

Для контент-менеджера:

- Создать заявку
- Редактировать заявку (доступно, только если она не была отправлена на согласование)
- Отправить заявку на согласование
- Просмотр согласованных заявок
- Просмотр отклоненных заявок
- Поиск и фильтрация заявок по создателю, дате создания, номеру

Составим таблицу целей API на примере менеджера. Использован подход из А.Лоре «Проектирование веб-арі»

Уточнения:

- в постановке не представлены данные заявки, предполагаю можно уточнить как ФТ позже., поэтому без них, называю обобщенно – «данные»
- номер заявки — это идентификатор с точки зрения создания заявки как ресурса через Post?
- не хватает действия «получить заявку», так как просмотр и поиск возвращают набор заявок и если по каждой получать все параметры – то это объемно, можно возвращать ограниченный набор отображаемых параметров, чтобы не захлмлять результат поиска. Возможно, предполагается использование действия поиск по номеру как действия «получить» заявку.
- Действие «Редактировать заявку» с точки зрения менеджера – это обновить?
- Видит ли менеджер заявки, которые на согласовании – может ли их запросить?
- В каких статусах-состояниях можно редактировать заявку? Можно ли редактировать отправленные на согласования, отклоненные, согласованные? Этого нет в постановке.
- Возможно, нужно действие аутентификации/авторизации для ролей менеджера и директора, но это вне рамок постановки
- Введем статус заявки: 0 – черновик, 1 – новая 2 -согласованная, 3- отклонена

Кто?	Что?	Как?	Входные данные (источник)	Результаты	Цели
------	------	------	---------------------------	------------	------

					(Функциональность)
Контент-менеджер	Управление заявками в каталоге	Создать заявку	Данные заявки (вне рамок задания уточняются на этапе сбора ФТ) (источник – контент менеджер)	Созданная заявка	Создать заявку в каталоге заявок
		Редактировать заявку	Новые данные заявки (источник – контент менеджер, поиск заявок)	Обновленная заявка	Обновить заявку в каталоге заявок
		Отправить заявку на согласование	Новые данные - новый статус заявки (источник – контент менеджер, поиск заявок) (ИЛИ создание новой заявки для отправки на согласование, минус – создание новых сущностей, неидемпотентность)	Обновленная заявка	Обновить заявку в каталоге заявок
		Просмотр согласованных заявок (поля просмотр по действиям)	Поиск заявок в каталоге с параметром «согласовано» (источник – контент менеджер, поиск заявок, директор)	Согласованные заявки	Получить заявки в статусе «согласовано»

		совпадают с поиском)			
		Просмотр отклоненных заявок	Поиск заявок в каталоге с параметром «отклонено» (источник – контент менеджер, поиск заявок, директор)	Отклоненные заявки	Получить заявки в статусе «отклонено»
		Поиск и фильтрация заявок по создателю, дате создания, номеру	Поиск заявок в каталоге по параметрам (источник – контент менеджер, поиск заявок)	Заявки, соответствующие параметру поиска	Получить заявки согласно указанным параметрам

Заготовки для методов:

Функциональность: Добавить заявку в каталог заявок

HTTP-глагол и ресурс: POST /statements/

Что передаем в теле запроса: параметры заявки (нужно уточнять, например: название курса, стоимость, длительность, даты, создатель и тд)

Что получаем в теле ответа: HTTP статус: успех/неуспех, идентификатор (и/или номер) заявки

POST - для уникального начального формирования идентификатора заявки, чтобы не потерять и не грузить менеджера придумыванием идентификаторов

«и/или» так как идентификатор может быть техническим типа uuid, а менеджеру может быть нужен более читабельный номер заявки, например для какого-то отчета, нужно уточнять. Идентификатор нужен для последующих запросов редактирования, просмотра, поиска.

