

Proyecto Fin de Ciclo  
Rafael Gómez Cabero - 2ºASIR  
Implantación de Aplicaciones  
Web

IES La Campiña

<b>1. Introducción</b>	<b>2</b>
<b>2. Configuración</b>	<b>2</b>
<b>3. Descripción de la aplicación</b>	<b>6</b>
<b>4. Diseño de la aplicación</b>	<b>6</b>
<b>5. Desarrollo de la aplicación</b>	<b>9</b>
conexion.php	9
index.html	9
pokemon.php	10
registrarPokemon.php	13
estadisticas.php	17
anadirPokemon.php	17
eliminar.php	21
combate.php	23
crearCombate.php	25
<b>6. Conclusión</b>	<b>31</b>

# 1. Introducción

Para el funcionamiento de nuestra aplicación usaremos Xampp con los servicios de Apache y MySQL, así como phpmyadmin para la gestión de la base de datos.

Para el diseño y la visualización de nuestra aplicación, usaremos bootstrap combinado con CSS para potenciar las partes de las que bootstrap no se pueda hacer cargo.

Finalmente, para controlar los cambios y las versiones de nuestra aplicación necesitaremos la ayuda de Git y de Github.

## 2. Configuración

Instalamos Visual Studio Code y Git. Para comprobar que Git está instalado, abrimos una cmd y ponemos el comando `git -version`.

```
C:\Users\Rafael>git --version
git version 2.41.0.windows.1
```

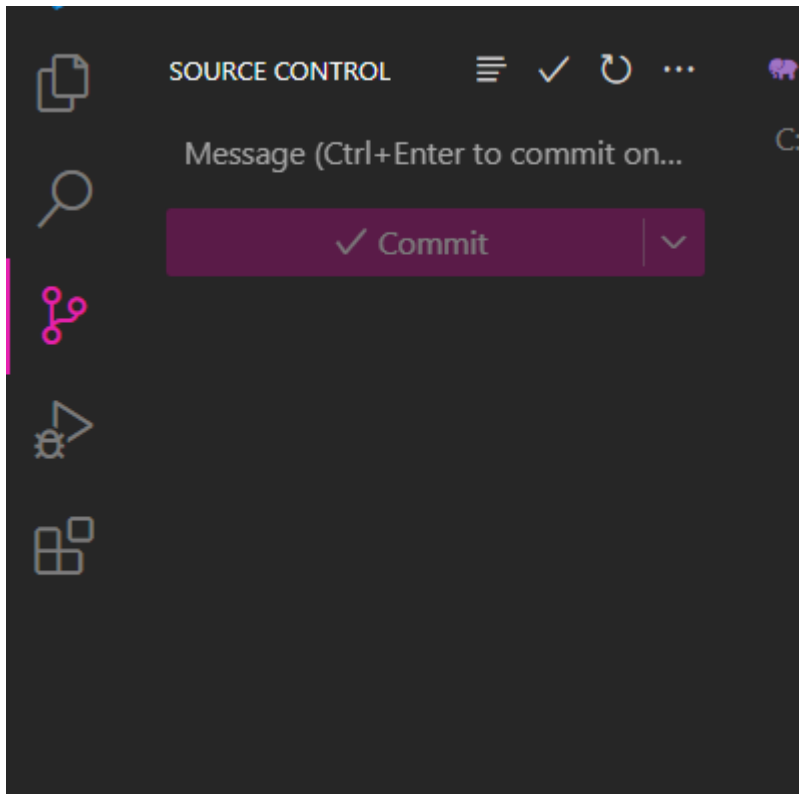
Configuramos ahora nuestro usuario y correo de Git.

```
C:\Users\Rafael>git config --global user.name "Rafael"
C:\Users\Rafael>git config --global user.email rgomcab714a@ieslacampina.es
C:\Users\Rafael>
```

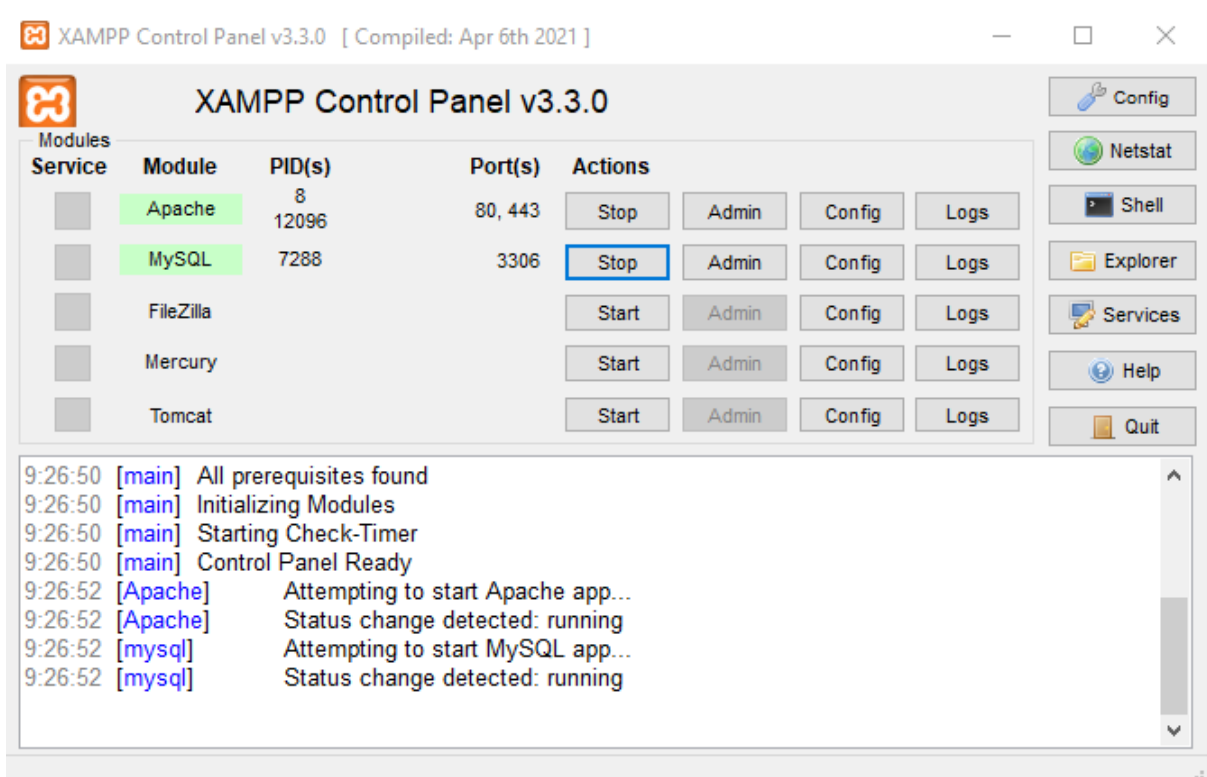
Ahora nos vamos a la carpeta donde queramos colocar el repositorio y lo añadimos a GitHub.

```
rafael@DESKTOP-BHIVESC MINGW64 /c/xampp/htdocs/PRUEBA/Proyecto IAW (mas
$ git remote add origin url https://github.com/rafgomcab/PFC-IAW.git
```

