**Технические требования**

**по организации взаимодействия микросервисов**

**в рамках внедрения проекта СКУД АО «Минудобрения»**

1. Настоящие технические требования описывают порядок взаимодействия прикладных программных систем в микросервисной архитектуре, внедряемой в рамках проекта СКУД АО «Минудобрения».
2. Описание используемых технологий
   1. Для взаимодействия микросервисов используется REST интерфейс, основанный на HTTP 1.1 протоколе с единственным используемым типом контента — сообщения в формате JSON ('Content-Type': 'application/json').
   2. Для HTTP запросов используются только следующие типы методов:

|  |  |
| --- | --- |
| Метод | Назначение |
| GET | Запрос данных, требующий ответа от сервиса |
| POST | Оповещение о событии или команда на создание объекта |
| PATCH | Команда на корректировку данных существующего объекта |
| DELETE | Команда на удаление объекта |

* 1. HTTP URL содержит указание на конкретный ресурс, к которому осуществляется запрос в формате '/microservice/v1/resource-name', где:
* microservice — уникальная строка, характеризующая запрос к брокеру сообщений микросервисов;
* v1 — версия протокола интеграции;
* resource-name — наименование ресурса, например, invoice, pass и т.д.
  1. Сообщение JSON содержит ровно один объект со следующими полями:

| Поле | Тип | Назначение | Запол­няется | |
| --- | --- | --- | --- | --- |
| инициатором | целью |
| initiator | строка | Краткое название микросервиса, инициировавшего запрос | + | + |
| target | строка | Краткое название микросервиса, отвечающего на запрос | - | + |
| request | объект | Объект с необходимыми параметрами для формирования ответа | + | + |
| response | список | Список объектов с результатами исполнения запроса | - | \* |
| count | число | Количество объектов в списке response | - | + |
| first | число | Номер первого объекта в списке response | \* | + |
| last | число | Номер последнего объекта в списке response | - | + |
| total | число | Общее количество объектов, удовлетворяющих запросу | - | + |
| limit | число | Максимальное количество объектов, передаваемых за один запрос | \* | \* |
| hide | список | Список полей объекта, которые не должны включаться в ответ | \* | \* |
| show | список | Список полей, кроме которых остальные не должны включаться в ответ | \* | \* |
| status | строка | Статус ответа. Один из вариантов:  - completed — запрос выполнен;  - partial — запрос выполнен частично;  - check — ошибка в запросе была проигнорирована;  - error — ошибка в запросе не позволила сформировать ответ;  - critical — системная ошибка (например, нет связи с СУБД) | - | + |
| description | строка | Описание ошибки в человекочитаемом формате | - | + |
| delivered | список | Список наименований оповещённых сервисов | \* | \* |
| undelivered | список | Список наименований неоповещённых сервисов для частично доставленных оповещений | \* | \* |

где '+' означает обязательное заполнение, '\*' - возможное заполнение, '-' - запрет на заполнение.

Микросервис-цель при ответе инициатору запроса повторяет поля, относящиеся к запросу, дополняя их полями с ответом.

В случае, если инициатор ошибочно заполнил поле в запросе, микросервис-цель в ответе должен заменить это поле на правильное значение (которое использовалось для формирования ответа). При этом делается запись в лог с уровнем warn.

При запросе, подразумевающего список объектов в ответе, указываются параметры 'first' и 'limit':

| first | limit | Описание |
| --- | --- | --- |
| нет | нет | Требуются все подходящие под ответ объекты с первого и до последнего. При этом целевой сервис может самостоятельно установить limit в ответе |
| нет | есть | Требуются объекты с начала в количестве, не превышающем limit |
| есть | нет | Требуются все объекты, начиная (включительно) с номера, указанного в first, и до конца. Нумерация объектов начинается с 0 |
| есть | есть | Требуются объекты, начиная (включительно) с номера, указанного в first, в количестве, не превышающем limit |

В запросе может быть только одно поле 'hide' или 'show'. В ответе сами поля повторяются в том же виде, как пришли в запросе. Если в списке значений указано поле, не определённое для данного ресурса, оно игнорируется, запрос исполняется. Если в результате список 'show' оказывается пустым, запрос не исполняется. В зависимости от результата устанавливаются соответствующие 'status' и 'description'.

Поле 'delivered' заполняется целевым сервисом при ответе списком из одного собственного названия сервиса в случае, если запрос выполнен успешно ('status':'completed'). При многоадресных оповещениях брокер собирает все ответы от сервисов, формируя итоговый список 'delivered'. Все сервисы, не вернувшие своё наименование в поле 'delivered', собираются в список 'undelivered'. Пустые списки не передаются. Если список ничего не содержит, поле в сообщение не добавляется. Инициатор получает от брокера один ответ с итоговыми списками 'delivered' и 'undelivered'. В случае, когда при многоадресном оповещении в ответе присутствуют оба списка, поле 'status' устанавливается брокером в состояние 'partial'.

* 1. Кодировка символов в сообщении — UTF-8.
  2. Ответы на HTTP запросы могут содержать следующие коды ответов:

| Код ответа | Назначение |
| --- | --- |
| 200 | Запрос принят, ответ сформирован |
| 401 | Требуется авторизация доступа |
| любой другой код | Системная ошибка. Описание соответствует стандарту HTTP. Требуется повторить запрос |

* 1. Авторизация запросов осуществляется по дайджестной технологии (HTTP AuthType Digest). В качестве Realm используется краткое наименование сервиса.

1. Описание взаимодействия микросервисов
   1. Каждый микросервис обменивается REST сообщениями только с брокером. Брокер осуществляет дальнейшую маршрутизацию запросов и ответов.
   2. Со стороны каждого микросервиса необходимо реализовать REST клиент и REST сервер. Клиент запрашивает данные у других микросервисов, рассылает оповещения. Сервер отвечает на запросы других микросервисов, принимает оповещения.
   3. REST клиент работает по схеме: отправка запроса → ожидание ответа → в случае получения HTTP Response кода 200 производится обработка полученных данных, в случае кода 401 производится авторизация и повтор запроса, в ином случае (другой код или таймаут) производится запись в журнал ошибок и повтор запроса (но не более определённого количества раз).
   4. При получении клиентом в ответе 'status':'partial' необходимо повторить запрос, добавив в сообщение поле 'undelivered' со значением, ранее полученным в ответе.
   5. REST сервер функционирует следующим образом: поступление запроса → выбор обработчика на основании URL и HTTP метода (если URL отсутствует, возвращается ответ с кодом 404; если для данного URL запрошенный метод не поддерживается, то с кодом 405) → передача JSON сообщения обработчику → ожидание JSON ответа → отправка HTTP пакета с ответом;
   6. В обработчике запросов со стороны REST сервера необходимо применять логику, ограничивающую избыточные затраты ресурсов. Например, устанавливать значение 'limit', если не задано в запросе, исходя из реальных возможностей обработчика, и/или отказывать в исполнении «тотальных» запросов без каких-либо ограничивающих условий.
2. Требуемые настройки
   1. На клиенте, инициирующем REST запросы необходимо установить следующие настройки протокола TCP:

* tcp\_keepalive\_time = 300 (сек)
* tcp\_keepalive\_probes = 3
* tcp\_keepalive\_intvl = 20 (сек)
  1. Каждый сервис должен предусматривать сохранение следующих настроек интерфейса интеграции:
* IP адрес (имя хоста) и TCP порт брокера сообщений микросервисов;
* логин и пароль для аутентификации;
* краткое наименование собственного сервиса согласно таблице:

| Обозначение | Описание |
| --- | --- |
| start2000 | ПО СТАРТ2000 (автоматизированная система управления предприятием) |
| personnel | ПО Босс-Кадровик (система управления персоналом) |
| acs | ПО Сфинкс (система контроля и управления доступом) |
| scch | ПО 1С:Трактиръ (управление абонементами спорткомплекса «Химик») |
| canteen | ПО 1С:Трактиръ (автоматизация фронт-офиса цеха общественного питания) |
| catering | ПО Iiko (автоматизация фронт-офиса кейтеринговой компании) |
| mealticket | Модуль УТЛПП (учёт талонов на лечебно-профилактическое питание) |
| infringement | Модуль УРВ (учёт рабочего времени) |
| justification | Модуль УОД (учёт оправдательных документов) |
| scales | ПО Весовая (весовой учёт грузов) |

1. Описание работы методов GET, POST, PATCH, DELETE
   1. По результатам выполнения запроса GET сформированный ответ должен содержать список объектов, определённых для конкретного ресурса. В запросе могут использоваться поля ресурса с целью фильтрации выдаваемых результатов. При этом для поля field могут применяться поля:

* field — точное совпадение с указанным значением или шаблоном;
* fieldMin — не менее указанного значения;
* fieldMax — не более указанного значения.
  1. Шаблон для GET запроса может содержать следующие служебные символы:
* '%' - любая строка из ноля или более символов;
* '-' - любой одиночный символ;
* '[' и ']' - любой символ из перечисленных (например, '[abcde]');
* '[^' и ']' - любой символ кроме перечисленных (например, '[^vwxyz]').
  1. Методы POST, PATCH и DELETE предполагают создание, корректировку и удаление объекта соответственно. Состав полей в запросе должен соответствовать структуре объекта для заданного ресурса.

