

Описание API. Release 11

- 1 Общие положения
 - 1.1 Принципов взаимодействия
 - 1.2 Аутентификация
- 2 API аутентификации
 - 2.1 API аутентификации
 - 2.1.1 Метод получения случайных данных для подписания
 - 2.1.2 Метод получения аутентификационного токена
- 3 Заказы кодов маркировки
 - 3.1 Метод создания заказа в ИС МП (Метод устарел)
 - 3.2 Асинхронный метод создания заказа в ИС МП
 - 3.3 Асинхронный метод создания черновика заказа в ИС МП
 - 3.4 Асинхронный метод обновления черновика заказа в ИС МП
 - 3.5 Асинхронный метод создания заказа из черновика в ИС МП
 - 3.6 Метод получения статуса заказа в ИС МП
 - 3.7 Метод получения списка заказов
 - 3.8 Метод получения списка шаблонов этикеток участника
- 4 Коды маркировки и товары
 - 4.1 Метод получения списка КМ
 - 4.2 Метод получения информации о КМ
 - 4.3 Метод получения списка товаров
 - 4.4 Получение информации о маркированном товаре
 - 4.5 Метод получения списка КИЗов по идентификатору
- 5 Документы
 - 5.1 Метод получения списка документов, ранее загруженных в ИС МП
 - 5.2 Метод получения документа, ранее загруженного в ИС МП
 - 5.3 Методы создания документов
 - 5.4 Метод создания черновика документа
 - 5.5 Метод получения черновика документа
 - 5.6 Метод обновления черновика документа
 - 5.7 Метод удаления черновика документа
 - 5.8 Метод поиска и фильтрации черновика документа
 - 5.9 Метод получения списка КИЗов по идентификатору
 - 5.10 Добавление заявки на регистрацию УОТ
- 6 Регистрация участника
 - 6.1 Добавление заявки на регистрацию УОТ
 - 6.2 Добавление нового пользователя в реестр ИС МП
 - 6.3 Добавление сертификатов пользователям
 - 6.4 Блокировка пользователя
 - 6.5 Возобновление пользователя
 - 6.6 Удаление пользователя
 - 6.7 Списание нанесенных КМ
 - 6.8 Списание не нанесенных КМ
- 7 Справочник товаров GS1
 - 7.1 Метод получения списка gtin участника оборота товара по ИНН
- 8 Справочник кодов ТН ВЭД
 - 8.1 Метод получения списка 10-ти значных кодов ТН ВЭД
- 9 Реестр СУЗ (станция управления заказами)
 - 9.1 Метод для добавления СУЗ (пользовательский)
 - 9.2 Метод для удаления СУЗ (пользовательский)
 - 9.3 Метод для получения текущего активного СУЗ (пользовательский)
- 10 Реестр регистраторов эмиссии
 - 10.1 Метод получения списка регистраторов эмиссии
- 11 Шаблоны этикеток
 - 11.1 Метод получения списка шаблонов
- 12 Справочник товаров GS1
 - 12.1 Метод получения списка gtin участника оборота товара по ИНН
 - 12.2 Метод получения доверенности
 - 12.3 Метод поиска по доверенностям
 - 12.4 Метод получения PDF файла доверенности
- 13 Приложение. Документ заказа КМ
 - 13.1 Формат JSON
 - 13.2 Формат XML
 - 13.3 Формат CSV
- 14 Ролевая модель
 - 14.1 Добавление подписанной заявки на добавление роли пользователя
 - 14.2 Добавление подписанной заявки на изменение роли пользователя
 - 14.3 Добавление подписанной заявки на удаление роли пользователя
 - 14.4 Добавление подписанной заявки на присвоению роли пользователю
 - 14.5 Получение списка ролей участника