Una vez hecho esto, en Github nos vamos a nuestro repositorio y copiamos el link http que tenemos ahí. En VSC al establecer un repositorio introducimos ese link y así tendremos establecida la conexión entre nuestro entorno de desarrollo y nuestro repositorio.



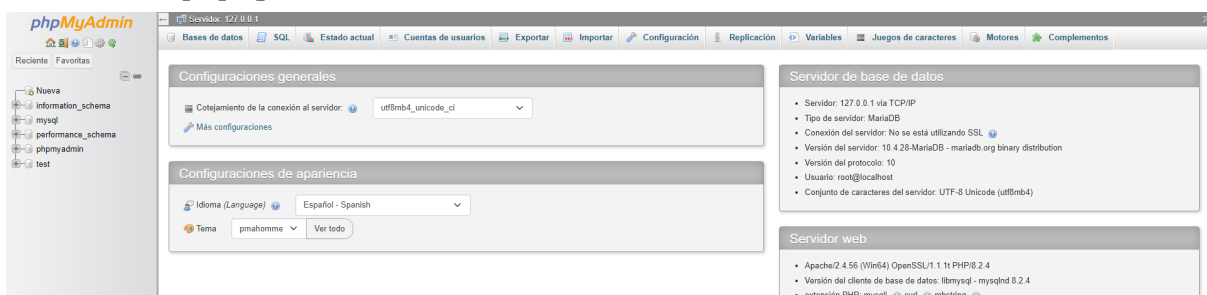
Lo siguiente será instalar Xampp y una vez instalado, ejecutarlo y activar los servicios de Apache y de MySQL.



Al entrar en VSC con xampp abierto nos pedirá la ruta del ejecutable de php. Se la introducimos y ya podríamos usar PHP

```
conexion.php index.php Settings settings.json X
C: > Users > Rafael > AppData > Roaming > Code > User > settings.json > ...
1 {
2     "workbench.colorTheme": "Lunar Pink Satellite",
3     "window.zoomLevel": 1,
4     "php.validate.executablePath": "C:\\xampp\\php\\php.exe"
5 }
```

Por último, tendríamos que crear nuestra base de datos en PHPMyAdmin. Accedemos a él desde el navegador con localhost/phpmyadmin.



Aquí crearemos una nueva base de datos dónde ubicaremos los datos de nuestra aplicación.



Luego creamos 3 tablas que es de lo que consta nuestra aplicación.

Servidor: 127.0.0.1 » Base de datos: proyectopokemon » Tabla: especie										
Examinar	Estructura	SQL	Buscar	Insertar	Exportar	Importar	Privilegios	Operaciones	Seguimiento	
Estructura de tabla		Vista de relaciones								
#	Nombre	Tipo	Cotejamiento	Atributos	Nulo	Predeterminado	Comentarios	Extra	Acción	
<input type="checkbox"/>	1 id	int(10)			No	Ninguna		AUTO_INCREMENT		Eliminar Más
<input type="checkbox"/>	2 nombre	varchar(30)	utf8mb4_general_ci		No	Ninguna				Eliminar Más
<input type="checkbox"/>	3 tipo1	varchar(30)	utf8mb4_general_ci		No	Ninguna				Eliminar Más
<input type="checkbox"/>	4 tipo2	varchar(30)	utf8mb4_general_ci		No	Ninguna				Eliminar Más
<input type="checkbox"/>	5 region	varchar(30)	utf8mb4_general_ci		No	Ninguna				Eliminar Más
<input type="checkbox"/> Seleccionar todo Para los elementos que están marcados:          Examinar  Cambiar  Eliminar  Primaria  Único  Índice										

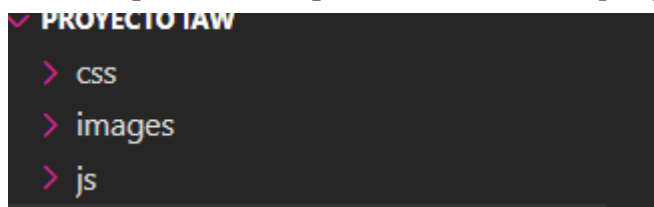
Servidor: 127.0.0.1 » Base de datos: proyectopokemon » Tabla: pokemon										
Examinar	Estructura	SQL	Buscar	Insertar	Exportar	Importar	Privilegios	Operaciones	Seguimiento	
Estructura de tabla		Vista de relaciones								
#	Nombre	Tipo	Cotejamiento	Atributos	Nulo	Predeterminado	Comentarios	Extra	Acción	
<input type="checkbox"/>	1 id	int(30)			No	Ninguna		AUTO_INCREMENT		Eliminar Más
<input type="checkbox"/>	2 especie	varchar(30)	utf8mb4_general_ci		No	Ninguna				Eliminar Más
<input type="checkbox"/>	3 nombre	varchar(30)	utf8mb4_general_ci		No	Ninguna				Eliminar Más
<input type="checkbox"/>	4 ataque	int(2)			No	Ninguna				Eliminar Más
<input type="checkbox"/>	5 defensa	int(2)			No	Ninguna				Eliminar Más
<input type="checkbox"/>	6 velocidad	int(2)			No	Ninguna				Eliminar Más
<input type="checkbox"/>	7 Nivel	int(3)			No	Ninguna				Eliminar Más
<input type="checkbox"/>	8 combates_ganados	int(5)			No	Ninguna				Eliminar Más
<input type="checkbox"/>	9 combates_perdidos	int(5)			No	Ninguna				Eliminar Más