1. Перечень использующихся ресурсов в текущей микросервисной архитектуре (v1) приведён в Приложении А.
2. Реестр изменений документа:

| Версия | Дата | Перечень правок |
| --- | --- | --- |
| v01 | 21.04.2017 | Начальная редакция |
| v02 | 22.06.2017 | Добавлено Приложение Б, внесены правки в ресурс /microservice/v1/pass с целью оптимизации интеграции микросервисов acs и scch |
| v03 | 29.06.2017 | Генерация изображений QR-кодов (используется PrinterTicket) |
| v04 | 28.04.2018 | Добавлены микросервисы СК Химик. Внесены корректировки в остальные с учётом фактической реализации |
| v05 | 08.08.2018 | Добавлен микросервис order |
| v06 | 18.11.2019 | Добавлены микросервисы scchAccess, accessPoints, mealTicket, luncheon, deduction, closeLuncheon |

Приложение А

**Перечень ресурсов с описанием**

|  |  |
| --- | --- |
| Ресурс | **/microservice/v1/pass** |
| Тип | запрос |
| Микросервис | **acs** |
| Описание | данные о выданных пропусках для автотранспорта покупателей, гостей предприятия, клиентов спорткомплекса |

Структура объекта:

| **Поле** | **Тип** | **Описание** | **GET запрос** | **GET ответ** | **POST** | **PATCH** | **DELETE** |
| --- | --- | --- | --- | --- | --- | --- | --- |
| passId | строка | Идентификатор пропуска | \* | + |  | + | \* |
| passDate | строка | Дата начала действия пропуска (здесь и далее даты в формате 'ГГГГ-ММ-ДД') | \* | + |  | \* | - |
| passEnd | строка | Дата последнего дня действия пропуска | \* | + |  | \* | - |
| create | число | Признак необходимости создания нового пропуска:  1 — создавать, если не найден подходящий;  2 — сгенерировать новый идентификатор (для QR-кода) в существующем пропуске;  все остальные варианты — не создавать никогда, ничего не менять | \* | + |  | - | - |
| humanId | строка | Идентификатор физлица (для сотрудников предприятия — последовательность цифр, для остальных — префикс EXT и последовательность из 12 цифр) | \* | + |  | - | - |
| name | строка | Полное ФИО получателя пропуска (водителя, гостя, клиента спорткомплекса) | \* | + |  | \* | - |
| docType | строка | Тип документа, подтверждающего личность:  passport — паспорт гражданина РФ;  license — водительское удостоверение;  other — иной документ | \* | + |  | \* | - |
| docNum | строка | Серия и номер документа через один пробел, буквы латиницей | \* | + |  | \* | - |
| passType | строка | Тип пропуска:  employee — сотрудник предприятия;  filial — сотрудник дочернего предприятия;  temporary — временный пропуск;  guest — посетитель предприятия;  scchTicket — абонемент спорткомплекса;  car — автотранспорт предприятия;  buyerTruck — автотранспорт покупателей. | \* | + |  | \* | \* |
| description | строка | Дополнительное описание пропуска, выводимое на КПП (ФИО пригласившего, номера накладных, тип груза и т. п.) | \* | + |  | \* | - |

где '+' означает, поле обязательно к заполнению точным значением, '-' - поле не может быть указано в запросе, '\*' - поле может быть заполнено, в запросе GET допускается использование шаблонов, указание диапазонов.

Дополнительные поля для 'passType':'buyerTruck':

| **Поле** | **Тип** | **Описание** | **GET запрос** | **GET ответ** | **POST** | **PATCH** | **DELETE** |
| --- | --- | --- | --- | --- | --- | --- | --- |
| carRegNum | строка | Государственный регистрационный номер транспортного средства | \* | + |  | - | - |
| carModel | строка | Марка и модель транспортного средства | \* | + |  | - | - |

Дополнительные поля для 'passType':'scchClient' и 'passType':'employee':

| **Поле** | **Тип** | **Описание** | **GET запрос** | **GET ответ** | **POST** | **PATCH** | **DELETE** |
| --- | --- | --- | --- | --- | --- | --- | --- |
| scchClient | число | Признак наличия действующего абонемента в спорткомплекс:  1 — абонемент есть;  любое другое значение — абонемента на пропуске нет. | \* | + |  | \* | - |
| scchSchedule | строка | Идентификатор расписания занятий | \* | + |  | \* | - |
| scchZone | строка | Зона доступа по абонементу:  poolMale — малый бассейн, мужская раздевалка  poolFemale — малый бассейн, женская раздевалка  common — общая зона  full — полный доступ  blocked — доступ в СК заблокирован | \* | + |  | \* | - |

Правила работы с ресурсом /microservice/v1/pass изложены в Приложении Б.

|  |  |
| --- | --- |
| Ресурс | **/microservice/v1/zone** |
| Тип | запрос |
| Микросервис | **acs** |
| Описание | данные о контролируемых СКУД зонах |

Структура объекта:

| **Поле** | **Тип** | **Описание** | **GET запрос** | **GET ответ** | **POST** | **PATCH** | **DELETE** |
| --- | --- | --- | --- | --- | --- | --- | --- |
| code | строка | Код зоны | \* | + |  |  |  |
| name | строка | Краткое наименование зоны | \* | + |  |  |  |
| description | строка | Полное наименование зоны | \* | + |  |  |  |

|  |  |
| --- | --- |
| Ресурс | **/microservice/v1/grant** |
| Тип | запрос |
| Микросервис | **acs** |
| Описание | сведения о назначенных специальных режимах доступа |

Структура объекта:

| **Поле** | **Тип** | **Описание** | **GET запрос** | **GET ответ** | **POST** | **PATCH** | **DELETE** |
| --- | --- | --- | --- | --- | --- | --- | --- |
| employeeId | строка | Идентификатор физлица для сотрудников предприятия | \* | + |  |  |  |
| code | строка | Код режима:  СД — свободный доступ;  КД — круглосуточный доступ;  РВ — рейсовый водитель. | \* | + |  |  |  |
| name | строка | Наименование режима | - | + |  |  |  |

При запросе допускается указание в 'employeeId' списка значений. В этом случае в ответе возвращается список объектов по одному для каждого найденного 'employeeId'.

|  |  |
| --- | --- |
| Ресурс | **/microservice/v1/accessPoints** |
| Тип | запрос |
| Микросервис | **acs** |
| Описание | сведения о точках доступа СКУД |

Структура объекта:

| **Поле** | **Тип** | **Описание** | **GET запрос** | **GET ответ** | **POST** | **PATCH** | **DELETE** |
| --- | --- | --- | --- | --- | --- | --- | --- |
| apCode | число | Номер точки прохода | + | + |  |  |  |
| apName | строка | Наименование точки прохода | - | + |  |  |  |

При запросе допускается указание в 'apCode' списка значений. В этом случае в ответе возвращается список объектов по одному для каждого найденного 'apCode'.

|  |  |
| --- | --- |
| Ресурс | **/microservice/v1/access** |
| Тип | оповещение |
| Микросервис | **acs** |
| Описание | оповещение о необходимости переформирования расписания доступа |

Структура объекта:

| **Поле** | **Тип** | **Описание** | **GET запрос** | **GET ответ** | **POST** | **PATCH** | **DELETE** |
| --- | --- | --- | --- | --- | --- | --- | --- |
| employeeId | строка | Идентификатор физлица для сотрудников предприятия |  |  | + |  |  |
| dateFrom | строка | Дата первого дня периода с изменениями |  |  | \* |  |  |
| dateTo | строка | Дата последнего дня периода отчёта |  |  | \* |  |  |

Допускается указание в 'employeeId' списка значений.

При отсутствии одного любого поля 'dateTo' или 'dateFrom' принимается период, равный одному дню.

