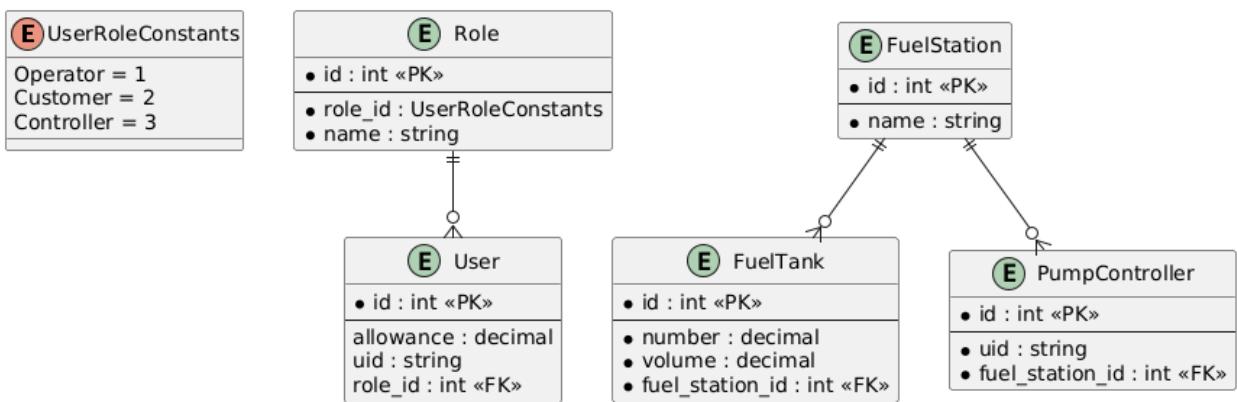


# Спецификация API облачного сервиса управления топливораздаточными колонками

## ER-модель



Комментарии к ролям:

- **Operator**. Оператор АЗС. Имеет право вызывать “Приём топлива”
- **Customer**. Клиент. Имеет право производить заправку.
- **Controller**. Контроллер ТРК в автономном режиме. Имеет право загружать списки пользователей и их параметры в пакетном режиме с целью синхронизации кеша.

## Методы облачного сервиса

Ниже приведены пять основных методов контроллера PumpController и соответствующие объекты передачи данных (DTO):

### 1. Авторизация и начало сессии

**Endpoint:** POST /api/pump/authorize

**Параметры запроса**

**JSON DeviceAuthorizeRequest:**

- PumpControllerUid (string): Уникальный идентификатор контроллера ТРК.
- UserId (string): Уникальный идентификатор пользователя.

## Успешный ответ - HTTP 200 Ok

### JSON DeviceAuthorizeResponse:

- Token (string): JWT-токен для последующих вызовов.
- RoleId (int): Идентификатор роли пользователя.
- FuelTanks (List<FuelTankItem>): Список резервуаров, на АЗС, к которой “приписан” контроллер. Для каждого резервуара нужно передать как минимум number и volume.
- Allowance (decimal): Оставшийся объём, разрешённый для заправки (только, если роль “Customer” (2)).
- Price (decimal?): Цена за литр (только, если роль “Customer” (2)). Необязательно даже для клиента. Если цена указана, заправку можно производить в рублях и в литрах. Если не указана, только в литрах.

## Конец сессии и деавторизация

- **Endpoint:** POST /api/pump/deauthorize
- **Параметры запроса:** отсутствуют, токен берётся из заголовков
- **Успешный ответ:** HTTP 204 No Content.

Сервису рекомендуется учесть возможное совпадение маловероятных обстоятельств, при которых контроллер ТРК не проведёт деавторизацию. Для защиты от такого случая необходимо периодически удалять из внутреннего кеша сессии с истёкшим сроком давности.

## Отчёт о приёме топлива

- Endpoint: POST /api/pump/fuelintake
- Запрос может выполняться только после авторизации и, если у клиента роль “Operator” (1)
- **Параметры запроса**  
**JSON FuelIntakeRequest:**
  - TankNumber (decimal): Номер резервуара.
  - IntakeVolume (decimal): Объём принятого топлива.
- **Успешный ответ:** HTTP 204 No Content.