Servidor: 127.0.0.1 » Base de datos: proyectopokemon » Tabla: combates										
Examinar	Estructura	SQL	Buscar	Insertar	Exportar	Importar	Privilegios	Operaciones	Seguimiento	Disparadores
Estructura de tabla		Vista de relaciones								
#	Nombre	Tipo	Cotejamiento	Atributos	Nulo	Predeterminado	Comentarios	Extra	Acción	
<input type="checkbox"/>	1 id_combate	int(5)			No	Ninguna		AUTO_INCREMENT		Eliminar Más
<input type="checkbox"/>	2 idpk1	int(5)			No	Ninguna				Eliminar Más
<input type="checkbox"/>	3 idpk2	int(5)			No	Ninguna				Eliminar Más
<input type="checkbox"/>	4 pokemon1	varchar(30)	utf8mb4_general_ci		No	Ninguna				Eliminar Más
<input type="checkbox"/>	5 pokemon2	varchar(30)	utf8mb4_general_ci		No	Ninguna				Eliminar Más
<input type="checkbox"/>	6 fecha	date			No	Ninguna				Eliminar Más
<input type="checkbox"/>	7 ganador	varchar(30)	utf8mb4_general_ci		No	Ninguna				Eliminar Más
<input type="checkbox"/> Seleccionar todo Para los elementos que están marcados:          Examinar  Cambiar  Eliminar  Primaria  Único  Índice  Espacial  Texto completo  Agregar a columnas centrales Eliminar de las columnas centrales										

Metemos algunos datos en estas tablas para poder utilizarlos en nuestra aplicación.

Ahora, en nuestro proyecto de VSC incluiremos las carpetas que contienen los elementos de bootstrap y los elementos de javascript útiles y con eso ya estaría todo preparado.



### 3. Descripción de la aplicación

Nuestra aplicación web consistirá en establecer un torneo Pokémon en el que se podrán registrar cualquier Pokémon existente de

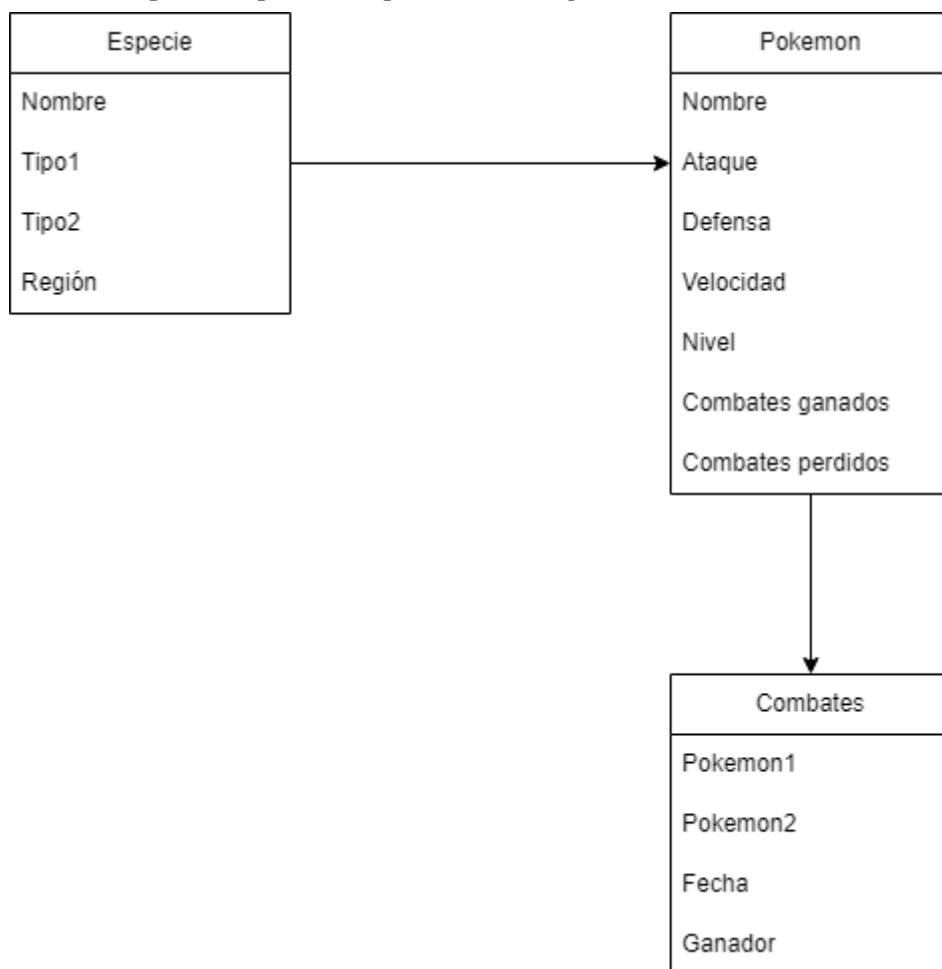
distintos entrenadores, con sus estadísticas individuales. Podemos introducir más de un Pokémon de la misma especie. De cada especie sabremos el nombre, los tipos o tipos y de qué región procede.

Podremos consultar el listado de Pokémon que tenemos registrado y en cada Pokémon ver cuántos de ese Pokémon están participando en el torneo. De cada Pokémon que tengamos registrado podremos ver su nombre individual (puesto por el entrenador), su ataque, defensa y velocidad, su nivel y cuántos combates ha ganado o perdido.

A su vez veremos una lista de los combates con su fecha y ganador correspondiente, así como generar nuevos combates que se añadirán a la lista. En el registro de combates añadiremos dos combates existentes ya en el torneo y la fecha en la que combaten, obteniendo así el resultado del combate.

## 4. Diseño de la aplicación

El esquema principal de nuestra aplicación será parecido al de esta imagen. De cada especie puede haber varios Pokémon y cada Pokémon puede participar en ningún combate o en varios.



El mockup de nuestra aplicación será algo parecido a esto.