При отсутствии обоих полей 'dateFrom' и 'dateTo' подразумеваются текущие календарные сутки по московскому времени.

|  |  |
| --- | --- |
| Ресурс | **/microservice/v1/acsEvent** |
| Тип | запрос |
| Микросервис | **acs** |
| Описание | перечень событий контроля доступа |

Структура объекта:

| **Поле** | **Тип** | **Описание** | **GET запрос** | **GET ответ** | **POST** | **PATCH** | **DELETE** |
| --- | --- | --- | --- | --- | --- | --- | --- |
| time | строка | Время события | + | + |  |  |  |
| date | строка | Дата события | \* | + |  |  |  |
| passId | строка | Идентификатор пропуска | \* | + |  |  |  |
| passType | строка | Тип пропуска:  employee — сотрудник предприятия;  filial — сотрудник дочернего предприятия;  temporary — временный пропуск;  guest — посетитель предприятия;  scchTicket — абонемент спорткомплекса;  car — автотранспорт предприятия;  buyerTruck — автотранспорт покупателей. | \* | + |  |  |  |
| eventType | строка | Тип события:  in — проход в зону;  out — выход из зоны. | \* | + |  |  |  |
| eventDescription | строка | Описание события в человекочитаемом формате | - | + |  |  |  |
| zone | строка | Код контроллируемой зоны | \* | + |  |  |  |
| status | строка | Статус события:  permit — разрешено;  sanction — разрешено с санкции охраны;  approval — санкция охраны подтверждена оправдательным документом;  deny — запрещено;  alcohol — запрещено из-за алкотеста;  timeout — запрещено из-за незавершения процедуры. | \* | + |  |  |  |
| alcohol | число | Содержание спирта в выдыхаемом воздухе (мг/л) | - | \* |  |  |  |

Дополнительные поля для 'passType':'employee':

| **Поле** | **Тип** | **Описание** | **GET запрос** | **GET ответ** | **POST** | **PATCH** | **DELETE** |
| --- | --- | --- | --- | --- | --- | --- | --- |
| employeeId | строка | Идентификатор физлица для сотрудников предприятия | - | \* |  |  |  |

|  |  |
| --- | --- |
| Ресурс | **/microservice/v1/scchAccess** |
| Тип | оповещение |
| Микросервис | **scch** |
| Описание | оповещение о проходе клиента через турникеты спорткомплекса |

Структура объекта:

| **Поле** | **Тип** | **Описание** | **GET запрос** | **GET ответ** | **POST** | **PATCH** | **DELETE** |
| --- | --- | --- | --- | --- | --- | --- | --- |
| time | строка | Время события |  |  | + |  |  |
| date | строка | Дата события |  |  | + |  |  |
| passId | строка | Идентификатор пропуска |  |  | + |  |  |
| humanId | строка | Идентификатор физлица клиента спорткомплекса |  |  | + |  |  |
| passType | строка | Тип пропуска всегда:  scchTicket — абонемент спорткомплекса |  |  | + |  |  |
| eventType | строка | Тип события:  in — проход в зону;  out — выход из зоны. |  |  | + |  |  |
| scchZone | строка | Зона доступа по абонементу:  poolMale — малый бассейн, мужская раздевалка  poolFemale — малый бассейн, женская раздевалка  common — общая зона  full — полный доступ  blocked — доступ в СК заблокирован |  |  | + |  |  |
| apCode | число | Номер точки доступа |  |  | + |  |  |

|  |  |
| --- | --- |
| Ресурс | **/microservice/v1/scchSchedule** |
| Тип | оповещение |
| Микросервис | **acs** |
| Описание | данные об изменениях расписаний прохода на территорию спорткомплекса «Химик» |

Структура объекта:

| **Поле** | **Тип** | **Описание** | **GET запрос** | **GET ответ** | **POST** | **PATCH** | **DELETE** |
| --- | --- | --- | --- | --- | --- | --- | --- |
| code | строка | Код расписания |  |  | + | + | + |
| name | строка | Наименование расписания |  |  | + | \* | - |
| interval | список | Список временных интервалов |  |  | + | \* | - |

Объекты списка 'interval':

| **Поле** | **Тип** | **Описание** | **GET запрос** | **GET ответ** | **POST** | **PATCH** | **DELETE** |
| --- | --- | --- | --- | --- | --- | --- | --- |
| day | строка | День недели ('пн', 'вт', ...) |  |  | + | \* | - |
| from | строка | Время начала интервала в формате ЧЧ:ММ |  |  | + | \* | - |
| to | список | Время окончания интервала в формате ЧЧ:ММ |  |  | + | \* | - |

|  |  |
| --- | --- |
| Ресурс | **/microservice/v1/employee** |
| Тип | запрос |
| Микросервис | **personnel** |
| Описание | данные о сотрудниках предприятия |

Структура объекта:

| **Поле** | **Тип** | **Описание** | **GET запрос** | **GET ответ** | **POST** | **PATCH** | **DELETE** |
| --- | --- | --- | --- | --- | --- | --- | --- |
| employeeId | строка | Идентификатор физлица для сотрудников предприятия | \* | + |  |  |  |
| syncDate | строка | Дата актуальности данных о сотруднике | \* | + |  |  |  |
| firstName | строка | Имя | \* | + |  |  |  |
| midName | строка | Отчество | \* | + |  |  |  |
| lastName | строка | Фамилия | \* | + |  |  |  |
| department | строка | Корневое подразделение | \* | + |  |  |  |
| appointment | строка | Должность | \* | + |  |  |  |
| personnelNumber | строка | Табельный номер | \* | + |  |  |  |
| nutrition | строка | Категория спецпитания:  milk — выдача молока;  money — компенсация молока деньгами;  standard — стандартное ЛПП;  enchanced — усиленное ЛПП. | \* | + |  |  |  |
| alcotest | строка | Категория по алкотесту:  white — белый список (не проверять);  gray — серый список (общие правила проверки);  black — чёрный список (проверять каждый раз);  driver — водитель (проверять не реже 1 раза в смену). | \* | + |  |  |  |
| chief | число | 1 — имеет право подписи оправдательных документов (хотя бы одного вида);  0 — не имеет права подписи никаких оправдательных документов | \* | + |  |  |  |

Если в запросе присутствует поле 'syncDate', возвращаются сведения о сотруднике на запрошенную дату. Если поле 'syncDate' отсутствует, выдаются сведения на текущий календарный день по фактическому времени обращения в локальном часовом поясе.

|  |  |
| --- | --- |
| Ресурс | **/microservice/v1/schedule** |
| Тип | запрос |
| Микросервис | **personnel** |
| Описание | данные о режиме труда |

Структура объекта:

| **Поле** | **Тип** | **Описание** | **GET запрос** | **GET ответ** | **POST** | **PATCH** | **DELETE** |
| --- | --- | --- | --- | --- | --- | --- | --- |
| employeeId | строка | Идентификатор физлица для сотрудников предприятия | + | + |  |  |  |
| date | строка | Дата начала смены | + | + |  |  |  |
| dayType | строка | Предварительный тип дня ('Я', 'Б', 'К'...) | - | + |  |  |  |
| schedule | список | Список временных событий | - | + |  |  |  |

Поле 'employeeId' в запросе может содержать список строк — идентификаторов физлиц. В этом случае в ответе возвращаются отдельные объекты для каждого найденного идентификатора.

Структура объектов списка 'schedule':

| **Поле** | **Тип** | **Описание** | **GET запрос** | **GET ответ** | **POST** | **PATCH** | **DELETE** |
| --- | --- | --- | --- | --- | --- | --- | --- |
| type | строка | Тип события ('Начало прихода', 'Приход', ...) | - | + |  |  |  |
| time | строка | Время события в формате ЧЧ:ММ | - | + |  |  |  |

|  |  |
| --- | --- |
| Ресурс | **/microservice/v1/territoryAccess** |
| Тип | оповещение |
| Микросервис | **infringement** |
| Описание | уведомление о событии контроля доступа |

Структура объекта:

| **Поле** | **Тип** | **Описание** | **GET запрос** | **GET ответ** | **POST** | **PATCH** | **DELETE** |
| --- | --- | --- | --- | --- | --- | --- | --- |
| time | строка | Время события |  |  | + |  |  |
| passId | строка | Идентификатор пропуска |  |  | + |  |  |
| passType | строка | Тип пропуска:  employee — сотрудник предприятия;  filial — сотрудник дочернего предприятия;  temporary — временный пропуск;  guest — посетитель предприятия;  scchTicket — абонемент спорткомплекса;  car — автотранспорт предприятия;  buyerTruck — автотранспорт покупателей. |  |  | + |  |  |
| eventType | строка | Тип события:  in — проход в зону;  out — выход из зоны. |  |  | + |  |  |
| eventDescription | строка | Описание события в человекочитаемом формате |  |  | + |  |  |
| zone | строка | Код контроллируемой зоны |  |  | + |  |  |
| status | строка | Статус события:  permit — разрешено;  sanction — разрешено с санкции охраны;  approval — санкция охраны подтверждена оправдательным документом;  deny — запрещено;  alcohol — запрещено из-за алкотеста;  timeout — запрещено из-за незавершения процедуры. |  |  | + |  |  |
| alcohol | число | Содержание спирта в выдыхаемом воздухе (мг/л) |  |  | \* |  |  |
| employeeId | строка | Идентификатор физлица для сотрудников предприятия |  |  | \* |  |  |

|  |  |
| --- | --- |
| Ресурс | **/microservice/v1/violation** |
| Тип | оповещение |
| Микросервис | **personnel** |
| Описание | оповещение о нарушениях режима труда |

