* Задача 1: Используя фреймворк symfony / laravel создать сайт для вывода подборки квартир для клиента

Для решения задачи необходимо сделать следующие:

* Используя терминал создать базовый проект на выбранном фреймворке
* Создать базу данных со следующей структурой:

Сущность подборка (название таблицы - offers)

| Поле | Комментарий |
| --- | --- |
| id | Создается автоматически |
| b24\_contact\_id | id контакт из Битрикс24 (всегда число) |
| b24\_deal\_id | id сделки из Битрикс24 (всегда число) |
| b24\_manager\_id | id сотрудника из Битрикс24 (всегда число) |
| manager | ФИО Сотрудника |
| position | Должность сотрудника (Специалист отдела продаж) |
| phone | Телефон сотрудника |
| avatar | Ссылка на фото (Для задания ссылка на файл из текущего проекта) |
| status | Статус обработки (Новая, Отправлена клиенту, Просмотрена клиентом) |
| date\_end | Дата истечения действия подборки для клиента (только дата) |
| created\_at | Дата создания подборки |

Сущность элемент подборки (название таблицы offer\_items)

| Поле | Комментарий |
| --- | --- |
| id | Создается автоматически |
| offer\_id | Связь с подборкой (Один-ко-Многим) |
| cid | Уникальный id из 1С (тип текст) |
| type | Тип квартиры (Евродвушка) |
| square | Площадь |
| complex | Название ЖК (Да.Квартал Республика) |
| house | Название дома (ГП-2) |
| description | Описание квартиры |
| images | Массив ссылок на картинки (тип json array). Ссылки на картинки из текущего проекта |
| like | Отметка “Нравится” (тип boolean) |

* Заполнить базу тестовыми данными
* Разработать страницу согласно верстке с тестовыми данными из базы

## Задача 2: Для созданного сайта необходимо разработать rest api для наполнения базы данных подборками

Сервис принимает запросы от внешних источников в формате json, в таком же формате отдает ответы.

Для решения задачи необходимо сделать следующее:

* Создать контроллер где базовый путь будет “/api/v1/‘
* Для работы с каждой сущностью создать отдельные методы в контроллере.
* Реализовать выполнение следующих действий с сущностями:
  + Создание подборки (В формате json в сервис будет отправляться запрос со следующими данными (пример))
  + {
  + "b24\_contact\_id": 1001,
  + "b24\_deal\_id": 34532,
  + "b24\_manager\_id": 12,
  + "manager": "Иванов Иван",
  + "position": "Специалист отдела продаж",
  + "phone": "79526846266",
  + "avatar": ";ljksdflkasjdflk.jpg"
  + }
  + Обновление подборки. Для обновления в адресе обязательно будет указан id подборки и должны быть доступны следующие поля для обновления b24\_manager\_id, manager, position, avatar, status, date\_end
  + Создание элемента подборки (В формате json в сервис будет отправляться запрос со следующими данными (пример))

{

"offer\_id": 1001,

"cid": "936DA01F-9ABD-4d9d-80C7-02AF85C822A8",

"type": "Евродвушка",

"square": 54,3,

"complex": "Да.Квартал Республика",

"house": "ГП-2",

"description": "Lorem ipsum dolor sit amet, consectetur adipiscing elit, sed do eiusmod tempor incididunt ut labore et dolore magna aliqua. Ut enim ad minim veniam, quis nostrud exercitation ullamco laboris nisi ut aliquip ex ea commodo consequat. Duis aute irure dolor in reprehenderit in voluptate velit esse cillum dolore eu fugiat nulla pariatur. Excepteur sint occaecat cupidatat non proident, sunt in culpa qui officia deserunt mollit anim id est laborum.",

"images": ["",""]

}

* + Удаление элемента подборки, для удаления будет передаваться id элемента подборки

## Задача 3: Блокировать возможность добавлять новые элементы, если у подборки статус отличается от “Новая”

## Задача 4: Для исключения возможности перебора идентификаторов подборок необходимо реализовать сложный url

## Задача 5: Реализовать сбор статистики посещений и отметок “Нравится”

## Задача 6: Реализовать защиту для API (авторизация/проверка токена)

## Задача 7: Реализовать события для оповещения внешних систем об открытии подборки пользователем и новых отметках “Нравится”