- 14.6 Регистрация доменных объектов
 - 14.7 Обновление доменных объектов
- 15 Приложение. Документ по созданию роли
 - 15.1 Формат JSON
 - 15.2 Формат XML
- 16 Приложение. Документ по изменению роли
 - 16.1 Формат JSON
 - 16.2 Формат XML
- 17 Приложение. Документ по удалению роли
 - 17.1 Формат JSON
 - 17.2 Формат XML
- 18 Приложение. Документ по присвоению роли
 - 18.1 Формат JSON
 - 18.2 Формат XML
- 19 Приложение. Документ с информацией о СУЗ
 - 19.1 Формат JSON
- 20 Приложение. Документ о вводе в оборот
 - 20.1 Формат JSON
 - 20.2 Формат XML
 - 20.3 Формат CSV
- 21 Приложение. Документ о вводе в оборот (Импорт)
 - 21.1 Формат JSON
 - 21.2 Формат XML
 - 21.3 Формат CSV
- 22 Приложение. Отгрузка
 - 22.1 Формат JSON
 - 22.2 Формат XML
 - 22.3 Формат CSV
- 23 Приложение. Документ об отмене отгрузки товара
 - 23.1 Формат JSON
- 24 Приложение. Документ о приемке товара
 - 24.1 Формат JSON
 - 24.2 Формат XML
- 25 Приложение. Документ по регистрации УОТ
 - 25.1 Формат JSON:
 - 25.2 Формат XML
- 26 Приложение. Документ по блокировке/возобновлению/удалению пользователей
 - 26.1 Формат JSON:
 - 26.2 Формат XML
- 27 Приложение. Документ вывода товара из оборота при реализации конечному потребителю
 - 27.1 Формат JSON
 - 27.2 Формат XML
 - 27.3 Формат CSV
- 28 Приложение. Документ вывода товара из оборота при реализации договора рассрочки
 - 28.1 Формат JSON
 - 28.2 Формат XML
 - 28.3 Формат CSV
- 29 Приложение. Документ ввода в оборот, произведенного на территории РФ (производство сторонней организацией)
 - 29.1 Формат JSON
 - 29.2 Формат XML
 - 29.3 Формат CSV
- 30 Приложение. Документ ввода в оборот товара, полученного от физических лиц, не являющихся индивидуальными предпринимателями, в том числе при комиссионной торговле
 - 30.1 Формат JSON
 - 30.2 Формат XML
- 31 Приложение. Документ списания КМ с/без СИ
 - 31.1 Формат JSON
 - 31.2 Формат XML
 - 31.3 Формат CSV
- 32 Приложение. Документ перемаркировки товара
 - 32.1 Формат JSON
 - 32.2 Формат XML
 - 32.3 Формат CSV
- 33 Примеры
 - 33.1 Аутентификация по УКЭП
 - 33.1.1 Пример получения токена при авторизации 1C
 - 33.1.2 Пример получения токена при авторизации REACT JS.
 - 33.1.3 Пример получения токена при авторизации Java.

Общие положения

Принципов взаимодействия

Взаимодействие участников эксперимента и оператора ИС МП осуществляется в электронном виде. Обмен документами на бумажном носителе не предусмотрен. Электронное взаимодействие осуществляется по одному из двух вариантов:

1. Предоставление сведений. Обмен электронными документами, на основании которых могут вноситься изменения в реестры ИС МП. Представление сведений определено как обязанность участников.
2. Запрос сведений. Обмен, при котором сведения из реестров ИС МП предоставляются участникам в соответствии с установленными разрешениями на получение сведений; изменения в реестры не вносятся. Запрос и последующее представление сведений из ИС МП инициируется участником по своему усмотрению, и применяется, например, для проверки статусов кодов маркировки или статуса регистрации участника.

Электронное взаимодействие осуществляется путем обмена электронными документами в соответствии с определенным порядком. Для юридически значимых электронных документов применяется усиленная квалифицированная электронная подпись (УКЭП). Электронное взаимодействие осуществляется с применением программных интерфейсов взаимодействия (API) ИС МП и информационных систем участников.

Взаимодействие строится по модели "клиент-сервер", где ИС МП является сервером, а программные средства участника – клиентом.

При взаимодействии через API выполняются следующие требования:

1. Аутентификация и авторизация участника (определение полномочий на осуществление электронного взаимодействия) осуществляется при помощи УКЭП.
2. При предоставлении и запросе сведений защита выполняется TLS/HTTPS

(ГОСТ 28147-89) шифрованием канала передачи данных в адрес Оператора сертифицированными РФ СКЗИ (ГОСТ Р 34.10-2012).