Структура объекта:

| **Поле** | **Тип** | **Описание** | **GET запрос** | **GET ответ** | **POST** | **PATCH** | **DELETE** |
| --- | --- | --- | --- | --- | --- | --- | --- |
| employeeId | строка | Идентификатор физлица для сотрудников предприятия |  |  | + | + | + |
| date | строка | Дата начала смены |  |  | + | + | \* |
| time | строка | Время нарушения (включая дату) |  |  | + | + | \* |
| type | число | Тип нарушения:  1 - Слишком ранний приход  2 - Слишком поздний уход  3 - Несанкционированный приход в выходной  4 - Нахождение в торговой точке в рабочее время  5 - Отсутствие отметки о работе на удаленной площадке  6 - Опоздание к началу смены  7 - Опоздание с перерыва  8 - Несанкционированный уход |  |  | + | + | - |
| description | строка | Описание нарушения одной строкой в человекочитаемом формате |  |  | + | + | - |

|  |  |
| --- | --- |
| Ресурс | **/microservice/v1/timeInterval** |
| Тип | запрос |
| Микросервис | **infringement** |
| Описание | список временных интервалов, доступных для прохода сотрудников на предприятие |

Структура объекта:

| **Поле** | **Тип** | **Описание** | **GET запрос** | **GET ответ** | **POST** | **PATCH** | **DELETE** |
| --- | --- | --- | --- | --- | --- | --- | --- |
| employeeId | строка | Идентификатор физлица для сотрудников предприятия | + | + |  |  |  |
| timeFrom | строка | Время начала временного периода | + | + |  |  |  |
| timeTo | строка | Время окончания временного периода | + | + |  |  |  |
| interval | список | Список интервалов разрешённого прохода | - | + |  |  |  |

Структура объектов списка 'interval':

| **Поле** | **Тип** | **Описание** | **GET запрос** | **GET ответ** | **POST** | **PATCH** | **DELETE** |
| --- | --- | --- | --- | --- | --- | --- | --- |
| date | строка | Дата интервала | - | + |  |  |  |
| from | строка | Время начала интервала в формате ЧЧ:ММ | - | + |  |  |  |
| to | строка | Время окончания интервала в формате ЧЧ:ММ | - | + |  |  |  |

|  |  |
| --- | --- |
| Ресурс | **/microservice/v1/workTime** |
| Тип | оповещение |
| Микросервис | **infringement** |
| Описание | уведомление об изменении документов, влияющих на расписание рабочего времени сотрудников |

Структура объекта:

| **Поле** | **Тип** | **Описание** | **GET запрос** | **GET ответ** | **POST** | **PATCH** | **DELETE** |
| --- | --- | --- | --- | --- | --- | --- | --- |
| employeeId | строка | Идентификатор физлица для сотрудников предприятия |  |  | + |  |  |
| dateFrom | строка | Дата первого дня периода с изменениями |  |  | + |  |  |
| dateTo | строка | Дата последнего дня периода отчёта |  |  | + |  |  |

Допускается указание в 'employeeId' списка значений.

|  |  |
| --- | --- |
| Ресурс | **/microservice/v1/justification** |
| Тип | запрос |
| Микросервис | **justification** |
| Описание | сведения об оправдательных документах |

Структура объекта:

| **Поле** | **Тип** | **Описание** | **GET запрос** | **GET ответ** | **POST** | **PATCH** | **DELETE** |
| --- | --- | --- | --- | --- | --- | --- | --- |
| employeeId | строка | Идентификатор физлица для сотрудников предприятия | + | + |  |  |  |
| docType | строка | Тип оправдательного документа | \* | + |  |  |  |
| docNumber | строка | Номер оправдательного документа | - | + |  |  |  |
| docDate | строка | Дата формирования оправдательного документа | - | + |  |  |  |
| zone | строка | Код контролируемой зоны | \* | \* |  |  |  |
| date | строка | Дата действия оправдательного документа | + | + |  |  |  |
| timeFrom | строка | Время начала действия оправдательного документа | - | + |  |  |  |
| timeEarly | строка | Минимальное время разрешённого возврата из увольнительной | - | \* |  |  |  |
| timeTo | строка | Время окончания действия оправдательного документа | - | + |  |  |  |

|  |  |
| --- | --- |
| Ресурс | **/microservice/v1/weight** |
| Тип | запрос |
| Микросервис | **scales** |
| Описание | сведения о взвешиваниях отгружаемой продукции |

Структура объекта:

| **Поле** | **Тип** | **Описание** | **GET запрос** | **GET ответ** | **POST** | **PATCH** | **DELETE** |
| --- | --- | --- | --- | --- | --- | --- | --- |
| weighingId | число | Идентификатор взвешивания | \* | + |  |  |  |
| weighingTime | строка | Дата и время взвешивания с грузом | \* | + |  |  |  |
| weightEmpty | число | Вес тары | \* | + |  |  |  |
| weightBrutto | число | Вес брутто | \* | + |  |  |  |
| weightNetto | число | Вес нетто | \* | + |  |  |  |
| carRegNum | строка | Государственный регистрационный номер транспортного средства | \* | + |  |  |  |
| carModel | строка | Марка и модель транспортного средства | \* | + |  |  |  |
| driverName | строка | ФИО водителя | \* | + |  |  |  |
| driverLicense | строка | Номер водительского удостоверения (буквы кириллицей, между серией и номером один пробел) | \* | + |  |  |  |
| invoice | строка | Номер накладной ТОРГ-12 | \* | + |  |  |  |
| invoiceDate | строка | Дата накладной ТОРГ-12 | \* | + |  |  |  |
| carrier | строка | Краткое наименование перевозчика | \* | + |  |  |  |
| carrierTIN | строка | ИНН перевозчика | \* | + |  |  |  |
| contractor | строка | Краткое наименование покупателя | \* | + |  |  |  |
| contractorTIN | строка | ИНН покупателя | \* | + |  |  |  |
| loadType | строка | Идентификатор груза | \* | + |  |  |  |

|  |  |
| --- | --- |
| Ресурс | **/microservice/v1/invoice** |
| Тип | запрос |
| Микросервис | **start2000** |
| Описание | данные об отгрузках готовой продукции |

Структура объекта:

| **Поле** | **Тип** | **Описание** | **GET запрос** | **GET ответ** | **POST** | **PATCH** | **DELETE** |
| --- | --- | --- | --- | --- | --- | --- | --- |
| invoice | строка | Номер накладной ТОРГ-12 | \* | + |  | + |  |
| invoiceDate | строка | Дата накладной ТОРГ-12 | \* | + |  | + |  |
| contractor | строка | Краткое наименование покупателя | \* | + |  | - |  |
| contractorTIN | строка | ИНН покупателя | \* | + |  | - |  |
| carrier | строка | Краткое наименование перевозчика | \* | + |  | - |  |
| carrierTIN | строка | ИНН перевозчика | \* | + |  | - |  |
| loadType | строка | Идентификатор груза | \* | + |  | - |  |
| weight | число | Масса по накладной | \* | + |  | - |  |
| weightEmpty | число | Вес тары | \* | + |  | \* |  |
| weightBrutto | число | Вес брутто | \* | + |  | \* |  |
| weightNetto | число | Вес нетто | \* | + |  | \* |  |
| document | строка | Статус отправки отгрузочных документов:  none — документы не готовы;  ready — документы сформированы;  signed — документы подписаны АО «Минудобрения»;  handed — отдано грузополучателю в руки;  mailed — выслано грузополучателю;  digitalized — передано грузополучателю через систему электронного документооборота;  cancelled — документы уничтожены. |  |  |  |  |  |
| signature | список | Список подписей документа | \* | + |  | \* |  |
| status | строка | Статус заявки:  new — создана новая заявка;  ready — готова к отгрузке;  process — в процессе отгрузки;  completed — отгрузка выполнена полностью;  closed — заявка закрыта;  mistake — ошибочная отгрузка. | \* | + |  | \* |  |

Структура объектов списка 'signature':

| **Поле** | **Тип** | **Описание** | **GET запрос** | **GET ответ** | **POST** | **PATCH** | **DELETE** |
| --- | --- | --- | --- | --- | --- | --- | --- |
| category | строка | Категория подписанта (accountant, salesManager, …) | - | + |  |  |  |
| time | строка | Время подписания | - | \* |  |  |  |
| employeeId | строка | Идентификатор физлица подписанта | - | \* |  |  |  |