## Listado de Pokemon

Pokemon1	Región	Tipo1	Tipo2	<a href="#">Ver Individuos</a>
Pokemon2	Región	Tipo1	Tipo2	<a href="#">Ver Individuos</a>

Registrar Pokemon

Lleva al formulario de registro de un nuevo tipo de pokemon

Ver Combates

Lleva al listado de combates

### Pokemon2

Nombre	Ataque	Defensa	Velocidad	Nivel	Combates ganados	Combates Perdidos	Eliminar
Nombre1	1	2	7	17	2	5	Eliminar
Nombre2	4	5	1	7	3	4	Eliminar

Volver al listado general

Añadir Pokemon

Lleva a un formulario que nos dejará poner las estadísticas de un Pokemon de esta especie e incluirlo en la lista

Ir a Combates

### Combates

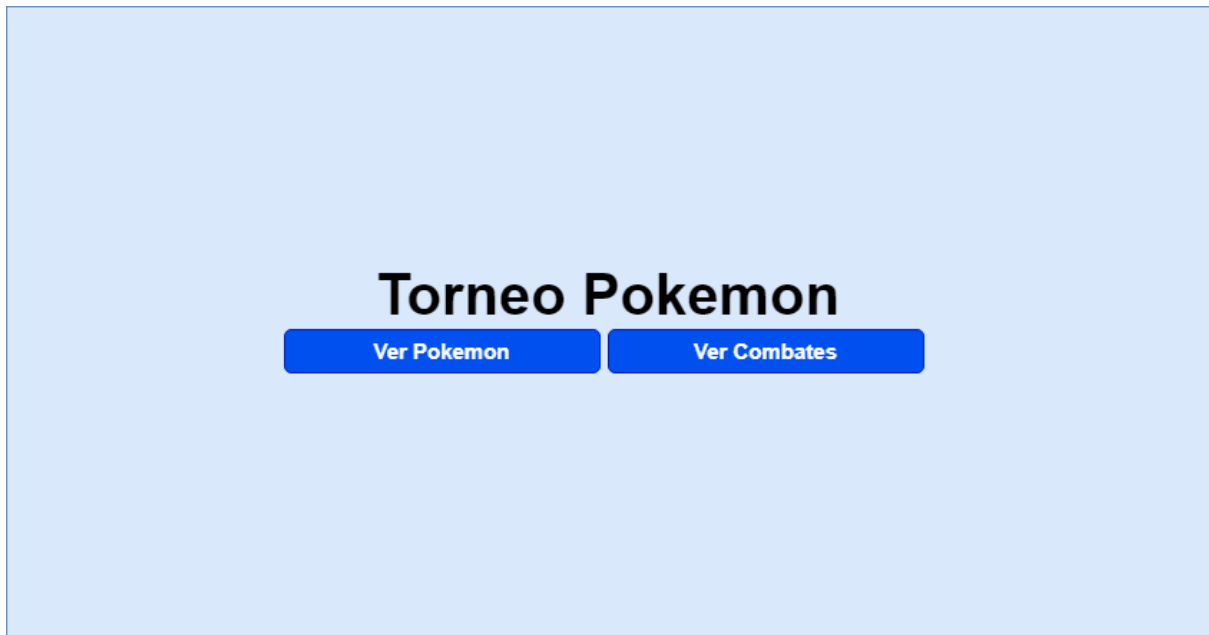
Pokemon1	Pokemon2	Fecha	Ganador
Dunsparce	Shuckle	19/02/23	Shuckle

Registrar Combate

Lleva a un formulario con los pokemon de la base de datos para seleccionar y enfrentar

Y tendríamos una portada que nos dejaría ir directamente a los Pokémon o los combates realizados que tenemos.





## 5. Desarrollo de la aplicación

### conexion.php

Establecerá la conexión con la base de datos.

```
<?php

//Nueva conexión de la clase mysqli
//Parámetros: ubicación del servidor, nombre de usuario,
contraseña, base de datos
$mysqli = new mysqli("localhost", "root", "", "proyectopokemon");
if($mysqli->connect_errno){
    echo "Fallo al conectar a MySQL: (" . $mysqli->connect_errno,
")", $mysqli->connect_error;
}
?>
```

### index.html

Esta pantalla será la primera que se nos abrirá al iniciar la aplicación web. Desde ella podremos ver un fondo simple, el título de nuestra aplicación centralizado y las dos opciones posibles debajo: ver Pokémon y ver combates.

```
<!DOCTYPE html>
<html lang="en">
```

```

<head>
  <meta charset="UTF-8">
  <meta http-equiv="X-UA-Compatible" content="IE=edge">
    <meta name="viewport" content="width=device-width,
initial-scale=1.0">
  <link rel="stylesheet" href="css/portada.css">
  <title>Portada</title>
</head>
<body>
  <header>
    <h1>Torneo Pokémon</h1>
  </header>

  <nav>
    <a href="pokemon.php">Ver Pokémon</a>
    <a href="combate.php">Ver Combates</a>
  </nav>

</body>
</html>

```

Visualización



## pokemon.php

En `pokemon.php`, podremos ver un listado de los Pokémon que hay registrados, uno por especie. Dentro de estos, podremos ver cada

individuo que pertenece a esa especie con un link Ver Individuos que nos llevará a estadísticas.php.

