

Guia desarrollador Clase Velada

Joel Acoran Cruz Morales



DOC_Guia_Desarrollador_Velada_Joel_A_Cruz_Mo
rales

Documento para la guía de desarrollador de la clase velada.

Fecha de versión: 15/04/2024

Historial de revisiones

Fecha	Descripción	Autor
05/04/2024	Creación del documento y documentación de la clase	Joel A. Cruz Morales
10/04/2024	Creación de la implementación en la aplicación	Joel A. Cruz Morales
15/04/2024	Última versión actualizada y terminada del documento	Joel A. Cruz Morales

DOC_Guia_Desarrollador_Velada_Joel_A_Cruz_Morales	1
Historial de revisiones	2
1. Objetivo	4
2. Alcance	4
3. Dependencias	4
4. Integración en la aplicación	4
4.1. Añadir referencias al proyecto	4
4.1.1. Uso de instancia de clases	5
4.1.2. Ejemplos	6

1. Objetivo

El objetivo de este documento es conseguir explicar el funcionamiento de este controlador, para manejar el CRUD, relacionado con la tabla Velada en la Base de datos.

2. Alcance

Esta guía está dirigida a todos los desarrolladores que necesitan entender y utilizar el controlador siguiente en sus proyectos similares haciendo uso de php, JavaScript y la conexión a la base de datos.

3. Dependencias

Este controlador depende de su modelo Velada

4. Integración en la aplicación

Esta clase, se encarga de manejar las operaciones relacionadas con las veladas en mi base de datos. Está ubicada dentro del directorio 'app/Http/Controllers' del proyecto en Laravel. En esta clase contiene los métodos necesarios para realizar un CRUD completo, ya sea obteniendo veladas por su id, obtener todas, crear nuevas, eliminar e incluso editar una velada ya existente.

4.1. Añadir referencias al proyecto

Primero debemos localizar nuestro modelo velada, que adjunto lo siguiente

<https://github.com/Joelcrmo/ProyectoUFC/blob/main/app/Models/Velada.php>,

una vez con este modelo declarado en nuestro controlador, podremos hacer usos de él.

4.1.1. Uso de instancia de clases

El controlador proporciona varios métodos para interactuar con la base de datos y gestionar las veladas:

- **obtenerVeladas():** Retorna todas las veladas con su respectiva localización.
- **obtenerVeladaPorID(\$ID_Velada):** Retorna la información de una velada específica basada en su ID.
- **crearVelada(Request \$request):** Crea una nueva velada en la base de datos.
- **eliminarVelada(\$ID_Velada):** Elimina una velada de la base de datos basada en su ID.
- **actualizarVeladaEnBD(\$ID_Velada, \$nombreVel, \$idLocalizacion, \$fechaVel):** Actualiza la información de una velada existente en la base de datos.
- **edit(\$ID_Velada):** Retorna la información de una velada específica para su edición.

4.1.2. Ejemplos

```
1 <?php
2
3 namespace App\Http\Controllers;
4
5 use Illuminate\Http\Request;
6 use Illuminate\Support\Facades\DB;
7 use App\Models\Velada; // Importamos la clase Velada del modelo
8
9 class VeladaDBController extends Controller
10 {
11     /**
12      * Obtiene una velada por su ID.
13      *
14      * @param int $ID_Velada
15      * @return \Illuminate\Http\JsonResponse
16      */
17     public function obtenerVeladaPorID($ID_Velada)
18     {
19         // Consulta la base de datos para obtener la velada con el ID proporcionado
20         $velada = DB::select('SELECT * FROM Velada WHERE ID_Velada = ?', [$ID_Velada]);
21
22         // Retorna la respuesta JSON con la velada obtenida
23         return response()->json($velada);
24     }
25 }
26
```

En este ejemplo:

- Se define un método **obtenerVeladaPorID()** en la clase **VeladaDBController** para obtener una velada por su id
- Se incluye comentarios describiendo el funcionamiento y los parámetros del método
- Dentro del método se utiliza la query para ejecutar la consulta SQL que selecciona la velada con el ID proporcionado.
- Se retorna la respuesta JSON con la velada obtenida.