Весь проект – админка для управления контентом. Функционал рассмотрен на примере страницы patronages

**app/Http/Controllers/Admin/PatronagesController.php**

index() – выводит список всех меценатов с постраничной навигацией

create() – выводит страницу с формой создания нового элемента

store() – создает новую запись. Для валидации используются классы с наследованием от FormRequest

edit() – выводит страницу с формой для редактирования записи

update() – обновляет существующую запись. Для валидации используются классы с наследованием от FormRequest

delete() – удаляет существующую запись

deleteDetailPicture() – удаляет только детальную картинку для записи

При вызове конструктора контроллера автоматической фабрикой создается сущность сервиса, в которой реализован функционал

**app/Services/PatronagesService.php**

create() – создает новую сущность модели и сохраняет данные. Если в запросе есть файлы, то для них формируется уникальное имя и происходит сохранение на сервере,а в БД записывается только путь к файлу.

update() – аналогично create(), но работает с переданной сущностью

delete() – удаляет запись

deleteDetailPicture() – удаляет только детальную картинку для записи

**app/Http/Controllers/Api/PatronagesController.php**

Для получения данных сайтом написано АПИ взаимодействия с админкой Все методы имеют phpDoc описание для генерации json документа Swagger через библиотеку darkaonline/l5-swagger

getPatronages() – возвращает массив всех меценатов

getPatronagesByPage() - возвращает массив конкретной страницы меценатов