Ejercicio PHP/Base de datos

Repositorio: https://github.com/mdloma/apiphp.git

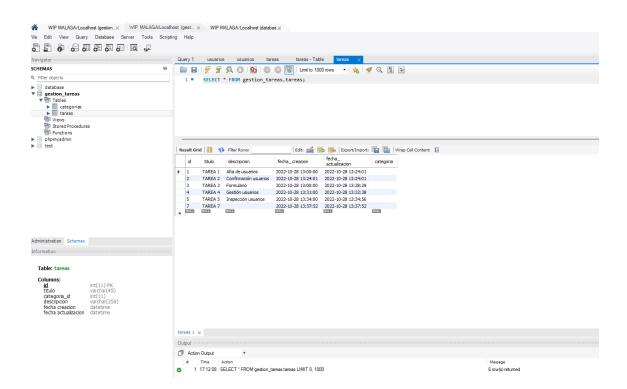
Descripción general

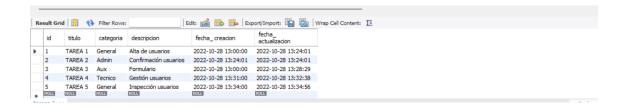
Crear un servicio REST para la gestión de tareas con PHP y MYSQL.

Requisitos mínimos

- ◆ Debe permitir realizar las operaciones CRUD sobre las tareas (listado, creación, obtención, edición y borrado)
- ◆ Las tareas deben tener al menos los campos: id, título, categoría, descripción, fecha creación y fecha actualización
- ◆ El título es el único campo obligatorio, el id será autogenerado, la categoría por defecto estará establecida a "General" y la descripción a una cadena vacía
- ◆ El servicio debe devolver al menos errores 404 al intentar obtener, editar o borrar una tarea que no exista y errores de validación como mínimo en la creación de una tarea si alguno de los datos no son correctos

MYSQL

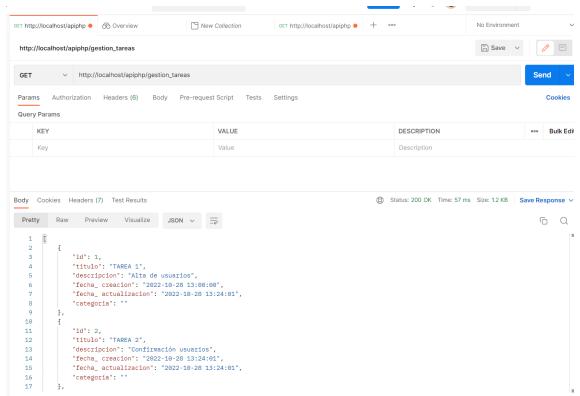




CRUD

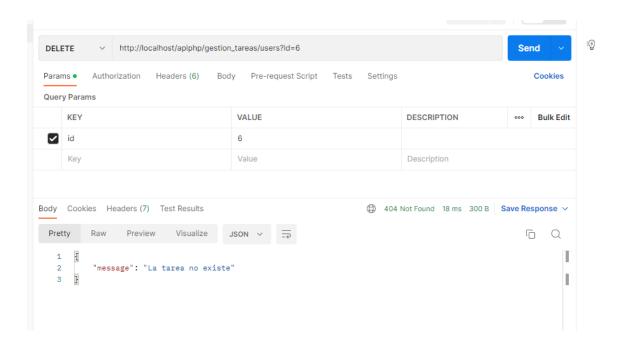
	#	Time	Action	Message	Duration / Fetch
0	1	18:22:28	SELECT * FROM gestion_tareas.tareas LIMIT 0, 1000	6 row(s) returned	0.000 sec / 0.000 sec
0	2	18:24:04	SELECT * FROM gestion_tareas.tareas LIMIT 0, 1000	6 row(s) returned	0.000 sec / 0.000 sec
•	3	19:12:54	DELETE FROM gestion_tareas.tareas WHERE id=7	1 row(s) affected	0.000 sec
8	4	19:16:40	UPDATE tareas SET titulo= "Prueba", descripcion= "Prueba", categoria= "Prueba" WHERE id=6	Error Code: 1146. Table 'database tareas' doesn't exist	0.000 sec
0	5	19:17:23	UPDATE gestion_tareas.tareas SET titulo="Prueba", descripcion="Prueba", categoria="Prue	0 row(s) affected Rows matched: 0 Changed: 0 Warnings: 0	0.016 sec

POSTMAN



Chokies
 A Canture requests
 Se Rootcamn
 Runner
 III Trach

```
"categoria": ""
  16
17
18
19
20
21
22
23
24
25
26
27
28
                             "id": 3,
"ittulo": "TAREA 3",
"descripcion": "Formulario",
"fecha_ creacion": "2022-10-28 13:00:00",
"fecha_ actualizacion": "2022-10-28 13:28:29",
"categoria": ""
                              "id": 4,
"titulo": "TAREA 4",
                              "titulo": "TAREA A",
"descripcion": "Gestión usuarios",
"fecha_ creacion": "2022-10-28 13:31:00",
"fecha_ actualizacion": "2022-10-28 13:32:38",
"categoría": ""
   29
30
31
32
                                                                                                                                                                                                       6 Cookies № Capture requests 😝 Bootcamp 🕞 Runner 🎹 Trash 🖟
Pretty Raw Preview Visualize JSON ∨ =
                                                                                                                                                                                                                                                                                                             © Q
                                "id": 5,
"titulo": "TAREA 5",
"descripcion": "Inspección usuarios",
"fecha_creacion": "2022-10-28 13:34:00",
"fecha_actualizacion": "2022-10-28 13:34:56",
"categoria": ""
     36
     38
39
40
     42
                                "id": 7,
"titulo": "TAREA 7",
     44
                                 "descripcion": "",
"fecha_ creacion": "2022-10-28 13:37:52",
"fecha_ actualizacion": "2022-10-28 13:37:52",
     46
                                 "categoria": "'
     48
    49
50
```