Запрос должен возвращать полный список требуемых подписантов с указанием их категории (поле 'category'). Наличие или отсутствие времени подписания (поле 'time') является признаком наличия или отсутствия соответствующей подписи.

|  |  |
| --- | --- |
| Ресурс | **/microservice/v1/waybill** |
| Тип | оповещение |
| Микросервис | **start2000** |
| Описание | оповещение об изменении статуса отгрузки |

Структура объекта:

| **Поле** | **Тип** | **Описание** | **GET запрос** | **GET ответ** | **POST** | **PATCH** | **DELETE** |
| --- | --- | --- | --- | --- | --- | --- | --- |
| invoice | строка | Номер накладной ТОРГ-12 |  |  | + |  |  |
| invoiceDate | строка | Дата накладной ТОРГ-12 |  |  | + |  |  |
| passId | строка | Идентификатор пропуска |  |  | + |  |  |
| weighingId | число | Идентификатор взвешивания |  |  | + |  |  |
| weighingTime | строка | Дата и время взвешивания с грузом |  |  | \* |  |  |
| weightEmpty | число | Вес тары |  |  | \* |  |  |
| weightBrutto | число | Вес брутто |  |  | \* |  |  |
| weightNetto | число | Вес нетто |  |  | \* |  |  |
| carRegNum | строка | Государственный регистрационный номер транспортного средства |  |  | + |  |  |
| carModel | строка | Марка и модель транспортного средства |  |  | + |  |  |
| driverName | строка | ФИО водителя |  |  | + |  |  |
| driverLicense | строка | Номер водительского удостоверения (буквы кириллицей, между серией и номером один пробел) |  |  | + |  |  |

|  |  |
| --- | --- |
| Ресурс | **/microservice/v1/invoicePrint** |
| Тип | запрос |
| Микросервис | **start2000** |
| Описание | печатная форма сопроводительных документов |

Структура объекта:

| **Поле** | **Тип** | **Описание** | **GET запрос** | **GET ответ** | **POST** | **PATCH** | **DELETE** |
| --- | --- | --- | --- | --- | --- | --- | --- |
| invoice | строка | Номер накладной ТОРГ-12 | \* | + |  |  |  |
| invoiceDate | строка | Дата накладной ТОРГ-12 | \* | + |  |  |  |
| contractor | строка | Краткое наименование покупателя | \* | + |  |  |  |
| contractorTIN | строка | ИНН покупателя | \* | + |  |  |  |
| carrier | строка | Краткое наименование перевозчика | \* | + |  |  |  |
| carrierTIN | строка | ИНН перевозчика | \* | + |  |  |  |
| loadType | строка | Идентификатор груза | \* | + |  |  |  |
| weight | число | Масса по накладной | \* | + |  |  |  |
| weightEmpty | число | Вес тары | \* | + |  |  |  |
| weightBrutto | число | Вес брутто | \* | + |  |  |  |
| weightNetto | число | Вес нетто | \* | + |  |  |  |
| document | список | Список документов для печати | - | + |  |  |  |

Структура объектов списка 'document':

| **Поле** | **Тип** | **Описание** | **GET запрос** | **GET ответ** | **POST** | **PATCH** | **DELETE** |
| --- | --- | --- | --- | --- | --- | --- | --- |
| name | строка | Краткое наименование документа | - | + |  |  |  |
| encoding | строка | Способ кодирования:  plain — простой текст;  base64 — кодирование Base64. | - | + |  |  |  |
| mimeType | строка | Тип MIME:  application/rtf — формат Rich Text Format. | - | + |  |  |  |
| data | строка | Содержимое документа в указанной кодировке | - | + |  |  |  |

|  |  |
| --- | --- |
| Ресурс | **/microservice/v1/waybillPrint** |
| Тип | запрос |
| Микросервис | **start2000** |
| Описание | печатная форма сопроводительных документов |

Структура объекта:

| **Поле** | **Тип** | **Описание** | **GET запрос** | **GET ответ** | **POST** | **PATCH** | **DELETE** |
| --- | --- | --- | --- | --- | --- | --- | --- |
| invoice | строка | Номер накладной ТОРГ-12 | \* | + |  |  |  |
| invoiceDate | строка | Дата накладной ТОРГ-12 | \* | + |  |  |  |
| contractor | строка | Краткое наименование покупателя | \* | + |  |  |  |
| contractorTIN | строка | ИНН покупателя | \* | + |  |  |  |
| carrier | строка | Краткое наименование перевозчика | \* | + |  |  |  |
| carrierTIN | строка | ИНН перевозчика | \* | + |  |  |  |
| loadType | строка | Идентификатор груза | \* | + |  |  |  |
| weight | число | Масса по накладной | \* | + |  |  |  |
| weightEmpty | число | Вес тары | \* | + |  |  |  |
| weightBrutto | число | Вес брутто | \* | + |  |  |  |
| weightNetto | число | Вес нетто | \* | + |  |  |  |
| carRegNum | строка | Государственный регистрационный номер транспортного средства | \* | + |  |  |  |
| carModel | строка | Марка и модель транспортного средства | \* | + |  |  |  |
| document | список | Список документов для печати | - | + |  |  |  |

Структура объектов списка 'document':

| **Поле** | **Тип** | **Описание** | **GET запрос** | **GET ответ** | **POST** | **PATCH** | **DELETE** |
| --- | --- | --- | --- | --- | --- | --- | --- |
| name | строка | Краткое наименование документа | - | + |  |  |  |
| encoding | строка | Способ кодирования:  plain — простой текст;  base64 — кодирование Base64. | - | + |  |  |  |
| mimeType | строка | Тип MIME:  application/rtf — формат Rich Text Format. | - | + |  |  |  |
| data | строка | Содержимое документа в указанной кодировке | - | + |  |  |  |

|  |  |
| --- | --- |
| Ресурс | **/microservice/v1/scchClient** |
| Тип | запрос |
| Микросервис | **start2000** |
| Описание | сведения об организациях-клиентах спорткомплекса «Химик» |

Структура объекта:

| **Поле** | **Тип** | **Описание** | **GET запрос** | **GET ответ** | **POST** | **PATCH** | **DELETE** |
| --- | --- | --- | --- | --- | --- | --- | --- |
| organization | строка | Краткое наименование покупателя | \* | + |  |  |  |
| organizationTIN | строка | ИНН покупателя | \* | + |  |  |  |
| balance | число | Сумма остатка предоплаты услуг спорткомплекса на момент запроса по данным СТАРТ2000 | - | + |  |  |  |
| mTime | строка | Время последнего изменения суммы остатка в СТАРТ2000 | - | + |  |  |  |

|  |  |
| --- | --- |
| Ресурс | **/microservice/v1/cacheFlow** |
| Тип | оповещение |
| Микросервис | **start2000** |
| Описание | уведомление о закрытии кассы |

Структура объекта:

| **Поле** | **Тип** | **Описание** | **GET запрос** | **GET ответ** | **POST** | **PATCH** | **DELETE** |
| --- | --- | --- | --- | --- | --- | --- | --- |
| registerCode | строка | Код кассовой точки |  |  | + |  |  |
| registerName | строка | Наименование кассовой точки |  |  | + |  |  |
| registerId | строка | Идентификатор фискального регистратора |  |  | + |  |  |
| time | строка | Время закрытия смены |  |  | + |  |  |
| begin | число | Остаток на начало смены |  |  | + |  |  |
| debit | число | Сумма расхода |  |  | + |  |  |
| credit | число | Сумма дохода |  |  | + |  |  |
| balance | число | Остаток на конец смены |  |  | + |  |  |

|  |  |
| --- | --- |
| Ресурс | **/microservice/v1/scchAccess** |
| Тип | оповещение |
| Микросервис | **scch** |
| Описание | оповещение о проходе через точку контроля в спорткомплексе «Химик» |

Структура объекта:

| **Поле** | **Тип** | **Описание** | **GET запрос** | **GET ответ** | **POST** | **PATCH** | **DELETE** |
| --- | --- | --- | --- | --- | --- | --- | --- |
| passId | строка | Идентификатор пропуска |  |  | + |  |  |
| scchZone | строка | Зона доступа по абонементу:  poolMale — малый бассейн, мужская раздевалка  poolFemale — малый бассейн, женская раздевалка  common — общая зона |  |  | + |  |  |
| direction | строка | Направление прохода (in, out) |  |  | + |  |  |
| time | строка | Время прохода |  |  | + |  |  |

|  |  |
| --- | --- |
| Ресурс | **/microservice/v1/scchTicket** |
| Тип | запрос |
| Микросервис | **scch** |
| Описание | отчёт о реализации абонементов спорткомплекса «Химик» |