También tendremos un botón Registrar Pokémon que nos permitirá registrar una nueva especie de Pokémon. Si ya existe no se hará la operación.

```
<?php
//Conectamos con la base de datos
require 'conexion.php';
//Realizamos la consulta de nuestra tabla para poder obtener los datos
que hay guardados
$sql = "SELECT * from especie;";
//Ejecutamos la consulta
$resultado = $mysqli->query($sql);

?>
<!DOCTYPE html>
<html lang="en">
<head>
    <!-- Required meta tags -->
    <meta charset="utf-8">
    <meta name="viewport" content="width=device-width,
initial-scale=1, shrink-to-fit=no">

    <!-- Bootstrap CSS -->
    <link rel="stylesheet" href="css/bootstrap.min.css">
    <link rel="stylesheet" href="css/jquery.dataTables.min.css">

    <!-- Optional JavaScript -->
    <!-- jQuery first, then Popper.js, then Bootstrap JS -->
    <script src="js/jquery-3.4.1.min.js" ></script>
    <script src="js/bootstrap.min.js" ></script>
    <script src="js/jquery.dataTables.min.js" ></script>

    <title>Listado Pokemon</title>

    <script>
        // DataTables
        $(document).ready( function () {
            $('#tabla').DataTable();
        } );
    </script>
```

```

</head>
<body>
<div class="container">
    <div class="row">
        <h1>Listado Pokemon</h1>
    </div>
    <br>

    <table id="tabla" class="display" style="width:100%">
        <thead>
            <tr>
                <th>Nombre</th>
                <th>Tipo 1</th>
                <th>Tipo 2</th>
                <th>Región</th>
                <th></th>
            </tr>
        </thead>
        <tbody>
            <?php
                while($fila = $resultado->fetch_assoc()){
                    echo "<tr>";
                    echo "<td>",$fila['nombre'], "</td>";
                    echo "<td>",$fila['tipo1'], "</td>";
                    echo "<td>",$fila['tipo2'], "</td>";
                    echo "<td>",$fila['region'], "</td>";
                    echo "<td><a href="
'estadisticas.php?especie=$fila[nombre]' class='btn btn-warning'>Ver
individuos</a></td>";
                    echo "</tr>";
                }
            ?>
        </tbody>
    </table>

    <div class="row" style="display:flex; justify-content:
space-between";>
        <!-- Registrar -->
        <p><a href="registrarPokemon.php" class="btn
btn-primary">Registrar Pokémon</a></p>
        <p><a href="combate.php" class="btn btn-primary">Ver
Combates</a></p>
    </div>

```

```

        <br>
    </div>
</div>
</body>
</html>

```

En esta segunda parte del código tengo establecida la tabla con todos sus campos y ya es visualizable en Web.

### Listado Pokemon

Show 10 entries Buscar:

Nombre	Tipo 1	Tipo 2	Región	
Bulbasaur	Planta	Veneno	Kanto	<a href="#">Ver individuos</a>

Mostrando 1 a 1 de 1 registros Previous 1 Next

[Registrar Pokémon](#) [Ver Combates](#)

## registrarPokemon.php

Esta página consistirá de un formulario donde introduciremos los datos de los Pokémon únicos que queramos añadir a la base de datos de la aplicación.

### Código

```

<!doctype html>
<html lang="es">
    <head>
        <!-- Required meta tags -->
        <meta charset="utf-8">
        <meta name="viewport" content="width=device-width,
initial-scale=1, shrink-to-fit=no">

        <!-- Bootstrap CSS -->
        <link rel="stylesheet" href="css/bootstrap.min.css">

```

```

<title>Registro de Pokemon</title>
</head>
<body>
  <div class="container">
    <div class="row">
      <h1>Registro de Pokemon</h1>
    </div>

    <div class="row">
      <div class="col-md-8">
        <!-- Completar atributos de form -->
        <form id="registro" name="registro" method="post"
action="registrar2.php" autocomplete="off">
          <div class="form-group">
            <!-- Nombre -->
            <label for="id">Número Pokédex: </label>
            <input type="text" name="id"
class="form-control" id="formGroupExampleInput" placeholder="Introduce
el número de la pokédex">
          </div>

          <div class="form-group">
            <!-- Nombre -->
            <label for="nombre">Nombre: </label>
            <input type="text" name="nombre"
class="form-control" id="formGroupExampleInput" placeholder="Introduce
el nombre">
          </div>

          <div class="form-group">
            <!-- Teléfono -->
            <label for="tipo1">Tipo 1: </label>
            <input type="text" name="tipo1"
class="form-control" id="formGroupExampleInput" placeholder="Introduce
el primer tipo">
          </div>

          <div class="form-group">
            <!-- Fecha nacimiento -->
            <label for="tipo2">Tipo 2: </label>
            <input type="text" name="tipo2"
class="form-control" placeholder="Introduce el segundo tipo">
          </div>

```

```

        <div class="form-group">
            <!-- Fecha nacimiento -->
            <label for="region">Región: </label>
            <input type="text" name="region"
class="form-control" placeholder="Introduce la región">
        </div>

        <div class="form-group">
            <!-- Registrar -->
            <input type="submit" value="Registrar
Pokémon" name="registrar" class="btn btn-primary">
        </div>
    </form>
</div>
</div>
</div>
</body>
</html>

```

Contará con una segunda parte que se encargará de realizar todas las operaciones para añadir los datos a la base de datos.

```

!doctype html>
<html lang="es">
    <head>
        <!-- Required meta tags -->
        <meta charset="utf-8">
        <meta name="viewport" content="width=device-width,
initial-scale=1, shrink-to-fit=no">

        <!-- Bootstrap CSS -->
        <link rel="stylesheet" href="css/bootstrap.min.css">

        <title>Registro Pokémon</title>
    </head>
    <body>
        <?php
        //Conectamos a la base de datos
        require 'conexion.php';

        //Guardamos las variables de nuestro formulario
        $nombre = $_POST['nombre'];

```

```

        $tipo1 = $_POST['tipo1'];
        $tipo2 = $_POST['tipo2'];
        $region = $_POST['region'];

        //Elaboramos la sentencia SQL para insertar los datos en la
base de datos
        $sql= "INSERT INTO especie(nombre, tipo1, tipo2, region)
values('$nombre', '$tipo1', '$tipo2', '$region')";
        //Llevamos a cabo la consulta
        $res = $mysqli->query($sql);
        //Analizamos el resultado
        if($res>0){
            echo "<p class='alert alert-primary'>Registro
correcto</p>";
            echo "<p><a href='pokemon.php' class='btn
btn-primary'>Regresar</a></p>";
        }else{
            echo "<p class=' alert alert-primary'>Error al
insertar</p>";
            echo "<p><a href='pokemon.php' class='btn
btn-primary'>Regresar</a></p>";
        }

        ?>

</body>

```

Visualización



---

## Registro de Pokemon

Número Pokédex:

Nombre:

Tipo 1:

Tipo 2:

Región:

## estadisticas.php

En este archivo podremos ver cada Pokémon de la misma especie que pertenece a la base de datos junto a sus características individuales. Se accede a él a través de la opción ver individuos de pokemon.php

Como podemos ver esta es la tabla de Bulbasaur, con sus estadísticas y nombres personalizados de cada uno de los Bulbasaur. Si hacemos clic sobre eliminar, se nos eliminará ese individuo en concreto.