Адрес промышленной версии ИС МП: ismp.crpt.ru, адрес демонстрационной версии ИС МП: demo.fashion.crpt.ru.

В ответах REST API ИС МП могут содержаться дополнительные служебные json атрибуты, не указанные в документации. Неизвестные атрибуты следует игнорировать.

Для обмена с ИС МП используется json стандарта RFC 7159 (<https://tools.ietf.org/html/rfc7159>)

Аутентификация

Все методы взаимодействия с ИС МП требуют установленного заголовка: Authorization: Bearer <token>, где token - аутентификационный токен.

Для получения аутентификационного токена участника оборота товара в ИС МП реализовано API аутентификации и регистрации по УКЭП (см. п 2).

API аутентификации

API аутентификации

Для получения аутентификационного токена необходимо запросить в ИС МП массив случайных данных (GET /api/v3/auth/cert/key), подписать их УКЭП и отправить в ИС МП для проверки (POST /api/v3/auth/cert/). В случае успешной проверки подписи, ИС МП в ответ вернет аутентификационный токен формата jwt (см. jwt.io)

Метод получения случайных данных для подписания

GET /api/v3/auth/cert/key

Заголовки ответа:

Content-Type: application/json

Тело ответа:

```
{
  "uuid": "<GUID> ",
  "data": "<RANDOM STRING> "
}
```

Атрибутивный состав ответа:

uuid - уникальный идентификатор сгенерированных случайных данных, тип string

data - случайная строка, тип string

http статус ответа в случае успеха:

200 - ОК

Пример запроса:

```
curl -v https://ismp.crpt.ru/api/v3/auth/cert/key
:
< HTTP/1.1 200
< Server: nginx/1.14.0
< Date: Fri, 25 Jan 2019 15:49:47 GMT
< Content-Type: application/json;charset=UTF-8
< Transfer-Encoding: chunked
< Connection: keep-alive
<
* Connection #0 to host ismp.crpt.ru left intact
{"uuid": "b7b1abc9-f4ee-47db-8a20-f80ac83504e8", "data": "QNRPNPFGJZFUXCERQ
MTWLRMBRNRAAP" }
```

Метод получения аутентификационного токена

POST /api/v3/auth/cert/

Заголовок запроса:

Content-Type: application/json

Тело запроса:

```
{
  "uuid": "<GUID>",
  "data": "<  base64>"
}
```

Атрибутивный состав ответа:

uuid - уникальный идентификатор подписанных случайных данных (uuid из метода GET /api/v3/auth/cert/key), тип string

data - подписанные УКЭП зарегистрированного участника случайные данные в base64,

Заголовки ответа:

Content-Type: application/json

Тело ответа:

```
{
  "token": "<JWT token>"
}
```

Атрибутивный состав ответа:

token - аутентификационный токен в формате JWT (<https://jwt.io/>).

JWT токен:

Тело ответа:

```
{
  "user_status": <STRING>, : "ACTIVE",
  "full_name": <STRING>, : "  ",
  "scope": < >, [ "trusted" ],
  "inn": <STRING>, : "6820022500",
  "id": <NUMBER>, : 50,
  "exp": <NUMBER>, : 1548525191,
  "authorities": < >, : [ "INN_6820022500",
"ROLE_ORG_TRADE_PARTICIPANT" ],
  "jti": <STRING>, : "496c046e-2226-490c-93d8-333e9928a1dc",
  "client_id": "crpt-service"
}
```

Атрибутивный состав JWT токена:

user_status - текущий статус пользователя в системе ИС МП, тип строковый

full_name - ФИО пользователя, тип строковый

scope - служебный атрибут ИС МП, тип массив

inn - ИНН участника, тип строковый

id - регистрационный номер участника в ИС МП, тип числовой

exp - время окончания срока действия, формат Unix time, тип числовой

authorities - служебный атрибут ИС МП, тип массив

jti - служебный атрибут ИС МП, тип строка

client_id служебный атрибут ИС МП, тип строка

http статус ответа в случае успеха:

