

Лабораторная работа №4

Многопользовательское приложение

Подготовка окружения

Для выполнения лабораторной работы достаточно настроек окружения из предыдущей лабораторной работы.

Выполнение задания

Используйте приложение из предыдущей лабораторной работы.

До этого с вашим приложением мог работать только один пользователь, добавлять и просматривать объекты данных. Используйте функционал Laravel и встроенную сущность User для реализации многопользовательского приложения.

Реализуйте новый функционал:

- При помощи [миграций](#) измените таблицу в базе данных sqlite (можно использовать mysql, postgresql и другие совместимые СУБД) для хранения данных об объектах вашего варианта. Создайте ключ 'user_id' для связи с таблицей users.
- Установите пакет [Laravel Breeze](#). Убедитесь, что появились маршруты `/login` и `/register` для аутентификации и регистрации пользователя.
- Измените роуты и контроллер для работы с вашими объектами так, чтобы у каждого пользователя выводился список объектов. Зарегистрируйте несколько пользователей, присвойте им ранее добавленные объекты. Убедитесь, что по ID каждого пользователя загружается связанный список объектов.
- При помощи миграций модифицируйте таблицу users, добавьте флаг is_admin со значением по умолчанию false. Установите одному из пользователей значение в true при помощи laravel tinker.
- Модифицируйте действия create, store, edit, update, destroy. Так, чтобы работало следующие правила: 1) создавать новые объекты может любой пользователь. 2) редактировать пользователь может только созданные им объекты. 3) пользователь может [мягко удалить](#) созданный им объект 4) пользователь с флагом "администратор" может восстановить мягко удаленный объект или удалить его без возврата. Реализуйте соответствующие пути в routes/web.php. Используйте проверку [текущего пользователя](#). Используйте [Events или Closures](#) для проверки прав пользователя на определенные действия на уровне модели и [Gates](#) для определения правил.
- Убедитесь, что все CRUD-операции для каждого объекта корректно работают.

Критерии оценки

Базовый уровень (4 балла):

- Корректно установлен Laravel Breeze, пользователь может зарегистрироваться и авторизоваться.
- После авторизации пользователь видит списки объектов всех пользователей, перебирая их ID в адресной строке.
- Добавлять, удалять и редактировать объекты может только авторизованный пользователь.
- При добавлении нового объекта в базу данных записывается ID пользователя, который это выполнил.

Расширенный уровень (6 баллов):

- Критерии базового уровня
- Реализован отдельный метод для вывода всех пользователей с возможностью навигации по ним средствами шаблона.
- Реализован пользователь с правами администратора, которому доступны действия по редактированию и удалению объектов у любых пользователей.
- Реализован механизм Soft Deletes, действия восстановления и полного удаления для пользователя-администратора, возможность просмотра полного списка объектов пользователя, включая удаленные им объекты.
- Проверка прав доступа пользователя реализована на уровне интерфейса (кнопки не появляются, если нет прав), на уровне контроллера или маршрута (при помощи Gates) и на уровне модели (через Events / Closures)
- Ссылка для вывода объектов пользователя использует не ID пользователя, а его username. Подробнее о том как это правильно сделать см. [скринкаст](#)

Справочные материалы:

- Видеоурок по использованию Laravel Breeze
<https://www.youtube.com/watch?v=DnN43kUPzy4>