Структура объекта:

| **Поле** | **Тип** | **Описание** | **GET запрос** | **GET ответ** | **POST** | **PATCH** | **DELETE** |
| --- | --- | --- | --- | --- | --- | --- | --- |
| dateFrom | строка | Дата первого дня периода отчёта | + | + |  |  |  |
| dateTo | строка | Дата последнего дня периода отчёта | + | + |  |  |  |
| detail | список | Список наименований полей, по которым будет осуществляться детализация отчёта | \* | + |  |  |  |
| report | список | Список строк отчёта (см. ниже) | - | + |  |  |  |
| match | объект | Правила для выборки данных в отчёт | \* | \* |  |  |  |
| filter | объект | Правила для исключения данных из выборки по правилам 'match' | \* | \* |  |  |  |

Поле report содержит список объектов следующего типа (перечень полей в ответе сокращается при их отсутствии в списке detail запроса):

| **Поле** | **Тип** | **Описание** | **GET запрос** | **GET ответ** | **POST** | **PATCH** | **DELETE** |
| --- | --- | --- | --- | --- | --- | --- | --- |
| section | строка | Краткое наименование секции | \* | \* |  |  |  |
| sectionDescription | строка | Полное наименование секции | \* | \* |  |  |  |
| coachId | строка | Идентификатор физлица тренера | \* | \* |  |  |  |
| coachName | строка | ФИО тренера | \* | \* |  |  |  |
| value | число | Количество занятий в абонементе | \* | \* |  |  |  |
| category | строка | Категория абонемента | \* | \* |  |  |  |
| quantity | число | Количество абонементов | \* | \* |  |  |  |
| price | число | Цена абонемента (с НДС) | \* | \* |  |  |  |
| total | число | Сумма без НДС | - | + |  |  |  |
| VAT | число | Сумма НДС | - | + |  |  |  |
| totalVAT | число | Сумма с НДС | - | + |  |  |  |
| VATrate | число | Процент ставки НДС | \* | \* |  |  |  |
| clientName | строка | ФИО клиента | \* | \* |  |  |  |
| clientId | строка | Идентификатор физлица клиента/сотрудника | \* | \* |  |  |  |
| payer | строка | Краткое наименование юрлица плательщика | \* | \* |  |  |  |
| payerTIN | строка | ИНН юрлица плательщика | \* | \* |  |  |  |

Поля 'match' и 'filter' содержат произвольное число полей из объекта 'report' с соответствующими значениями. В отчёт включаются только те данные, которые соответствуют каждому требованию объекта 'match', и исключаются те, которые соответствуют каждому требованию объекта 'filter'. Объектами 'match' и 'filter' поддерживаются шаблоны и диапазоны.

|  |  |
| --- | --- |
| Ресурс | **/microservice/v1/scchHistory** |
| Тип | запрос |
| Микросервис | **scch** |
| Описание | отчёт о посещениях спорткомплекса «Химик» |

Структура объекта:

| **Поле** | **Тип** | **Описание** | **GET запрос** | **GET ответ** | **POST** | **PATCH** | **DELETE** |
| --- | --- | --- | --- | --- | --- | --- | --- |
| dateFrom | строка | Дата первого дня периода отчёта | + | + |  |  |  |
| dateTo | строка | Дата последнего дня периода отчёта | + | + |  |  |  |
| detail | список | Список наименований полей, по которым будет осуществляться детализация отчёта | \* | + |  |  |  |
| report | список | Список строк отчёта (см. ниже) | - | + |  |  |  |
| match | объект | Правила для выборки данных в отчёт | \* | \* |  |  |  |
| filter | объект | Правила для исключения данных из выборки по правилам 'match' | \* | \* |  |  |  |

Поле report содержит список объектов следующего типа (перечень полей в ответе сокращается при их отсутствии в списке detail запроса):

| **Поле** | **Тип** | **Описание** | **GET запрос** | **GET ответ** | **POST** | **PATCH** | **DELETE** |
| --- | --- | --- | --- | --- | --- | --- | --- |
| section | строка | Краткое наименование секции | \* | \* |  |  |  |
| sectionDescription | строка | Полное наименование секции | \* | \* |  |  |  |
| coachId | строка | Идентификатор физлица тренера | \* | \* |  |  |  |
| coachName | строка | ФИО тренера | \* | \* |  |  |  |
| value | число | Количество занятий в абонементе | \* | \* |  |  |  |
| category | строка | Категория абонемента | \* | \* |  |  |  |
| used | число | Количество посещений по абонементу | \* | \* |  |  |  |
| date | строка | Дата посещения | \* | \* |  |  |  |
| duration | число | Продолжительность посещения в минутах | \* | \* |  |  |  |
| clientName | строка | ФИО клиента | \* | \* |  |  |  |
| clientId | строка | Идентификатор физлица клиента/сотрудника | \* | \* |  |  |  |
| payer | строка | Краткое наименование юрлица плательщика | \* | \* |  |  |  |
| payerTIN | строка | ИНН юрлица плательщика | \* | \* |  |  |  |

Поля 'match' и 'filter' содержат произвольное число полей из объекта 'report' с соответствующими значениями. В отчёт включаются только те данные, которые соответствуют каждому требованию объекта 'match', и исключаются те, которые соответствуют каждому требованию объекта 'filter'. Объектами 'match' и 'filter' поддерживаются шаблоны и диапазоны.

|  |  |
| --- | --- |
| Ресурс | **/microservice/v1/mealTicket** |
| Тип | запрос |
| Микросервис | **mealticket** |
| Описание | данные о талонах на лечебно-профилактическое питание |

Структура объекта:

| **Поле** | **Тип** | **Описание** | **GET запрос** | **GET ответ** | **POST** | **PATCH** | **DELETE** |
| --- | --- | --- | --- | --- | --- | --- | --- |
| ticketId | число | Номер талона | \* | + |  | + | + |
| employeeId | строка | Идентификатор физлица (сотрудника), получившего талон | \* | + |  | \* | - |
| type | строка | Тип питания по талону:  milk — молоко;  standard — стандартное ЛПП;  enchanced — усиленное ЛПП | \* | + |  | - | - |
| state | строка | Состояние талона:  new — новый;  expired — просроченный;  used — использованный;  illegal — неправомерно использованный;  deleted — аннулированный | \* | + |  | \* | - |
| startWork | строка | Дата и время начала смены, за которую начислен талон | \* | + |  | - | - |
| expired | строка | Дата и время окончания действия талона | \* | + |  | \* | - |
| created | строка | Дата и время создания талона | \* | + |  | - | - |
| creatorLogin | строка | Учётная запись создателя талона («СКУД» для автоматически созданных) | \* | + |  | - | - |
| creatorName | строка | ФИО создателя | \* | \* |  | - | - |
| edited | строка | Дата и время последнего редактирования | \* | \* |  | - | - |
| editorLogin | строка | Учётная запись редактора | \* | \* |  | \* | - |
| editorName | строка | ФИО редактора | \* | \* |  | \* | - |
| reason | строка | Словесное описание причины последних изменений | \* | \* |  | \* | - |
| used | строка | Дата и время использования талона | \* | \* |  | \* | - |
| department | строка | Наименование корневого подразделения сотрудника | \* | \* |  | - | - |
| depCode | строка | Код корневого подразделения сотрудника | \* | \* |  | - | - |

В запросе допускается использовать наборы (списки) значений.

|  |  |
| --- | --- |
| Ресурс | **/microservice/v1/order** |
| Тип | запрос |
| Микросервис | **canteen** |
| Описание | отчёт по реализациям из торговой системы |

Структура объекта:

| **Поле** | **Тип** | **Описание** | **GET запрос** | **GET ответ** | **POST** | **PATCH** | **DELETE** |
| --- | --- | --- | --- | --- | --- | --- | --- |
| dateFrom | строка | Дата первого дня периода отчёта | + | + |  |  |  |
| dateTo | строка | Дата последнего дня периода отчёта | + | + |  |  |  |
| detail | список | Список наименований полей, по которым будет осуществляться детализация отчёта | \* | + |  |  |  |
| report | список | Список строк отчёта (см. ниже) | - | + |  |  |  |
| match | объект | Правила для выборки данных в отчёт | \* | \* |  |  |  |
| filter | объект | Правила для исключения данных из выборки по правилам 'match' | \* | \* |  |  |  |