200 - ОК

Пример запроса:

```
curl -X POST -v 'https://ismp.crpt.ru/api/v3/auth/cert/' -H
'content-type: application/json;charset=UTF-8' --data-binary
'{"uuid": "50b47eeb-3dbc-46a8-beld-dac17449aaa4", "data": "< base64>"
:
< HTTP/2 200
< server: nginx/1.14.1
< date: Sat, 26 Jan 2019 07:46:11 GMT
< content-type: application/json;charset=UTF-8
< vary: Origin
< vary: Access-Control-Request-Method
< vary: Access-Control-Request-Headers
< access-control-allow-origin: https://ismp.crpt.ru
< access-control-allow-credentials: true
<
* Connection #0 to host ismp.crpt.ru left intact
{"token": "eyJhbGciOiJIUzI1NiIsInR5cCI6IkpXVCJ9.eyJ1c2VyX3N0YXR1cyI6IkkFDV
ElWRSIsImZlbGxhbmFtZSI6Ii50b47eeb-3dbc-46a8-beld-dac17449aaa4", "data": "
NC80LjRgNC-0LLQvdCwIiwidXNlcnI6ImV4Ym91bnQ6IiwiaWF0IjoiMj01OTUyMjYyOTUy
mlubiI6IjY4MjAwMjI1MDAiLCJpZCI6NTAsImV4cCI6MTU0ODUyNDc3MSwiYXV0aG9yaXRpZ
XMiOlsiSU5OXzY4MjAwMjI1MDAiLCJST0xFOX09SR19UUkFERV9QQVJUSUNJUEFOVCJdLCJqd
GkiOiJmODI0ZWU3ZC1lMzh1LTRkMDAtYmY2ZS1hMmE4NmJjYjYmZmZmZmZmZmZmZmZmZm
iJjcnB0LXNlcnZpY2UifQ.mWffwihDEaYFzstRFIFq4Ni2wYN6fMY7VfCKJbbCjwU" }
```

Заказы кодов маркировки

Метод создания заказа в ИС МФТ (Метод устарел)

POST /api/v3/facade/order/create

Заголовок запроса:

Content-Type: application/json

Тело запроса:

```
{
  "document": "< base64>",
  "document_format": "< >"
  "signature": "< base64>"
}
```

Атрибутивный состав ответа:

document - документ заказа КМ, закодированный в base64. Описание документа приведено в п. "Приложение. Документ заказа КМ". Тип строковый

document_format - формат документа заказа КМ, возможные значения "MANUAL" - json, "CSV" - csv, "XML" - xml. Тип строковый.

signature - открепленная УКЭП закодированный в base64. Документ необходимо подписывать до перевода его в base64. Тип строковый.

Заголовки ответа:

Content-Type: application/json

Тело ответа:

```
{
  "emission_registrar_order_id": <GUID>,
  "order_status": "READY_TO_PROCESS"
}
```

Атрибутивный состав ответа:

emission_registrar_order_id - уникальный номер заказа КМ в ИС МП, тип строковый

order_status - статус заказа КМ в ИС МП

http статус ответа в случае успеха:

201 - Created

http статус ответа, если документ не прошел валидацию:

400 - Bad Request

Пример запроса:

```
curl 'https://ismp.crpt.ru/api/v3/facade/order/create' -H 'content-type:
application/json' -H 'Authorization: Bearer <>'
--data-binary '{ "document": "< base64>",
"document_format": "MANUAL", "signature": "< base64>" }'
:
< HTTP/1.1 201
< Server: nginx/1.14.0
< Date: Sun, 27 Jan 2019 09:41:23 GMT
< Content-Type: application/json; charset=UTF-8
< Transfer-Encoding: chunked
< Connection: keep-alive
< X-Content-Type-Options: nosniff
< X-XSS-Protection: 1; mode=block
< Cache-Control: no-cache, no-store, max-age=0, must-revalidate
< Pragma: no-cache
< Expires: 0
< X-Frame-Options: DENY
<
* Connection #0 to host ismp.crpt.ru left intact
{"emission_registrar_order_id": "a82e4f78-3dc5-43a0-9a36-4c8c2ec8eb20", "o
rder_status": "READY_TO_PROCESS"}
```