Функциональность: Редактировать заявку

HTTP-глагол и ресурс: PUT /statements/{id}

Что передаем в теле запроса: параметры заявки, которые редактируем, в параметрах идентификатор заявки, который вернул POST

Что получаем в теле ответа: HTTP статус: успех/неуспех, возможно ответ вида «Заявка с номером {номер} отредактирована» или ошибка

Функциональность: Отправить заявку на согласование

HTTP-глагол и ресурс: PUT /statements/{id}

Что передаем в теле запроса: статус заявки «новая» - это «стартовый» статус для директора, устанавливаем statem_status = 0

Что получаем в теле ответа: HTTP статус: успех/неуспех, возможно ответ вида «Заявка с номером {номер} направлена на согласование» или ошибка

Тут возникает вопрос по глаголу «отправить» - просится метод Post, но в этом случае мы порождаем новую сущность + в случае сбоя из-за неидемпотентности post при повторной отправке можем плодить дубли, поэтому отправку решил делать через обновление Put. Иначе если идти через Post, то при смене других статусов пришлось бы делать новые заявки. НО: если архитектурно сервис менеджера и директора это разные МС, то тогда возможно использование POST для отправки в сервис директора, но нужно продумывать как предотвратить дублирование заявок на согласовании и как связывать номера-идентификаторы заявок в двух сервисах, синхронизировать статусы – сложно, не пошел бы этим путем.

Функциональность: Поиск и фильтрация заявки по номеру/идентификатору

HTTP-глагол и ресурс: GET /statements/{id}

Что передаем в теле запроса: ничего

Что получаем в теле ответа: HTTP статус: успех/неуспех, все данные заявки

Введу доп метод для получения полных данных заявки

Функциональность: Поиск и фильтрация заявок по создателю, дате создания, номеру

HTTP-глагол и ресурс: GET /statements?author=value1&date_being=value2&statem_id =value3

Что передаем в теле запроса: ничего

Что получаем в теле ответа: HTTP статус: успех/неуспех, список заявок: идентификаторы (и/или номера) заявок, сокращённый набор данных, чтобы не загружать выдачу пользователю и канал связи, подробные данные можно получить через предыдущий метод GET /statements/{id}

Как вариант добавить сюда еще параметр статус &statem_status =value4 и можно свети в этом метод действия просмотр согласованных и отмененных заявок

Функциональность: Просмотр согласованных заявок

HTTP-глагол и ресурс: GET /statements?statem_status = 2

Что передаем в теле запроса: ничего

Что получаем в теле ответа: HTTP статус: успех/неуспех, список заявок: идентификаторы (и/или номера) заявок, сокращённый набор данных, чтобы не загружать выдачу пользователю и канал связи, подробные данные можно получить через предыдущий метод GET /statements/{id}

Функциональность: Просмотр отклоненных заявок

HTTP-глагол и ресурс: GET /statements?statem_status = 3

Что передаем в теле запроса: ничего

Что получаем в теле ответа: HTTP статус: успех/неуспех, список заявок: идентификаторы (и/или номера) заявок, сокращённый набор данных, чтобы не загружать выдачу пользователю и канал связи, подробные данные можно получить через предыдущий метод GET /statements/{id}

В последних трех методах возможно есть смысл думать о пагинации, но это оставлю на развитие – пусть данный набор будет mvp 😊

Для директора:

Директору тоже нужен метод GET /statements/{id}

- Отклонить заявку

Функциональность: Отклонить заявку

HTTP-глагол и ресурс: PUT /statements/{id}

Что передаем в теле запроса: новый статус заявки statem_status = 3

Что получаем в теле ответа: HTTP статус: успех/неуспех, возможно ответ вида «Заявка с номером {номер} отклонена » или ошибка

- Согласовать заявку

Функциональность: Отклонить заявку

HTTP-глагол и ресурс: PUT /statements/{id}

Что передаем в теле запроса: новый статус заявки statem_status = 2

Что получаем в теле ответа: HTTP статус: успех/неуспех, возможно ответ вида «Заявка с номером {номер} согласована » или ошибка

- Просмотр новых заявок

Функциональность: Просмотр новых заявок

HTTP-глагол и ресурс: GET /statements?statem_status = 0

Что передаем в теле запроса: ничего

Что получаем в теле ответа: HTTP статус: успех/неуспех, список заявок: идентификаторы (и/или номера) заявок, сокращённый набор данных, чтобы не загружать выдачу пользователю и канал связи, подробные данные можно получить через предыдущий метод GET /statements/{id}