Поле report содержит список объектов следующего типа (перечень полей в ответе сокращается при их отсутствии в списке detail запроса):

| **Поле** | **Тип** | **Описание** | **GET запрос** | **GET ответ** | **POST** | **PATCH** | **DELETE** |
| --- | --- | --- | --- | --- | --- | --- | --- |
| number | строка | Номер заказа (чека) | \* | \* |  |  |  |
| date | строка | Дата покупки | \* | \* |  |  |  |
| client | строка | ФИО клиента (для сотрудников) | \* | \* |  |  |  |
| clientId | строка | Идентификатор физлица (для сотрудников) | \* | \* |  |  |  |
| place | строка | Наименование места продажи | \* | \* |  |  |  |
| typeOfPayment | строка | Тип оплаты («Талон», «Наличная», «Безналичная») | \* | \* |  |  |  |
| product | строка | Наименование товара | \* | \* |  |  |  |
| quantity | число | Количество товара по строке чека | \* | \* |  |  |  |
| cost | число | Цена единицы товара по строке чека | \* | \* |  |  |  |
| summ | число | Сумма по строке чека | \* | \* |  |  |  |
| bitNDS | строка | Ставка НДС | \* | \* |  |  |  |
| summNDS | число | Сумма НДС по строке чека | \* | \* |  |  |  |
| total | число | Итого (с учётом запрошенной детализации) | \* | \* |  |  |  |

|  |  |
| --- | --- |
| Ресурс | **/microservice/v1/luncheon** |
| Тип | запрос |
| Микросервис | **catering** |
| Описание | накопительные суммы долга по сотрудникам на запрошенную дату |

Структура объекта:

| **Поле** | **Тип** | **Описание** | **GET запрос** | **GET ответ** | **POST** | **PATCH** | **DELETE** |
| --- | --- | --- | --- | --- | --- | --- | --- |
| date | строка | Дата, на конец календарных суток которой посчитан долг | + | + |  |  |  |
| employeeId | строка | Идентификатор физлица сотрудника предприятия | \* | + |  |  |  |
| sum | число | Сумма накопленного долга на конец суток запрошенной даты | - | + |  |  |  |

При запросе допускается указание в 'employeeId' списка значений. В этом случае в ответе возвращается список объектов по одному для каждого найденного 'employeeId'. При отсутствии 'employeeId' возвращаются данные по всем 'employeeId', имеющим ненулевой долг на запрошенную дату.

|  |  |
| --- | --- |
| Ресурс | **/microservice/v1/deduction** |
| Тип | оповещение |
| Микросервис | **catering** |
| Описание | удержания из з/п за питание |

Структура объекта:

| **Поле** | **Тип** | **Описание** | **GET запрос** | **GET ответ** | **POST** | **PATCH** | **DELETE** |
| --- | --- | --- | --- | --- | --- | --- | --- |
| dateTime | строка | Дата и время учёта удержания в кадровой системе |  |  | + |  |  |
| employeeId | строка | Идентификатор физлица сотрудника предприятия |  |  | + |  |  |
| sum | число | Сумма удержания за питание, проведённая в кадровой системе |  |  | + |  |  |

Оповещение об удержании может быть повторено (например, при потере ответа). Необходимо дедуплицировать удержания по дате/времени и сотруднику. В случае, когда для той же пары время/сотрудник приходит новое значение суммы удержания, необходимо корректировать расчёт накопительного долга.

|  |  |
| --- | --- |
| Ресурс | **/microservice/v1/closeLuncheon** |
| Тип | запрос |
| Микросервис | **catering** |
| Описание | требование немедленного прекращения обслуживания сотрудника торговой системой |

Структура объекта:

| **Поле** | **Тип** | **Описание** | **GET запрос** | **GET ответ** | **POST** | **PATCH** | **DELETE** |
| --- | --- | --- | --- | --- | --- | --- | --- |
| employeeId | строка | Идентификатор физлица сотрудника предприятия | + | + |  |  |  |
| status | строка | Код исполнения требования по конкретному сотруднику:  done — требование исполнено;  absent — обслуживание и не велось. | - | + |  |  |  |

При запросе допускается указание в 'employeeId' списка значений. В этом случае в ответе возвращается список объектов по одному для каждого найденного 'employeeId'.

Приложение Б

**Правила работы с ресурсами**

|  |  |
| --- | --- |
| Ресурс | **/microservice/v1/pass** |

Назначение абонемента на пропуск сотрудника АО «Минудобрения» (или дочерней организации)

1. Запросить данные физлица по его пропуску. Метод GET. Обязательные поля в запросе: 'passId'. Поле 'create' отсутствует.

{

"show": [ "passId", "passDate", "passEnd", "humanId", "name", "docType", "docNum", "department", "passType", "scchClient" ],

"request": {

"passId": "0357844630"

},

"count": 1,

"first": 0,

"last": 0,

"total": 1,

"response": [

{

"passId": "0357844630",

"passDate": "2017-06-01",

"passEnd": "9999-12-31",

"humanId": "17450",

"name": "Иванов Иван Иванович",

"docType": "passport",

"docNum": "2012 123456",

"department": "Профком",

"passType": "employee",

"scchClient": 0

}

]

}

Необходимо убедиться, что:

* время действия пропуска полностью перекрывает время действия абонемента;
* тип пропуска — employee (или filial);
* другого абонемента к пропуску не привязано - 'scchClient': 0.

1. Разрешить по пропуску сотрудника проходить в спорткомплекс как клиенту. Метод PATCH. Актуально только для пропусков с 'passType': 'employee' и 'passType': 'filial'. Обязательные поля 'passId', 'scchClient'. Инициатору scch запрещено менять любые поля кроме 'scchClient', 'scchZone', 'scchSchedule'.

{

"show": [ "passId", "scchClient", "scchSchedule", "scchZone" ],

"request": {

"passId": "0357844630",

"passType": "employee",

"scchClient": 1,

"scchZone": "common"

},

"count": 1,

"first": 0,

"last": 0,

"total": 1,

"response": [

{

"passId": "0357844630",

"scchClient": 1,

"scchSchedule": "Спорткомплекс: Клиент",

"scchZone": "common"

}

]

}

Если в запросе отсутствует 'scchZone', при привязке абонемента назначается 'common'.

Если в запросе отсутствует 'scchSchedule', при привязке абонемента назначается 'Спорткомплекс: Клиент'. Все наименования расписаний спорткомплекса должны иметь префикс 'Спорткомплекс: '. Если в запросе префикс отсутствует, он добавляется микросервисом СКУД. Если расписание с таким наименованием не найдено в базе СКУД, возвращается ошибка ('status': 'error').

Отмена назначения абонемента на пропуск

1. Определить идентификатор пропуска по сохранённому идентификатору физлица. Метод GET. Обязательные поля 'humanId'. Поле 'create' отсутствует.

{

"show": [ "passId", "humanId", "scchClient", "passType" ],

"request": {

"humanId": "17450",

"scchClient": 1

},

"count": 2,

"first": 0,

"last": 1,

"total": 2,

"response": [

{

"passId": "0357844630",

"humanId": "17450",

"scchClient": 1,

"passType": "employee"

},

{

"passId": "9445947234",

"humanId": "17450",

"scchClient": 1,

"passType": "scchTicket"

}

]

}

1. Выбрать подходящий пропуск по 'passType': 'employee' (или 'passType': 'filial').
2. Запретить по пропуску сотрудника проходить в спорткомплекс как клиенту. Метод PATCH. Актуально только для пропусков с 'passType': 'employee' или 'passType': 'filial'. Обязательные поля 'passId', 'scchClient'. Инициатору scch запрещено менять любые поля кроме 'scchClient', 'scchZone', 'scchSchedule'. Вне зависимости от переданных значений 'scchZone' и 'scchSchedule' при отмене назначения абонемента данные параметры устанавливаются в СКУД пустыми. Вызов осуществляется аналогично назначению.

Выдача QR-кода сотруднику

1. Запросить создание отдельного пропуска в спорткомплекс сотруднику. Метод GET. Актуально только для запросов с 'passType': 'scchTicket' и 'scchClient': 1. Обязательные поля 'create', 'passEnd', 'passType', 'humanId', 'scchClient'. Поле 'passId' отсутствует.

{

"show": [ "passId", "create", "humanId", "passType", "scchClient", "scchSchedule", "scchZone", "qrCode" ],

"request": {

"humanId": "17450",

"passEnd": "2018-03-31",

"passType": "scchTicket",

"scchClient": 1,

"create": 1

},

"count": 1,

"first": 0,

"last": 0,

"total": 1,

"response": [

{

"passId": "9445947234",

"create": 1,

"humanId": "17450",

"passType": "scchTicket",

"scchClient": 1,

"scchSchedule": "Спорткомплекс: Клиент",

"scchZone": "common",

"qrCode": "<?xml version=\"1.0\" encoding=\"UTF-8\" standalone=\"yes\"?>

<!-- Created with acs microservice -->

<svg width=\"4.0cm\" height=\"4.0cm\" viewBox=\"0 0 29 29\" preserveAspectRatio=\"xMidYMid meet\" version=\"1.1\" xmlns=\"http://www.w3.org/2000/svg\">

<g id=\"QRcode\">

<rect x=\"0\" y=\"0\" width=\"29\" height=\"29\" fill=\"#ffffff\" />

<g id=\"Pattern\">

<rect x=\"4\" y=\"4\" width=\"1\" height=\"1\" fill=\"#000000\" />

<!-- ... ... ещё много-много строк с прямоугольниками ... ... -->

<rect x=\"24\" y=\"24\" width=\"1\" height=\"1\" fill=\"#000000\" />

</g>

</g>

</svg>"

}

]

}

В случае, если подходящий под критерии пропуск найден среди действующих, в ответе возвращается 'create': 0.

