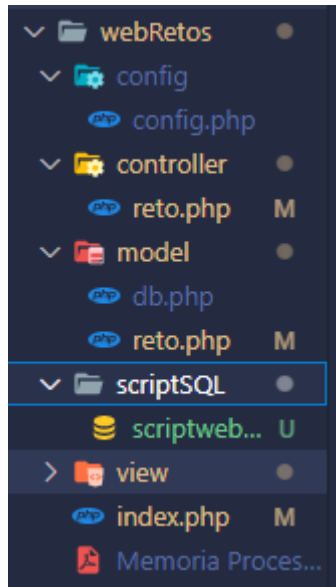


Rivera Salgado Sergio

Estructura mvc(Con index aplicado)



Proceso de alta de retos

- Introduce a la perfección con datos correctos.
- Si al introducir el nombre está repetido hacemos un control de errores del servidor para que la vista nos muestre un mensaje personalizado sobre el error.
- Si al introducir una fecha de inicio mayor que una de fin hacemos un control de errores del servidor para que la vista nos muestre un mensaje personalizado sobre el error.
- Si ingresamos campos vacíos y la columna no permite nulos también controlamos los errores del servidor y se le envía a la vista un mensaje explicando el error.
- Cuando el campo viene vacío nos interesa que se guarde en la base de datos como null, por lo tanto se hizo un control de campos vacíos para que se registraran si fueran posible como nulos, por ejemplo en el campo descripción.
- Para poder hacer un alta necesitamos consultar la tabla categorías , se hace y se despliega con un select en el proceso de alta.
- Se utilizan consultas preparadas por seguridad, menos para el select el cual utiliza query por temas de rendimiento.

Proceso de baja de retos

- Se elimina correctamente
- Para el proceso de eliminación pasamos por id a una vista de confirmación de eliminación, la cual te permite arrepentirte y cancelar el proceso o seguir adelante y eliminarlo. Una vez sigas adelante se eliminará el reto seleccionado por id. Se usa una consulta preparada para eliminar, por seguridad.

Proceso de listado de retos

- Se listan correctamente.
- El array donde se sitúan todos los datos que queremos llevar a las vista `$dataToView[]` nos permite listar los retos. Para llenar el array utilizamos: `$result->fetch_all(MYSQLI_ASSOC)` ¿Por qué? es simple, el método `fetch_all()` de la clase `result` nos da todas las filas de la consulta. Nosotros utilizamos el parámetro opcional `MYSQLY_ASSOC` para que nos lo dé en forma de array asociativo. Luego en la vista listar mediante un `foreach` recorreremos cada fila hasta el fin y en cada iteración mostramos los datos que nos interesan.
- Desde esta vista se nos permite editar, crear, eliminar y hacer filtrados por nombre y categoria.
- Podemos filtrar el listado por nombre y por categorías.

Por nombre:

```
$sql = "SELECT * FROM ".$this->table." WHERE nombre LIKE  
"."'".$busqueda."'" ;
```

Por categorías:

```
$sql = "SELECT * FROM ".$this->table." INNER JOIN  
".$this->table2." ON retos.idCategoria = categoria.idCategoria  
where categoria.nombreCategoria LIKE "."'".$busqueda."'" ;
```

Proceso de modificación de retos

- Se modifican correctamente.
- En primer lugar el formulario de modificación ya se encuentra relleno gracias al array `$dataToView[]` el cual se rellena si al iniciar la vista le enviamos el id del reto a modificar se hará una consulta por id a la base de datos y nos devolverá una fila, esta fila se mete en el array `$dataToView[]` el cual sí está lleno crearemos variables en la vista modificar con los datos del array y rellenaremos los campos con esas variables. El usuario podrá modificarlos y tendrá un control de errores exactamente igual que el proceso de alta. Por lo general el proceso de modificación es

exactamente igual técnicamente que el de alta, solo que se rellena el formulario con los datos del reto a cambiar.