una eliminada a mano, puedan moverse sus imágenes de ese directorio al directorio *Images/*, para que puedan volver a ser visibles. Del mismo modo, recordar introducir ahí todas las imágenes que hubiera antes al realizar una restauración, o se verán como un enlace roto.
- Las imágenes de los usuarios están en el directorio *usersProfiles/*, y el nombre de cada imagen sigue el formato "nombreusuarioapellidosusuario.jpg". Cuando vamos a cambiar nuestros datos personales, se crea una copia de esta en un directorio temporal, *tmpUsers/*. La imagen mostrada en el formulario proviene de ahí, y ahí es donde se suben las fotos que seleccionamos, de forma que el usuario pueda ver como su foto cambia y como se vería, pero sin que esta esté cambiando realmente en el directorio final. Se hace de este modo para que, si el usuario tiene seleccionada actualmente otra foto que no es la suya actual, y decide salir del formulario sin guardar, no se realicen los cambios en la foto. Finalmente, cuando aceptamos los cambios, esa foto en el directorio temporal se mueve al final, sustituyéndola por el anterior, y cambiando el nombre por el nombre actual del usuario (por si ha cambiado en el proceso de edición de datos, que se siga viendo).
- Se ha seguido el esquema MVC, donde podemos ver que el modelo es el directorio "db.php" y los controladores el resto de ficheros php. Uso el sistema de plantillas Twig, por lo que la vista queda relegada a los ficheros del directorio *TW/*, que, aunque todos contienen la extensión *html*, realmente no son ficheros de tal tipo.
- Al principio veremos la lista de recetas más comentadas vacía. Esto se debe a que en esa lista no se tiene en cuenta a las recetas con 0 comentarios, por lo que cuando empezamos a introducir comentarios en las recetas veremos como se completa la lista. También se ha dado algunos casos donde recetas con el mismo número de comentarios cambian su posición en la lista de + comentadas si ambas están dentro (ej, si la segunda y la tercera tienen 2 comentarios, a veces aparecen cambiadas de orden). No es relevante, pero cabe comentarlo por si se detecta algún cambio raro en la lista al refrescar la página sin haber introducido comentarios.
- Para no tener problemas con los nombres de los usuarios creadores de recetas y anónimos (problemas tipo que un mismo usuario publique recetas con varios nombres), en todos los formularios (creación/edición de receta y contacto), añadimos de forma automática el nombre y apellidos del usuario (o "Anónimo"), y lo metemos en un campo con valor "disabled", para que no se pueda modificar.

- Si no se introducen imágenes en la receta al crearla se establece una imagen por defecto ,*no-imagen*. En el momento en el que añadamos una imagen, ya sea en la creación o modificación de la receta, esa *no-imagen* desaparece y es sustituida por las imágenes correspondientes de la receta.
- Se ha tenido en cuenta la seguridad, de forma que por ejemplo un usuario no registrado no pueda acceder a páginas que su permiso no le corresponde. Lo mismo ocurre con los usuarios registrados no administradores.
- También se ha tenido en cuenta la casuística de que dos usuarios pueden compartir nombre o apellidos, pero no ambos a la vez. Si intentamos crear un usuario/modificar nuestro usuario y añadirle el nombre y apellidos de otro ya existente, se nos indicará que ese usuario ya existe en el sistema, por lo que debemos escoger otra combinación nombre/apellidos.
- A la hora de meter imagenes en el directorio *tmp/*, tengo en cuenta si las imagenes ya existen o no para no introducir más de una vez la misma imagen a la vez. Sin embargo, si la imagen ya existe ( está en *Images/*), si puede volver a introducirse.
- Cuidado si dos recetas contienen la misma imagen. La eliminación de una de las recetas hará que sea imagen pase al directorio *deletedImages/* lo cual puede suponer un problema