PHP

index.php

functions.php

```
<?php
function getRequestBody(){
 $json = file_get_contents('php://input');
 return json_decode($json, true);
function insertTarea($conn, $tareaData){
 $insertTarea = [
    ':titulo' => $tareaData['titulo'],
    ':descripcion' => $tareaData['descripcion'],
    ':categoria' => $tareaData['categoria'],
    ':fecha_creacion' => (new DateTime())->format('Y-m-d H:i:s'),
    ':fecha_actualizacion' => (new DateTime())->format('Y-m-d H:i:s')
  $insertSQL = "INSERT INTO tareas ( titulo, categoria ,descripcion,
  fecha_creacion, fecha_ actualizacion) VALUES (:titulo,:categoria,
  :descripcion, :fecha_creacion,:fecha_actualizacion)";
  $query = $conn->prepare($insertSQL);
  try{
    if($query->execute($insertTarea)) {
       return $conn->lastInsertId();
  }catch(Exception $e){
   return $e->getMessage();
function getTarea($conn, $id){
 $tareaSQL = "SELECT * FROM tareas WHERE id=:id";
 $query = $conn->prepare($tareaSQL);
  $query->setFetchMode(PDO::FETCH_ASSOC);
  $query->execute([':id' => $id]);
  $tareas = $query->fetchAll();
  if(count($tareas) === 0){
   return null;
  return $tareas[0];
```

```
$usersSQL = "SELECT * FROM tareas ORDER BY titulo ASC";
 $query = $conn->prepare($tareaSQL);
 $query->setFetchMode(PDO::FETCH_ASSOC);
 $query->execute();
 return $query->fetchAll();
function editTarea($conn, $tareaData, $id){
 $updateTarea = [
   ':titulo' => $tareaData['titulo'],
   ':categoria' => $tareaData['categoria'],
   ':descripcion' => $tareaData['descripcion'],
   ':id' => $id
 $updateSQL = "UPDATE tareas SET titulo=:titulo, descripcion=:descripcion,
 categoria=:categoria, WHERE id=:id";
 $query = $conn->prepare($updateSQL);
 try{
   if($query->execute($updateTarea)) {
       return $id;
 }catch(Exception $e){
   return null;
function deleteTarea($conn, $id){
 $deleteTarea = [
   ':id' => $id
 $deleteSQL = "DELETE FROM gestion_tareas.tareas WHERE id=:id";
 $query = $conn->prepare($deleteSQL);
 try{
   if($query->execute($deleteTarea)) {
     return $query->rowCount();
```

```
if($query->execute($deleteTarea)) {
    return $query->rowCount();
    }
}catch(Exception $e){
    return 0;
}
```

Exercices-db.sql

```
exercise-db.sql
     DROP TABLE IF EXISTS `tareas`;
     CREATE TABLE 'usuarios' (
       `id` int NOT NULL AUTO_INCREMENT,
       `titulo` varchar(45) NOT NULL,
        `categoria` varchar(45) NOT NULL,
        `descripcion` varchar(250) NOT NULL DEFAULT '',
        `fecha_creacion` datetime NOT NULL DEFAULT CURRENT_TIMESTAMP,
      PRIMARY KEY ('id'),
      ) ENGINE=InnoDB AUTO INCREMENT=20 DEFAULT CHARSET=utf8mb4
      COLLATE=utf8mb4 0900 ai ci;
     LOCK TABLES `tareas` WRITE;
     INSERT INTO `tareas` VALUES (6, 'TAREA 1', 'General', 'Alta usuario', '2022-10-19
      11:45:26'),(8,'TAREA 8','General','2022-10-19 11:45:26')(9,'TAREA 9',
      'Confirmacion usuarios', 2022-10-19 12:03:30');
     UNLOCK TABLES;
```

db.php

Delete.controler.php

```
$uriParts = explode('/',substr($uri,1));

$affectedRows = deleteTarea($conn, $uriParts[1]);

if($affectedRows === 0){

   http_response_code(404);
   echo '{"message": "La tarea no existe"}';
   return;
}

http_response_code(204);
echo '';
return;
```

Get.controller.php

Post.controller.php

```
ıtrollers > 🦬
    <?php
        $tareaData = getRequestBody();
        if(!is_array($tareaData)){
          http_response_code(400);
          echo '{"message": "Petición mal formada"}';
          return;
        if(!key_exists('titulo',$tareaData)){
          http response code(400);
          echo '{"message": "Es necesario especificar el titulo de la tarea"}';
          return;
        if(!key_exists('categoria',$tareaData)){
          http_response_code(400);
          echo '{"message": "Es necesario que especifiques una categoria de la
          tarea"}';
          return;
        if(!key_exists('descripcion',$userData)){
          http_response_code(400);
          echo '{"message": "Es necesario que especifiques una descripcion de la
          tarea"}';
```

Put.controller.php

.htacess

```
.htaccess
1   RewriteEngine On
2   RewriteCond %{REQUEST_FILENAME} !-f
3   RewriteRule ^(.*) index.php
```