

Mini documentacion del crud

Puesto que todo se ha dado en clase, voy a destacar los aspectos que considero más importantes, como son las tablas de la base de datos y el manejo de las imágenes.

Primero la base de datos

```
tablas.sql X
sql > tablas.sql > ...
  ▶ Run SQL
1 drop table if exists videojuegos;
2
  ▶ Run SQL
3 drop table if exists tienda;
4
  ▶ Run SQL
5 CREATE Table videojuego(
6     id int PRIMARY key AUTO_INCREMENT,
7     nombre varchar(20) NOT NULL,
8     descripcion text not null,
9     fecha date not null,
10    img VARCHAR(120) not NULL
11 );
12
  ▶ Run SQL
13 CREATE Table tienda(
14     id int PRIMARY KEY AUTO_INCREMENT,
15     nombre VARCHAR(10) NOT NULL,
16     videojuego_id INT,
17     constraint fk_tienda_videojuego FOREIGN KEY(videojuego_id) REFERENCES videojuego(id) on delete CASCADE on update CASCADE
18 );
```

Ahora a ver el funcionamiento de la creación de la imagen por defecto en el crear videojuego

```
cvideojuego.php X
public > videojuegos > cvideojuego.php
34
35 //COMPROBAMOS LA IMAGEN
36 //1. Vemos si se ha subido imagen
37 if(is_uploaded_file($_FILES['img']['tmp_name'])){
38     //Se ha subido algo, vemos si es una imagen
39     if(isImage($_FILES['img']['type'])){
40         //He subido una imagen
41
42         $nombreImg=$nombre."_".$fecha."_".$_FILES['img']['name'];
43
44         //Movemos la imagen
45         if(move_uploaded_file($_FILES['img']['tmp_name'], dirname(__DIR__, 1)."/img/$nombreImg")){
46
47             $imagen="../img/$nombreImg";
48
49             //Si llega aqui es que la imagen se ha movido correctamente
50
51         }else{
52             //No se ha podido mover la imagen
53             $_SESSION['imagen']="No se ha podido mover la imagen!!";
54             $exito=false;
55         }
56     }else{
57         //Lo que se ha subido no es una imagen
58         $_SESSION['imagen']="No se ha subido una imagen!!";
59         $exito=false;
60     }
61 }else{
62     //No se ha subido nada, guardamos la imagen por defecto
63     $RUTA_IMG="../img/default.png";
64     $imagen=$RUTA_IMG;
65 }
66
67 if($exito==true){
68     //todo está correcto, creamos el videojuego
69     (new Videojuegos)->setNombre($nombre)->setDescripcion($descripcion)->setFecha($fecha)->setImg($imagen)->create();
70 }
```

El nombre que se le da es una combinación del nombre del videojuego, su fecha de lanzamiento seguido del nombre del archivo, todo eso concatenado por '_'.

A la hora de borrar las imágenes cuando se borra el videojuego se hace de la siguiente forma

```
bvideojuego.php X
public > videojuegos > bvideojuego.php
1  <?php
2  session_start();
3
4  if(!isset($_POST['id'])){
5      header('Location:index.php');
6      die();
7  }
8
9  require dirname(__DIR__,2)."/vendor/autoload.php";
10 use TiendaVideojuegos\Videojuegos;
11
12 //Leemos el videojuego haciendo uso de su id
13 $elVideojuego=(new Videojuegos)->read($_POST['id']);
14
15 //Antes de borrar el videojuego, borramos la imagen que tiene en caso de que sea distinta de la default
16 //Primero vemos si la imagen que tiene es la default
17 if(str_contains($elVideojuego->img, "default.png")) {
18     //Si entra aqui quiere decir que es la imagen por defecto, no hacemos nada
19 }else{
20     //SI entra aqui quiere decir que tiene una imagen diferente, la borramos
21     unlink(dirname(__DIR__, 1)."/img/".$elVideojuego->img);
22     // basename("/home/usuario/doc1.pdf") = doc1.pdf
23 }
24
25 (new Videojuegos)->delete($_POST['id']);
26 $_SESSION['mensaje']="Videojuego borrado con exito";
27 header('Location:index.php');
```

Se hace al pulsar el botón de borrar el videojuego. Antes de ejecutar el delete, se comprueba si la imagen que tiene es la por defecto, si no lo es se borra y después se borra el videojuego.

