

# Como se hizo

Alumno :Antonio David López Machado

## 1.Estructura principal

Nuestro index está dividido en 3 secciones. Las cuales son :

- **Header.php** : Que contiene la información del header de nuestra web.
- **Content.php** : Que contiene la información del body de nuestra web. Para cambiar de contenido la web utilizaremos una variable `get(category)` la cual indicará que fichero php mostrar nuestro body.
- **Footer.php** : Que contiene la información del footer de nuestra web.

## 2.Estructura de clases

He desarrollado 4 clases las cuales nos permiten gestionar de una forma más sencilla las diferentes tablas de nuestra base de datos. Las clases desarrolladas han sido:

- **DateBaseAccess** : Nos permite gestionar la conexión y las consultas sobre la base de datos.
- **User** : Nos permite gestionar la tabla de usuario.
- **Entry** : Nos permite gestionar la tabla de entradas.
- **Commentary** : Nos permite gestionar la tabla de comentarios.

## 3.Base de datos

Se ha desarrollado una base de datos sencilla con 3 tablas. Dichas tablas son

- **User** : La cual contiene la información de los usuarios registrados en el sistema
- **Entry** : La cual contiene la información de las entradas creadas en el sistema.
- **Commentary** : La cual contiene la información de las comentarios creados en el sistema.

## 4. Descripción de la funcionalidad relevante

### 4.1 Registro de un usuario

El registro de un usuario en nuestro sitio web tiene 3 fases.

- Fase 1 - El usuario introduce su información en el formulario de registro
- Fase 2 - Nuestro javascript valida dicha información ( no haya campos vacíos )
- Fase 3 - Nuestro script de registro del usuario realiza la inserción de la información del usuario en la tabla user de nuestra base de datos.

El cliente quedará registrado y podrá acceder al sitio web con su usuario.

### 4.2 Login del usuario

El acceso del usuario a nuestro sitio web tiene 3 fases.

- Fase 1 - El usuario introduce su información de usuario en el panel de logueo.
- Fase 2 - Nuestro javascript valida dicha información ( no haya campos vacíos )
- Fase 3 - Nuestro script de logueo comprueba que los datos pertenezcan a un usuario de la base de datos y crea la sesión con la información de dicho usuario.

El cliente visualizará la portada de nuestro sitio web.

#### **4.3 Cierre de sesión**

El cierre de sesión de un usuario en nuestro sitio web tiene 2 fases.

- Fase 1 - El usuario logueado pulsa la opción "Desconectar".
- Fase 2 - Nuestro script de cierre de sesión destruye la sesión con la información de dicho usuario.

El cliente visualizará la página de inicio de sesión de nuestro sitio web.

#### **4.4 Cambiar información de usuario**

El cambio de la información de un usuario en nuestro sitio web tiene 3 fases.

- Fase 1 - El usuario introduce su nueva información en el formulario de actualización
- Fase 2 - Nuestro javascript valida dicha información( no haya campos vacíos )
- Fase 3 - Nuestro script de actualización de un usuario realiza la actualización de la información del usuario concreto.

El cliente habrá cambiado su información y podrá visualizar dichos cambios.

#### **4.5 Añadir entrada**

Para añadir una nueva entrada en nuestro sitio web tiene 3 fases.

- Fase 1 - El usuario introduce la información de la entrada en el formulario de entrada.
- Fase 2 - Nuestro javascript valida dicha información( no haya campos vacíos )
- Fase 3 - Nuestro script de creación de entradas inserta una nueva entrada en nuestra base de datos con la información enviada por el usuario.

El cliente podrá ver su nueva entrada en su biografía, portada y en la página de creación de entrada.

#### **4.6 Añadir Comentario**

Para añadir un nuevo comentario en una entrada de nuestro sitio web tiene 3 fases.

- Fase 1 - El usuario introduce la información del comentario en el formulario de comentario de la entrada en la que quiera comentar.
- Fase 2 - Nuestro javascript valida dicha información( no haya campos vacíos )
- Fase 3 - Nuestro script de creación de comentarios inserta un nuevo comentario en nuestra base de datos con la información enviada por el usuario.

El cliente podrá ver su nuevo comentario en la entrada de dicho usuario.

#### **4.7 Mostrar usuarios conectados**

Para visualizar la usuarios conectados hemos introducido un campo booleano en nuestra clase user. De esta forma cuando se loguea un usuario este campo se pondrá a true y los otros usuarios podrán verlo como usuario activo. Cuando un usuario cierre su sesión este campo se pondrá a false y los otros usuarios ya no podrán verlo como usuario activo.

## **5.Aspectos innovadores**

Para esta sección se ha investigado la forma de determinar el número de horas que han pasado desde la creación de una entrada o comentario siempre y cuando estemos en el mismo día.