Асинхронный метод создания заказа в ИС МП

POST /api/v3/facade/order

Заголовок запроса:

Content-Type: application/json

Тело запроса:

```
{
  "document": "< base64>",
  "document_format": "< >",
  "signature": "< base64>"
}
```

Атрибутивный состав запроса:

document - документ заказа КМ, закодированный в base64. Описание документа приведено в п. "Приложение. Документ заказа КМ". Тип строковый

document_format - формат документа заказа КМ, возможные значения "MANUAL" - json, "CSV" - csv, "XML" - xml. Тип строковый.

signature - открепленная УКЭП закодированный в base64. Документ необходимо подписывать до перевода его в base64. Тип строковый.

Заголовки ответа:

Content-Type: application/json

Тело ответа:

```
{
  "order_id": <GUID>
}
```

Атрибутивный состав ответа:

order_id - уникальный номер заказа КМ в ИС МП, тип строковый в формате GUID

http статус ответа в случае успеха:

202 - Accepted

http статус ответа в случае ошибки:

400, 422 - при ошибках в атрибутивном составе тела запроса

401 - Требуется авторизация

500 - произошла внутренняя ошибка сервера

Пример запроса:


```
curl -X POST 'https://ismp.crpt.ru/api/v3/facade/order' -H
'content-type: application/json' -H 'Authorization: Bearer <>'
--data-binary '{ "document": "< base64>",
"document_format": "MANUAL", "signature": "< base64>" }'
:
< HTTP/1.1 202
< Server: nginx/1.14.0
< Date: Sun, 27 Jan 2019 09:41:23 GMT
< Content-Type: application/json; charset=UTF-8
< Transfer-Encoding: chunked
< Connection: keep-alive
< X-Content-Type-Options: nosniff
< X-XSS-Protection: 1; mode=block
< Cache-Control: no-cache, no-store, max-age=0, must-revalidate
< Pragma: no-cache
< Expires: 0
< X-Frame-Options: DENY
<
* Connection #0 to host ismp.crpt.ru left intact
{"order_id": "a82e4f78-3dc5-43a0-9a36-4c8c2ec8eb20" }
```

Асинхронный метод создания черновика заказа в ИС МП

POST /api/v3/facade/order/draft

Заголовок запроса:

Content-Type: application/json

Тело запроса:

```
{
  "document": "< base64>",
  "document_format": "< >"
}
```

Атрибутивный состав запроса:

document - документ заказа КМ, закодированный в base64. Описание документа приведено в п. "Приложение. Документ заказа КМ". Тип строковый

document_format - формат документа заказа КМ, возможные значения "MANUAL" - json, "CSV" - csv, "XML" - xml. Тип строковый.

Заголовки ответа:

Content-Type: application/json

Тело ответа:

```
{
  "order_id": <GUID>
}
```

Атрибутивный состав ответа:

order_id - уникальный номер заказа КМ в ИС МП, тип строковый в формате GUID

http статус ответа в случае успеха:

202 - Accepted

http статус ответа в случае ошибки:

400 - при ошибках в атрибутивном составе тела запроса

401 - Требуется авторизация

500 - произошла внутренняя ошибка сервера

Пример запроса:

```
curl -X POST 'https://ismp.crpt.ru/api/v3/facade/order' -H
'content-type: application/json' -H 'Authorization: Bearer <>'
--data-binary '{ "document": "< base64>", "document_format": "MANUAL" }'
:
< HTTP/1.1 202
< Server: nginx/1.14.0
< Date: Sun, 27 Jan 2019 09:41:23 GMT
< Content-Type: application/json; charset=UTF-8
< Transfer-Encoding: chunked
< Connection: keep-alive
< X-Content-Type-Options: nosniff
< X-XSS-Protection: 1; mode=block
< Cache-Control: no-cache, no-store, max-age=0, must-revalidate
< Pragma: no-cache
< Expires: 0
< X-Frame-Options: DENY
<
* Connection #0 to host ismp.crpt.ru left intact
{"order_id": "a82e4f78-3dc5-43a0-9a36-4c8c2ec8eb20" }
```