La opción de Volver al listado general nos regresará a pokemon.php-

La opción de Añadir Pokemon nos permitirá añadir un nuevo pokemon individual con sus estadísticas personales. Por defecto entrarán con 0 victorias y 0 derrotas.

La opción de Ir a combates nos llevará directamente a combate.php y ver todos los combates que se han producido.

## Visualización

## Bulbasaur

Show 10 entries

Buscar:

Nombre	Ataque	Defensa	Velocidad	Nivel	Combates ganados	Combates perdidos	
dsadas	1	2	3	12	0	0	<a href="#">Eliminar</a>
Pichuli	5	5	5	50	0	0	<a href="#">Eliminar</a>
Rayito Mcqueen	1	1	10	100	0	0	<a href="#">Eliminar</a>

Mostrando 1 a 3 de 3 registros

[Volver al listado general](#) [Añadir Pokémon](#)

Previous [1](#) Next [Ir a Combates](#)

## anadirPokemon.php

Esta página nos llevará a un archivo php en el que tendremos un formulario para introducir los datos de nuestro Pokémon individual.

En el formulario habrá un campo de tipo hidden que nos permitirá traernos la información de la especie en la que estemos y cuando añadamos al Pokémon, volver a llevarla a estadisticas.php de manera correcta.

```
<?php
$especie = $_GET['especie'];
?>
<!doctype html>
<html lang="es">
  <head>
    <!-- Required meta tags -->
    <meta charset="utf-8">
    <meta name="viewport" content="width=device-width,
initial-scale=1, shrink-to-fit=no">

    <!-- Bootstrap CSS -->
    <link rel="stylesheet" href="css/bootstrap.min.css">

    <title>Añadir Pokemon</title>
  </head>
  <body>
```

```

<div class="container">
  <div class="row">
    <h1>Añadir Pokemon entrenado</h1>
  </div>

  <div class="row">
    <div class="col-md-8">
      <!-- Completar atributos de form -->
      <form id="registro" name="registro" method="post"
action="anadirPokemon2.php" autocomplete="off">

        <div class="form-group">
          <!-- Nombre -->
          <label for="nombre">Nombre: </label>
          <input type="text" name="nombre"
class="form-control" id="formGroupExampleInput" placeholder="Introduce
el nombre">

        </div>

        <div class="form-group">
          <!-- tipo1 -->
          <label for="ataque">Ataque: </label>
          <input type="number" name="ataque"
class="form-control" id="formGroupExampleInput" placeholder="1 al 10"
max="10" min="1">

        </div>

        <div class="form-group">
          <!-- tipo2 -->
          <label for="defensa">Defensa: </label>
          <input type="number" name="defensa"
class="form-control" id="formGroupExampleInput" placeholder="1 al 10"
max="10" min="1">

        </div>

        <div class="form-group">
          <!-- region -->
          <label for="region">Velocidad: </label>
          <input type="number" name="velocidad"
class="form-control" id="formGroupExampleInput" placeholder="1 al 10"
max="10" min="1">

        </div>
      </form>
    </div>
  </div>
</div>

```

```

        <div class="form-group">
            <!-- region -->
            <label for="nivel">Nivel: </label>
                <input type="number" name="nivel"
class="form-control" id="formGroupExampleInput" placeholder="1 al 100"
max="100" min="1">
        </div>

        <input type="hidden" name="especie" value=<?php
echo $especie ?>>

        <div class="form-group">
            <!-- Registrar -->
            <input type="submit" value="Añadir Pokemon"
name="registrar" class="btn btn-primary">
        </div>
    </form>
</div>
</div>
</div>
</body>
</html>

```

Tendrá una segunda parte que nos realizará las operaciones para incluir nuestros datos dentro de la base de datos.

```

<!doctype html>
<html lang="es">
    <head>
        <!-- Required meta tags -->
        <meta charset="utf-8">
            <meta name="viewport" content="width=device-width,
initial-scale=1, shrink-to-fit=no">

        <!-- Bootstrap CSS -->
        <link rel="stylesheet" href="css/bootstrap.min.css">

        <title>Añadir Pokemon</title>
    </head>
    <body>
        <?php
            //Conectamos a la base de datos
            require 'conexion.php';

```

```

        //Guardamos las variables de nuestro formulario
        $nombre = $_POST['nombre'];
        $ataque = $_POST['ataque'];
        $defensa = $_POST['defensa'];
        $velocidad = $_POST['velocidad'];
        $nivel = $_POST['nivel'];
        $especie = $_POST['especie'];

        //Elaboramos la sentencia SQL para insertar los datos en la
base de datos
        $sql= "INSERT INTO pokemon(nombre, especie, ataque, defensa,
velocidad, Nivel) values('$nombre', '$especie', '$ataque', '$defensa',
'$velocidad', '$nivel')";
        //Llevamos a cabo la consulta
        $res = $mysqli->query($sql);
        //Analizamos el resultado
        if($res>0){
            echo "<p class='alert alert-primary'>Registro
correcto</p>";
            echo "<p><a href='estadisticas.php?especie=$especie'
class='btn btn-primary'>Regresar</a></p>";
        }else{
            echo "<p class=' alert alert-primary'>Error al
insertar</p>";
            echo "<p><a href='estadisticas.php?especie=$especie'
class='btn btn-primary'>Regresar</a></p>";
        }

        ?>
    </body>
</html>

```

Visualización

# Añadir Pokemon entrenado

Nombre:

Ataque:

Defensa:

Velocidad:

Nivel:

Añadir Pokemon

Saldrá un mensaje correcto si el Pokémon se ha introducido adecuadamente.