## Отчёт о заправке

- Endpoint: POST /api/pump/refuel
- Запрос может выполняться только после авторизации и, если роль “Customer” (2)
- **Параметры запроса**  
**JSON FuelIntakeRequest:**
  - TankNumber (decimal): Номер резервуара.
  - FuelVolume (decimal): Объём заправленного топлива.
- **Успешный ответ:** HTTP 204 No Content.

## Запрос информации о пользователях для синхронизации

- Endpoint: GET /api/pump/user?first=<int>&number=<int>

- Запрос может выполняться только после авторизации и, если у клиента есть роль “Controller” (3)
- Параметры запроса:
  - first(int): Номер первого запрашиваемого элемента в списке пользователей, начиная с 0
  - number (int): Количество элементов.

Желательно возвращать только пользователей с ролями “Operator” (1), “Customer” (2). Список можно ограничить теми пользователями, у которых есть право работать на данной АЗС.

- **Успешный ответ - HTTP 200 Ok** с List<PumpUserItem> или **HTTP 204 No Content**, если в указанном диапазоне пользователей нет.

#### JSON PumpUserItem

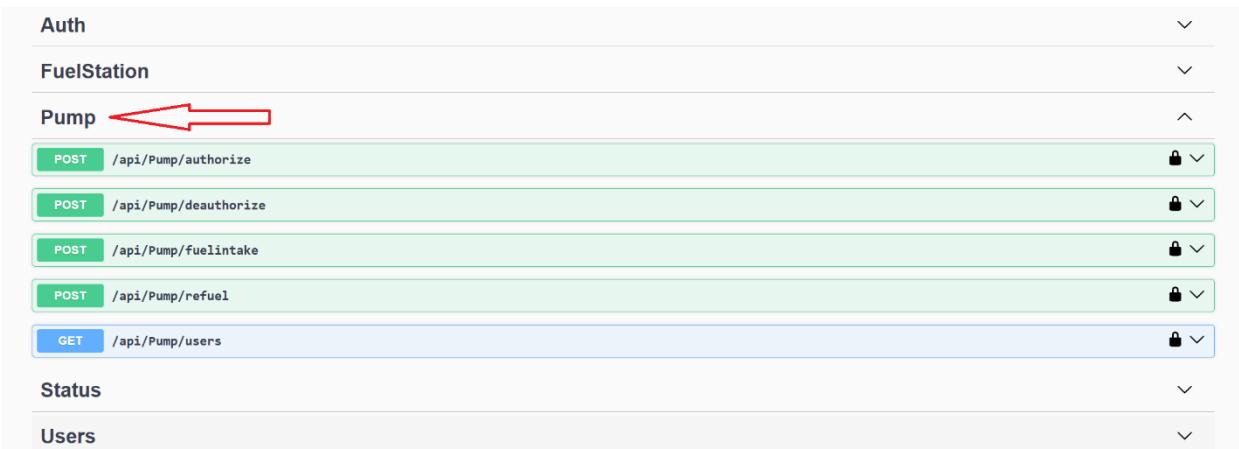
- Uid (string): Уникальный идентификатор пользователя.
- RoleId (int): Идентификатор роли пользователя.
- Allowance (decimal): Оставшийся объём, разрешённый для заправки (только, если роль “Customer” (2)).

Количество записей в списке может быть меньше, чем запрошено в Number

## Swagger

<https://fuelflux.sw.consulting:8087/swagger/index.html>

Прототипы методов, описанных в этом документе, реализованы в контроллере Pump



The screenshot shows the Swagger UI interface for the Pump API. The 'Auth' section is expanded, showing the 'FuelStation' section. Under 'FuelStation', the 'Pump' section is expanded, highlighting the '/api/Pump/users' endpoint with a red arrow. This endpoint is a GET method. Other endpoints listed under 'Pump' include POST methods for '/api/Pump/authorize', '/api/Pump/deauthorize', '/api/Pump/fuelintake', and '/api/Pump/refuel'. The 'Status' and 'Users' sections are also visible but collapsed.

В системе созданы 3 пользователя

Uid 1111-1111-1111-1111	Роль “Operator”
Uid 2222-2222-2222-2222	Роль “Customer”
Uid 3333-3333-3333-3333	Роль “Controller”

И 2 контроллера с Uid 1111-1111, 1111-2222

## История изменений

Версия	Дата	Описание
0.1	25.07.2025	Рабочий вариант для обсуждения