MANEJO DE PERMISOS

La gestión de permisos de la página se respalda mediante paquete diseñado para Laravel:

* spatie/laravel-permission

Este permite gestionar roles y permisos de usuarios de forma eficiente desde la base de datos. Facilita la asignación de roles a los usuarios y la configuración de permisos específicos, tanto a nivel de rol como de manera individual. Gracias a esto, se simplifica la implementación de sistemas de autorización y control de acceso basados en roles, mejorando la seguridad y organización dentro de la aplicación.

Se implementa la configuración de dos roles:

* ADMIN
* USER

ADMIN:

Posee el acceso TOTAL de toda la página:

* Manejo de datos: cargar, editar y eliminar datos que se encuentran cargados.
* Ver datos: Permite visualizar la información, cargada en la base de datos, que se carga en los .blade.
* Login: Permite el registro o inicio de sesión de manera sencilla, brindando acceso a las secciones restringidas de la página destinadas a usuarios con rol ADMIN.

USER:

Posee el acceso a la VISTA de los datos en la página:

* Ver datos: Permite visualizar la información, cargada en la base de datos, que se muestra en los .blade.

Se compone siete permisos para poder ser asignados únicamente al ADMIN:

Tres permisos se encuentran relacionados a la sección de emprendimientos:

* Crear emprendimiento: Permite la realización de un nuevo emprendimiento para ser cargado en la página.
* Editar emprendimiento: Accede a un formulario con la información del emprendimiento ya cargada para poder ser actualizada en la Base de Datos.
* Eliminar emprendimiento: Deshace un emprendimiento para que no sea más visible en la página.

Un permiso se encuentra vinculado a la parte de Solicitantes:

* Descarga Excel: Permite al Admin la descarga de datos en un archivo Excel según ciertos filtros que se pueden establecer antes de obtener el archivo en formato Excel.

Los restantes permisos se asocian al sector de noticias:

* Crear noticia: Permite la carga de una nueva noticia para ser mostrado en la página.
* Editar noticia: Accede a un formulario con la información de la noticia ya cargada para poder ser actualizada en la Base de Datos y ser mostrada con los detalles actualizados.
* Eliminar noticia: Deshace una noticia para que no sea más visible en la página.

TABLA DE PERMISOS

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  | Permisos | Crear emprendimiento | Editar emprendimiento | Eliminar emprendimiento | Crear noticia | Editar noticia | Eliminar noticia | Filtrar datos |
| Roles |  |  |  |  |  |  |  |  |
| ADMIN |  |  |  |  |  |  |  |  |
| USER |  |  |  |  |  |  |  |  |

CONCLUSIÓN:

La implementación del paquete **Laravel-permission** ha aportado múltiples beneficios en distintos componentes del sistema, los cuales se detallaran a continuación.

En cuanto al Ruteo, nos permitió proteger rutas mediante Middleware, asegurando que solo los usuarios con los permisos correspondientes asignados a sus roles puedan acceder a determinadas funciones del controlador. Esto logró fortalecer la seguridad y la organización del acceso a los recursos.

En el manejo de sesión en el Frontend, su excelente manipulación de datos permite consultar, de forma sencilla, si ha iniciado sesión o si tiene un rol/permiso asignado, permitiendo que, en dos líneas, se pueda hacer este control de una forma completamente segura, simple y rápida.

En el **manejo de sesiones desde el frontend**, nos facilitó la verificación de autenticación y permisos de forma sencilla y eficiente. Con apenas unas pocas líneas de código, se puede comprobar si un usuario ha iniciado sesión y si posee un rol o permiso específico, lo que agiliza el desarrollo sin comprometer la seguridad.

En general, la integración de Laravel-permission fue **sencilla, liviana y ágil**, mejorando significativamente el control de acceso. Nos permitió gestionar roles y permisos de manera clara, estructurada y fácilmente escalable, optimizando tanto la implementación como el mantenimiento del sistema.