## eliminar.php

Esta página nos eliminará el individuo seleccionado de una especie, eliminando de la base de datos todas sus estadísticas.

```
<!doctype html>
<html lang="es">
  <head>
    <!-- Required meta tags -->
    <meta charset="utf-8">
    <meta name="viewport" content="width=device-width,
initial-scale=1, shrink-to-fit=no">

    <!-- Bootstrap CSS -->
    <link rel="stylesheet" href="css/bootstrap.min.css">
```

```

        <title>Eliminar Pokemon</title>
    </head>
    <body>
        <?php
            require 'conexion.php';

            $id= $_GET['id'];
            $especie = $_GET['especie'];

            //realizamos la sentencia para traernos los datos

            $sql="DELETE from pokemon where id = $id";

            //Ejecutamos la consulta
            $resultado=$mysqli->query($sql);
            //Mostramos el mensaje segun el resultado de la ejecucion
            if($resultado>0){
                echo "<p class='alert alert-primary'>Registro
eliminar</p>";
                echo "<p><a href='estadisticas.php?especie=$especie'
class='btn btn-primary'>Regresar</a></p>";

            }else{
                echo "<p class='alert alert-primary'>Error al
eliminar</p>";
                echo "<p><a href='estadisticas.php?especie=$especie'
class='btn btn-primary'>Regresar</a></p>";
            }
        ?>

    </body>
</html>

```

## combate.php

En combate.php podremos ver una tabla que muestra los combates que se han producido entre todos los Pokémon de la base de datos. Podemos ver los dos Pokémon que se han enfrentado, la fecha en la que lo hicieron y el ganador.

Tendremos un botón que nos llevará a un formulario crearCombate.php que nos permitirá crear nuevos enfrentamientos. Podremos volver a listado general de Pokémon con otro botón.

```

<?php
//Conectamos con la base de datos
require 'conexion.php';
//Realizamos la consulta de nuestra tabla para poder obtener los datos
que hay guardados
$sql = "SELECT * from combates;";
//Ejecutamos la consulta
$resultado = $mysqli->query($sql);

?>
<!DOCTYPE html>
<html lang="en">
<head>
    <!-- Required meta tags -->
    <meta charset="utf-8">
    <meta name="viewport" content="width=device-width,
initial-scale=1, shrink-to-fit=no">

    <!-- Bootstrap CSS -->
    <link rel="stylesheet" href="css/bootstrap.min.css">
    <link rel="stylesheet" href="css/jquery.dataTables.min.css">

    <!-- Optional JavaScript -->
    <!-- jQuery first, then Popper.js, then Bootstrap JS -->
    <script src="js/jquery-3.4.1.min.js" ></script>
    <script src="js/bootstrap.min.js" ></script>
    <script src="js/jquery.dataTables.min.js" ></script>

    <title>Combates</title>

    <script>
        // DataTables
        $(document).ready( function () {
            $('#tabla').DataTable();
        } );
    </script>

</head>
<body>
<div class="container">
    <div class="row">
        <h1>Combates</h1>

```



```

        </div>
        <br>

        <table id="tabla" class="display" style="width:100%">
            <thead>
                <tr>
                    <th>Contrincante 1</th>
                    <th>Contrincante 2</th>
                    <th>Fecha</th>
                    <th>Ganador</th>
                </tr>
            </thead>
            <tbody>
                <?php
                    while($fila = $resultado->fetch_assoc()){
                        echo "<tr>";
                                echo "<td style='background-color:
lightblue;'>",$fila['pokemon1'], "</td>";
                                echo "<td style='background-color:
lightyellow;'>",$fila['pokemon2'], "</td>";
                                echo "<td>",$fila['fecha'], "</td>";
                                echo "<td style='background-color:
lightgreen;'>",$fila['ganador'], "</td>";
                                echo "</tr>";
                    }
                <?>
            </tbody>
        </table>

        <div class="row" style="display:flex; justify-content:
space-between">
            <div class="row">
                <p><a href="crearCombate.php" class="btn
btn-primary">Crear combate</a></p>
            </div>
            <!-- Registrar -->
            <div class="row">
                <p><a href="pokemon.php" class="btn btn-danger">Volver
al listado general</a></p>
            </div>
        </div>
        <br>

```

```

        </div>
    </div>
</body>
</html>

```

## Visualización

### Combates

Show  entries

Buscar:

Contrincante 1	Contrincante 2	Fecha	Ganador
DugChamp	Pablo	2010-10-10	DugChamp
DugChamp	Tobi	1025-04-07	DugChamp
DugGod	DugChamp	1478-12-15	DugGod

Mostrando 1 a 3 de 3 registros

Previous  Next

[Crear combate](#) [Volver al listado general](#)

## crearCombate.php

Será un formulario en el que podremos seleccionar entre los Pokémon que se encuentran en la base de datos y la fecha del combate que tendrán. Esto nos redirigirá a crearCombate2.php dónde se calculará el ganador siguiendo un algoritmo propio: la suma de ataque, defensa y velocidad sumado al nivel del pokémon entre 4.

```

<?php
//Conectamos con la base de datos
require 'conexion.php';
//Creamos la consulta que nos devuelve todo los pokemon individuales
$sql = "select * from pokemon";
//Ejecutamos la consulta
$resultado = $mysqli->query($sql);
//Ejecutamos una segunda consulta igual para poder usarla en el select
$resultado2 = $mysqli->query($sql);
?>

<!doctype html>
<html lang="es">

<head>
    <!-- Required meta tags -->
    <meta charset="utf-8">
    <meta name="viewport" content="width=device-width, initial-scale=1, shrink-to-fit=no">

    <!-- Bootstrap CSS -->
    <link rel="stylesheet" href="css/bootstrap.min.css">

```

```

        <title>Creación de un combate</title>
</head>

<body>
    <div class="container">
        <div class="row">
            <h1>Crear combate</h1>
        </div>

        <div class="row">
            <div class="col-md-8">
                <!-- Completar atributos de form -->
                <form id="registro" name="registro" method="post"
action="crearCombate2.php" autocomplete="off">

                    <div class="form-group">
                        <!-- Nombre -->
                        <label for="contrincante1">Contrincante 1:
</label>

                        <select name="contrincante1" id="contrincante1"
class="custom-select" required>
                            <?php
                                while ($fila = $resultado->fetch_assoc()) {
                                    echo "<option value=", $fila['id'],
">", $fila['nombre'], "</option>";
                                }
                            <?>
                        </select>
                    </div>

                    <div class="form-group">
                        <!-- Nombre -->
                        <label for="contrincante2">Contrincante 2:
</label>

                        <select name="contrincante2" id="contrincante2"
class="custom-select" required>
                            <?php
                                while ($fila2 = $resultado2->fetch_assoc())
{
                                    echo "<option value=", $fila2['id'],
">", $fila2['nombre'], "</option>";
                                }