Асинхронный метод обновления черновика заказа в ИС МП

PATCH /api/v3/facade/order/{id черновика заказа}

Параметры url:

{id черновика заказа} - уникальный номер черновика в ИС МП

Заголовок запроса:

Content-Type: application/json

Тело запроса:

```
{
  "document": "< base64>",
  "document_format": "< >"
}
```

Атрибутивный состав запроса:

document - документ заказа КМ, закодированный в base64. Описание документа приведено в п. "Приложение. Документ заказа КМ". Тип строковый

document_format - формат документа заказа КМ, возможные значения "MANUAL" - json, "CSV" - csv, "XML" - xml. Тип строковый.

Заголовки ответа:

Content-Type: application/json

Тело ответа:

```
{
  "order_id": <GUID>
}
```

Атрибутивный состав ответа:

order_id - уникальный номер заказа КМ в ИС МП, тип строковый в формате GUID

http статус ответа в случае успеха:

202 - Accepted

http статус ответа в случае ошибки:

400 - при ошибках в атрибутивном составе тела запроса

401 - Требуется авторизация

500 - произошла внутренняя ошибка сервера

Пример запроса:

```
curl -X PATCH 'https://ismp.crpt.ru/api/v3/facade/order' -H
'content-type: application/json' -H 'Authorization: Bearer <>'
--data-binary '{ "document": "< base64>", "document_format": "MANUAL"}'
:
< HTTP/1.1 202
< Server: nginx/1.14.0
< Date: Sun, 27 Jan 2019 09:41:23 GMT
< Content-Type: application/json; charset=UTF-8
< Transfer-Encoding: chunked
< Connection: keep-alive
< X-Content-Type-Options: nosniff
< X-XSS-Protection: 1; mode=block
< Cache-Control: no-cache, no-store, max-age=0, must-revalidate
< Pragma: no-cache
< Expires: 0
< X-Frame-Options: DENY
<
* Connection #0 to host ismp.crpt.ru left intact
{"order_id": "a82e4f78-3dc5-43a0-9a36-4c8c2ec8eb20" }
```

Асинхронный метод создания заказа из черновика в ИС МП

POST /api/v3/facade/order/{id черновика заказа}

Заголовок запроса:

Content-Type: application/json

Тело запроса:

```
{
  "document": "< base64>",
  "document_format": "< >",
  "signature": "< base64>"
}
```

Атрибутивный состав запроса:

document - документ заказа КМ, закодированный в base64. Описание документа приведено в п. "Приложение. Документ заказа КМ". Тип строковый

document_format - формат документа заказа КМ, возможные значения "MANUAL" - json, "CSV" - csv, "XML" - xml. Тип строковый.

signature - открепленная УКЭП закодированный в base64. Документ необходимо подписывать до перевода его в base64. Тип строковый.

Заголовки ответа:

Content-Type: application/json

Тело ответа:

```
{
  "order_id": <GUID>
}
```

Атрибутивный состав ответа:

order_id - уникальный номер заказа КМ в ИС МП, тип строковый в формате GUID

http статус ответа в случае успеха:

202 - Accepted

http статус ответа в случае ошибки:

400, 422 - при ошибках в атрибутивном составе тела запроса

401 - Требуется авторизация

500 - произошла внутренняя ошибка сервера

Пример запроса:

```
curl -X POST 'https://ismp.crpt.ru/api/v3/facade/order' -H
'content-type: application/json' -H 'Authorization: Bearer <>'
--data-binary '{ "document": "< base64>",
"document_format": "MANUAL", "signature": "< base64>" }'
:
< HTTP/1.1 202
< Server: nginx/1.14.0
< Date: Sun, 27 Jan 2019 09:41:23 GMT
< Content-Type: application/json; charset=UTF-8
< Transfer-Encoding: chunked
< Connection: keep-alive
< X-Content-Type-Options: nosniff
< X-XSS-Protection: 1; mode=block
< Cache-Control: no-cache, no-store, max-age=0, must-revalidate
< Pragma: no-cache
< Expires: 0
< X-Frame-Options: DENY
<
* Connection #0 to host ismp.crpt.ru left intact
{"order_id": "a82e4f78-3dc5-43a0-9a36-4c8c2ec8eb20" }
```