Al momento de actualizar una imagen es la combinación entre la creación de la imagen y el borrar.

```
evideojuego.php X
public > videojuegos > evideojuego.php
46
47 //COMPROBAMOS LA IMAGEN
48 //1. Vemos si se ha subido imagen
49 if(is_uploaded_file($_FILES['img']['tmp_name']))){
50     //Se ha subido algo, vemos si es una imagen
51     if(isImage($_FILES['img']['type'])){
52         //He subido una imagen
53         $nombreImg=$nombre."_".$fecha."_".$_FILES['img']['name'];
54
55         //Movemos la imagen
56         if(move_uploaded_file($_FILES['img']['tmp_name'], dirname(__DIR__, 1)."/img/$nombreImg")){
57
58             $imagen="../img/$nombreImg";
59
60             //Si llega aqui es que la imagen se ha movido correctamente
61             //BORRAMOS LA IMAGEN QUE YA TIENE DE LA CARPETA PARA DAR PASO A LA NUEVA
62
63             //Primero vemos si la imagen que tiene es la default
64             if (str_contains($elVideojuego->img, "default.png")) {
65                 //Si entra aqui quiere decir que es la imagen por defecto, no hacemos nada
66             }else{
67                 //SI entra aqui quiere decir que tiene una imagen diferente, la borramos
68                 unlink(dirname(__DIR__, 1)."/img/".$elVideojuego->img);
69             }
70         }else{
71             //No se ha podido mover la imagen
72             $_SESSION['imagen']="No se ha podido mover la imagen!!";
73             $exito=false;
74         }
75     }else{
76         //Lo que se ha subido no es una imagen
77         $_SESSION['imagen']="No se ha subido una imagen!!";
78         $exito=false;
79     }
80 }else{
81     //No se ha subido nada, guardamos la imagen por defecto
82     $imagen=$elVideojuego->img;
83 }
```

Se comprueba que se ha subido algo, si no es así lo que se hace es asignar la imagen que ya tiene.

Ahora si se ha subido algo se comprueba que se ha subido una imagen mediante esto:

(También se usa en el crear videojuego)

```
//Nos dirá si el tipo de archivo es de una imagen o no
function isImagen($tipoArchivo){
    $tipos = ['image/gif', 'image/png', 'image/jpeg', 'image/bmp', 'image/webp'];
    return in_array($tipoArchivo, $tipos);
}
```

Si todo es correcto, se procede a crear su nombre y se mueve el archivo a la carpeta que hemos indicado.

Por último, si la imagen que tenía era la por defecto, no se hace nada. Por el otro lado si la imagen era otra, esta se borra para ser reemplazada por la nueva.

(Ahora muestro lo que sale en ciertos momentos del video que no se ven por culpa de la grabación de windows)

Lo primero a la hora de insertar la fecha de publicación

Nuevo videojuego

Nombre del videojuego

Descripcion

Fecha de lanzamiento

diciembre de 2021

L	M	X	J	V	S	D
29	30	1	2	3	4	5
6	7	8	9	10	11	12
13	14	15	16	17	18	19
20	21	22	23	24	25	26
27	28	29	30	31	1	2
3	4	5	6	7	8	9

BorrarHoy

Volver

Por cierto destacar que Faker tiene la opción de generar fechas aleatorias en el formato que admite mySql

```
public function generarVideojuegos($cantidad){
    if(!$this->hayVideojuegos()){ //Si no hay videojuegos los genera
        $faker=Faker\Factory::create('es_ES');

        $rutaImgDefecto="../img/default.png"; //Puesto que se cargan en la carpeta public/videojuegos, la ruta parte de dicha carpeta

        for($i=0;$i<$cantidad;$i++){
            $nombre=$faker->sentence(3);
            $descripcion=$faker->text(30);
            $fecha=$faker->date(); //Genera una fecha con formato yyyy-mm-dd (el que admite mysql)
            $img=$rutaImgDefecto;

            (new Videojuegos)->setNombre($nombre)->setDescripcion($descripcion)->setFecha($fecha)->setImg($img)->create();
        }
    }
}
```

Y por último el listado de videojuegos ya existentes a la hora de crear una nueva tienda

Nueva Tienda

Nombre de la tienda

Videojuego estrella

Aperiam quod nemo et

Aperiam quod nemo et

Cupiditate est error

Deserunt delectus in

Deserunt quod offici

Et ut ea.

Ferererree

Laudantium vitae qui

Minima quis quis ali

Natus minima id.

Nemo tenetur quia.

Odit voluptatem prov

Pepe

Provident accusantiu

Quaerat ut.

Qui voluptatibus qui

Reiciendis consequun

Repudiandae nemo atq

Repudiandae voluptat

Rerum non vel doloru

Rerum saepe eum perf