```

```

        ?>
        </select>
    </div>

    <div class="form-group">
        <!-- tipo2 -->
        <label for="tipo2">Fecha: </label>
        <input type="date" name="fecha"
class="form-control" placeholder="Introduce la fecha">
    </div>

        <div class="row" style="display:flex;
justify-content: space-between;">
        <input type="submit" value="Crear combate"
name="combate" class="btn btn-primary">
        <a href="combate.php" class="btn
btn-warning">Regresar</a>
    </div>
</form>

</div>
</div>
</div>
</body>
</html>

```

crearCombate2.php

Se calculará el ganador del combate, se introducirá en la tabla y se actualizarán sus estadísticas de combates perdidos y ganados en la tabla pokémon.

```

<!doctype html>
<html lang="es">
    <head>
        <!-- Required meta tags -->

```

```

<meta charset="utf-8">
    <meta name="viewport" content="width=device-width,
initial-scale=1, shrink-to-fit=no">

<!-- Bootstrap CSS -->
<link rel="stylesheet" href="css/bootstrap.min.css">

<title>Crear Combate</title>
</head>
<body>
    <?php
        //Conectamos a la base de datos
        require 'conexion.php';
        //Guardamos las variables de nuestro formulario
        $idpk1=$_POST['contrincante1'];
        $idpk2=$_POST['contrincante2'];
        $fecha=$_POST['fecha'];
        //inicializamos esta variable
        $poder1=0;
        $poder2=0;
        $victorias1=0;
        $victorias2=0;
        $derrotas1=0;
        $derrotas2=0;

        //Añadimos la excepción de si los dos contrincantes fueran el
mismo
        if($idpk1==$idpk2){
            echo "<p class=' alert alert-danger'>Error al insertar. No
puede combatirse a si mismo</p>";
            echo " <p><a href='crearCombate.php' class='btn
btn-primary'>Regresar al combate</a></p>";
        } else {
            //Calculamos cuál sería el ganador
            //Creamos dos consultas una para cada adversario
            $datos1="select * from pokemon where id like '$idpk1'";
            $datos2="select * from pokemon where id like '$idpk2'";
            //Ejecutamos las consultas
            $cons1=$mysqli->query($datos1);
            $cons2=$mysqli->query($datos2);
            //Creamos un valor de la fuerza de cada contrincante.
            //Esta fuerza será la suma de Ataque, Defensa, Velocidad y
el nivel dividido por 4
            while($fila=$cons1->fetch_assoc()){

```

```

        $poder1= $fila['ataque'] + $fila['defensa'] +
$fila['velocidad'] + ($fila['Nivel']/4);
        $pokemon1= $fila['nombre'];
        $victorias1=$fila['combates_ganados'];
        $derrotas1=$fila['combates_perdidos'];
    }

    while($fila2=$cons2->fetch_assoc()){
        $poder2= $fila2['ataque'] + $fila2['defensa'] +
$fila2['velocidad'] + ($fila2['Nivel']/4);
        $pokemon2= $fila2['nombre'];
        $victorias2=$fila2['combates_ganados'];
        $derrotas2=$fila2['combates_perdidos'];
    }

    //Una vez obtenidas las victorias y derrotas de cada uno,
actualizamos sus estadísticas.
    //Una vez tenemos las fuerzas comparamos los resultados
    if($poder1==$poder2){
        $ganador="Empate";
    } else if ($poder1>$poder2){
        $ganador=$pokemon1;
        $victorias1=$victorias1+1;
        $derrotas2=$derrotas2+1;

    } else {
        $ganador=$pokemon2;
        $derrotas1=$derrotas1+1;
        $victorias2=$victorias2+1;
    }

    //Elaboramos la sentencia SQL para insertar los datos en la
base de datos
        $sql= "INSERT INTO combates(idpk1, idpk2, pokemon1,
pokemon2, fecha, ganador) values('$idpk1', '$idpk2', '$pokemon1',
'$pokemon2', '$fecha', '$ganador')";
        //Actualizamos los datos automáticamente mediante la
sentencia.
        $sqlActuPoke1 = "UPDATE pokemon set combates_ganados =
$victorias1, combates_perdidos = $derrotas1 where id like '$idpk1'";

```

```

        $sqlActuPoke2 = "UPDATE pokemon set combates_ganados =
$Victorias2, combates_perdidos = $derrotas2 where id like '$idpk2'";
        //Llevamos a cabo la consulta
        $res = $mysqli->query($sql);
        $resActu1= $mysqli->query($sqlActuPoke1);
        $resActu2=$mysqli->query($sqlActuPoke2);

        //Analizamos el resultado

        if($res>0){
            echo "<p class='alert alert-primary'>Resultado:
$ganador</p>";
            echo "<p><a href='combate.php' class='btn
btn-primary'>Regresar</a></p>";
        }else{
            echo "<p class=' alert alert-primary'>Error al
insertar</p>";
            echo "<p><a href='combate.php' class='btn
btn-primary'>Regresar</a></p>";
        }
    }

    ?>

</body>
</html>

```

Visualización

## Crear combate

Contrincante 1:

DugChamp

Contrincante 2:

Dug1

Fecha:

15/04/2023

Crear combate

Regresar

Resultado: DugChamp

Regresar

## 6. Conclusión

Tuve que reducir las expectativas del proyecto desde el principio porque lo que tenía pensado era mucho más complejo y necesitaría todo el tiempo para llevarlo a cabo.

Al final he optado por una opción reducida de lo que se me ocurrió, que fue este torneo Pokémon, la cual es ampliable si tenemos en cuenta el factor del tipo de Pokémon a la hora de los combates.

También podría ampliarse incluyendo los movimientos que puede tener cada Pokémon y hacer a partir de ahí una simulación de combate aleatoria.

Cómo sabía que a todo esto no me iba a dar tiempo, opté por una versión más reducida y simple que cumpliera todas las características pedidas igualmente.

El proyecto me ha parecido muy interesante y probablemente desarrolle algo más con lo aprendido porque es algo que me motiva a aprender. Me ha dejado claro que me gustaría especializarme por



la parte de aplicaciones web o profundizar en temas de programación ya que esto es lo más parecido que hemos dado.