Выдача абонемента с QR-кодом стороннему клиенту

1. Запросить создание пропуска в спорткомплекс стороннему клиенту. Метод GET. Актуально только для запросов с 'passType': 'scchTicket' и 'scchClient': 1. Обязательные поля 'create', 'passEnd', 'passType', 'scchClient', 'name', 'docType', 'docNum'. Поля 'humanId' и 'passId' отсутствуют.

{

"show": [ "passId", "create", "humanId", "name", "docType", "docNum", "passType", "scchClient", "scchSchedule", "scchZone", "qrCode" ],

"request": {

"name": "Иванов Иван Иванович",

"docType": "passport",

"docNum": "2012 123456",

"passEnd": "2018-03-31",

"passType": "scchTicket",

"scchClient": 1,

"create": 1

},

"count": 1,

"first": 0,

"last": 0,

"total": 1,

"response": [

{

"passId": "9445947234",

"create": 1,

"humanId": "17450",

"name": "Иванов Иван Иванович",

"docType": "passport",

"docNum": "2012 123456",

"passType": "scchTicket",

"scchClient": 1,

"scchSchedule": "Спорткомплекс: Клиент",

"scchZone": "common",

"qrCode": "<?xml version=\"1.0\" encoding=\"UTF-8\" standalone=\"yes\"?>

<!-- Created with acs microservice -->

<svg width=\"4.0cm\" height=\"4.0cm\" viewBox=\"0 0 29 29\" preserveAspectRatio=\"xMidYMid meet\" version=\"1.1\" xmlns=\"http://www.w3.org/2000/svg\">

<g id=\"QRcode\">

<rect x=\"0\" y=\"0\" width=\"29\" height=\"29\" fill=\"#ffffff\" />

<g id=\"Pattern\">

<rect x=\"4\" y=\"4\" width=\"1\" height=\"1\" fill=\"#000000\" />

<!-- ... ... ещё много-много строк с прямоугольниками ... ... -->

<rect x=\"24\" y=\"24\" width=\"1\" height=\"1\" fill=\"#000000\" />

</g>

</g>

</svg>"

}

]

}

В случае, если подходящий под критерии пропуск найден среди действующих, в ответе возвращается 'create': 0.

Перевыпуск QR-кода

1. Определить идентификатор пропуска по сохранённому идентификатору физлица. Метод GET. Обязательные поля 'humanId'. Поле 'create' отсутствует.
2. Запросить перевыпуск нового QR-кода в спорткомплекс. Метод GET. Актуально только для запросов с 'passType': 'scchTicket' и 'scchClient': 1. Обязательные поля 'create', 'passId', 'passType', 'scchClient'. Другие поля в запросе не допускаются.

{

"show": [ "passId", "create", "humanId", "passType", "scchClient", "scchSchedule", "scchZone", "qrCode" ],

"request": {

"passId": "17450",

"passType": "scchTicket",

"scchClient": 1,

"create": 2

},

"count": 1,

"first": 0,

"last": 0,

"total": 1,

"response": [

{

"passId": "9445947234",

"create": 2,

"humanId": "17450",

"passType": "scchTicket",

"scchClient": 1,

"scchSchedule": "Спорткомплекс: Клиент",

"scchZone": "common",

"qrCode": "<?xml version=\"1.0\" encoding=\"UTF-8\" standalone=\"yes\"?>

<!-- Created with acs microservice -->

<svg width=\"4.0cm\" height=\"4.0cm\" viewBox=\"0 0 29 29\" preserveAspectRatio=\"xMidYMid meet\" version=\"1.1\" xmlns=\"http://www.w3.org/2000/svg\">

<g id=\"QRcode\">

<rect x=\"0\" y=\"0\" width=\"29\" height=\"29\" fill=\"#ffffff\" />

<g id=\"Pattern\">

<rect x=\"4\" y=\"4\" width=\"1\" height=\"1\" fill=\"#000000\" />

<!-- ... ... ещё много-много строк с прямоугольниками ... ... -->

<rect x=\"24\" y=\"24\" width=\"1\" height=\"1\" fill=\"#000000\" />

</g>

</g>

</svg>"

}

]

}

Корректировка абонемента с QR-кодом

1. Определить идентификатор пропуска по сохранённому идентификатору физлица. Метод GET. Обязательные поля 'humanId'. Поле 'create' отсутствует.
2. Выбрать пропуск с 'passType': 'scchTicket'.
3. Скорректировать данные пропуска. Метод PATCH. Актуально только для пропусков с 'passType': 'scchTicket'. Обязательные поля 'passId', 'passType', 'scchClient'.

{

"show": [ "passId", "scchClient", "passEnd" ],

"request": {

"passId": "0357844630",

"scchClient": 1,

"passType": "scchTicket",

"passEnd": "2018-03-31"

},

"count": 1,

"first": 0,

"last": 0,

"total": 1,

"response": [

{

"passId": "0357844630",

"scchClient": 1,

"passEnd": "2018-03-31"

}

]

}

Удаление абонемента с QR-кодом

1. Определить идентификатор пропуска по сохранённому идентификатору физлица. Метод GET. Обязательные поля 'humanId'. Поле 'create' отсутствует.
2. Выбрать пропуск с 'passType': 'scchTicket'.
3. Удалить пропуск. Метод DELETE. Актуально только для пропусков с 'passType': 'scchTicket'.

{

"request": {

"passId": "0357844630",

"passType": "scchTicket"

},

"count": 1,

"first": 0,

"last": 0,

"total": 1,

"response": [

{

"passId": "0357844630"

}

]

}

Дерево возможных обращений к /microservice/v1/pass от микросервиса scch:

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| GET | | | | | | | | | | | | | | |
| create отсутствует ИЛИ (create!=1 И create!=2) | create=1 | | | | | | | | | create=2 | | | | |
| поиск | passType=scchTicket И scchClient=1 | | | | | ИНАЧЕ ошибка | | | passType=scchTicket И scchClient=1 | | | | | ИНАЧЕ ошибка |
|  | humanId присутствует И passId отсутствует | | ИНАЧЕ ошибка | |  | | | passId присутствует | | | | ИНАЧЕ ошибка |  | |
|  | подходящий passId не найден | подходящий passId найден | |  | | |  | | | подходящий passId найден | подходящий passId не найден |  |  | |
|  | создание QR | возобновление абонемента | |  | | |  | | | смена QR | ошибка |  |  | |

|  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| PATCH | | | | | | | DELETE | | | |
| passType=employee ИЛИ passType=filial | | | passType=scchTicket И scchClient=1 | | ИНАЧЕ ошибка | | passType=scchTicket | | ИНАЧЕ ошибка | |
| scchClient=1 | scchClient=0 | ИНАЧЕ ошибка | | корректировка | |  | | удаление | |  |
| определение scchZone, scchSchedule | очистка scchZone, scchSchedule |  | |  | |  | |  | |  |
| назначение | снятие назначения |  | |  | |  | |  | |  |

|  |  |
| --- | --- |
| Ресурс | **/microservice/v1/timeInterval** |

Пример обращения:

{

"request": {

"employeeId": [ "0123456789", "987654321", "123%" ],

"timeFrom": "2017-06-28T00:00:00",

"timeTo": "2017-06-28T23:59:59"

},

"count": 2,

"first": 0,

"last": 1,

"total": 2,

"response": [

{

"employeeId": "987654321",

"timeFrom": "2017-06-28T00:00:00",

"timeTo": "2017-06-28T23:59:59",

"interval": [

{ "date": "2017-06-28", "from": "07:00", "to": "08:00" },

{ "date": "2017-06-28", "from": "12:00", "to": "13:00" },

{ "date": "2017-06-28", "from": "17:00", "to": "18:00" }

]

},

{

"employeeId": "123123123123",

"timeFrom": "2017-06-28T00:00:00",

"timeTo": "2017-06-28T23:59:59",

"interval": [

{ "date": "2017-06-28", "from": "07:00", "to": "18:00" }

]

}

]

}