

Introducción

Este proyecto Laravel es una aplicación para listar los datos almacenados en la base de datos Warehouse que se encuentra instalada en el mysql local .

Proporciona funcionalidades para gestionar filtrar la búsqueda de las respuestas por ID y por fecha.

Requisitos

Requisitos funcionales

- Mostrar un panel de control con estadísticas de encuestas por año y por mes.
- Mostrar la fecha con el mayor número de encuestas.
- Mostrar el año con el mayor número de encuestas.
- Cargar y buscar respuestas de encuestas.
- Procesar y mostrar metadatos en formato JSON.

Requisitos no funcionales

- Framework Laravel 8.0 o superior.
- Servidor web compatible con PHP.
- Base de datos MySQL o compatible.

Arquitectura

El proyecto sigue el patrón Modelo-Vista-Controlador (MVC) de Laravel. A continuación se describen los componentes principales:

- Controladores:
 - `AdminWarehouseController`: Controlador principal que maneja las acciones relacionadas con la administración del almacén.
- Modelos:
 - `Respuesta`: Representa una respuesta de encuesta.
 - `Encuesta`: Representa una encuesta.
 - `Comment`: Representa un comentario asociado a una respuesta de encuesta.
 - `Data`: Representa los metadatos de una respuesta de encuesta.
 - `Response`: Representa una respuesta asociada a una pregunta de encuesta.
 - `Root`: Representa la raíz de los metadatos de una respuesta de encuesta.
- Vistas:
 - `dashboard.blade.php`: Vista del panel de control con estadísticas.
 - `index.blade.php`: Vista de carga y búsqueda de respuestas de encuestas.
 - `mostrar_metadatos.blade.php`: Vista para mostrar los metadatos en formato JSON.

Configuración del entorno

- Requisitos:
 1. PHP 7.4 o superior.
 2. Laravel Framework 8.0 o superior.
 3. MySQL o un servidor de base de datos compatible.
- Configuración:
 1. Clonar el proyecto desde el repositorio Git.
 2. Ejecutar `composer install` para instalar las dependencias de Laravel.
 3. Configurar el archivo `.env` con los detalles de la base de datos.
 4. Ejecutar `php artisan migrate` para migrar las tablas de la base de datos.
 5. Ejecutar `php artisan serve` para iniciar el servidor de desarrollo de Laravel.

Uso del sistema

Panel de control

- URL: [/](#)
- Descripción: Esta página muestra estadísticas de encuestas por año y por mes.

Cargar y buscar respuestas de encuestas

- URL: [/index](#)
- Descripción: En esta página se pueden cargar y buscar respuestas de encuestas. También se puede filtrar por fecha y encuesta.

Procesar y mostrar metadatos

- URL: [/mostrar_metadatos](#)
- Descripción: Esta página permite procesar y mostrar metadatos en formato JSON. Los metadatos se ingresan como una cadena JSON y se presentan en forma de tabla.

API

La aplicación no proporciona una API pública en este momento.

Consideraciones de seguridad

- Se recomienda implementar autenticación y autorización para proteger las funcionalidades sensibles.
- Validar los datos de entrada para prevenir ataques de inyección SQL y XSS.
- Mantener actualizadas las dependencias y seguir las mejores prácticas de seguridad de Laravel.

Gestión de errores y depuración

- Los errores y excepciones se manejan a través del mecanismo de manejo de errores incorporado de Laravel.
- Se pueden consultar los registros de errores en el archivo `storage/logs/laravel.log`.

Funciones Utilizadas

Clase: AdminWarehouseController

Función: mostrarDashboard(Request \$request)

- Descripción: Muestra el panel de control del sistema.
- Parámetros:
 - `$request`: Objeto de la clase `Request` que contiene los datos de la solicitud HTTP.
- Valor de retorno: Vista `dashboard` con las variables `$encuestasPorAnho` y `$encuestasPorMes`.

Función: mostrarFechaMayorEncuestas()

- Descripción: Obtiene la fecha en la que se registró la mayor cantidad de encuestas.
- Valor de retorno: La fecha en formato "d/m/Y" si se encuentra una fecha, de lo contrario, retorna `null`.

Función: mostrarEncuestasAnho()

- Descripción: Obtiene el año en el que se registró la mayor cantidad de encuestas.
- Valor de retorno: El año en formato "Y" si se encuentra un año, de lo contrario, retorna `null`.

Función: mostrarEncuestasTotal()

- Descripción: Obtiene el total de respuestas de las encuestas registradas.
- Valor de retorno: El número total de respuestas.

Función: cargarRespuestas(Request \$request)

- Descripción: Carga las respuestas de las encuestas de acuerdo a los filtros de búsqueda.
- Parámetros:
 - `$request`: Objeto de la clase `Request` que contiene los datos de la solicitud HTTP.
- Valor de retorno: Vista `index` con la variable `$respuestas` que contiene las respuestas de las encuestas paginadas.

Función: procesarDatosJson(Request \$request)

- Descripción: Procesa los datos JSON de las respuestas de las encuestas y genera una tabla HTML con los metadatos.
- Parámetros:
 - `$request`: Objeto de la clase `Request` que contiene los datos de la solicitud HTTP.
- Valor de retorno: Vista `mostrar_metadatos` con la variable `$tableContent` que contiene la tabla HTML con los metadatos.