Метод получения статуса заказа в ИС МП

GET /api/v3/facade/order/{UUID заказа}/details

Параметры URL:

<UUID заказа> - уникальный номер заказа в ИС МП

Заголовки ответа:

Content-Type: application/json

Тело ответа:

```
{
  "order_id": <GUID>,    "a82e4f78-3dc5-43a0-9a36-4c8c2ec8eb20",
  "order_status": <    >, "PROCESSING",
  "order_creation_date": <    >,    1548582083721,
  "document_information": {
    "contract_number": <    >
    "contract_date": <    >,    1546387200000,
    "production_type": <    >,    "SELF_MADE",
    "contact_person": <    >
    "label_template_id": <GUID    >,
    "products_information": [
      {
        "gtin": <GTIN >,
        "tn_ved_code": <4-x    >,
        "emission_type": <    >,
        "code_quantity": <    >,
        "code_type": <    >,
        "serial_num_type": <    >,
        "marking_type": <    >,
        "order_line_status": <    >
      }
    ]
  }
}
```

Атрибутивный состав ответа:

order_id - уникальный номер заказа в ИС МП. Тип строковый.

order_status - статус заказа КМ. Тип строковый.

order_creation_date - дата создания заказа. Формат Unix time в миллисекундах. Тип числовой.

document_information - основная информация о заказе КМ. Тип объект json.

contract_number - номер договора с оператором ИС МП. Тип строковый.

contract_date - дата договора с оператором ИС МП. Формат Unix time в миллисекундах. Тип числовой.

production_type - способ изготовления СИ. Возможные значения "SELF_MADE" - самостоятельно, СЕМ - центр этикетирования и маркировки. Тип строковый.

contact_person - контактное лицо. Тип строковый.

label_template_id - уникальный номер шаблона этикетке в ИС МП. Тип строковый.

products_information - информация о товарах заказа. Тип массив объектов json.

gtin - 14-ти значный GS1 код товара Если код товара меньше 14 символов, то его необходимо дополнить ведущими нулями. Тип строковый.

tn_ved_code - 4-х код ТН ВЭД. Тип строковый.

emission_type - способ выпуска товара. Возможные значения "LOCAL" - производство в РФ, "FOREIGN" - импорт. Тип строковый.

code_quantity - количество КМ в подзаказе. Тип числовой.

code_type - вид СИ. Возможные значения "PRINTABLE" - печатаемый, "GLUEABLE" - клеевой, "MOUNTABLE" - навесной. Тип строковый.

serial_num_type - способ формирования серийного номера. Возможные значения "OPERATOR" - генерируется оператором, "SELF_MADE" - самостоятельно (не рекомендованно к использованию). Тип строковый.

marking_type - вид маркировки. Возможные значения "PRODUCT_PACKAGE" - на потребительскую упаковку, PRODUCT - на товар, PRODUCT_LABEL - на товарный ярлык. Тип строковый.

order_line_status - статус подзаказа. Тип строковый.

http статус ответа в случае успеха:

200 - ОК

Пример запроса:

```
curl
'https://ismp.crpt.ru/api/v3/facade/order/a82e4f78-3dc5-43a0-9a36-4c8c2e
c8eb20/details' -H 'Authorization: Bearer <>'
:
< HTTP/1.1 200
< Server: nginx/1.14.0
< Date: Sun, 27 Jan 2019 16:10:33 GMT
< Content-Type: application/json;charset=UTF-8
< Transfer-Encoding: chunked
< Connection: keep-alive
< X-Content-Type-Options: nosniff
< X-XSS-Protection: 1; mode=block
< Cache-Control: no-cache, no-store, max-age=0, must-revalidate
< Pragma: no-cache
< Expires: 0
< X-Frame-Options: DENY
<

* Connection #0 to host ismp.crpt.ru left intact
{
  "order_id": "a82e4f78-3dc5-43a0-9a36-4c8c2ec8eb20",
  "order_status": "PRODUCTION",
  "order_creation_date": 1548582083721,
  "document_information": {
    "contract_number": "11",
    "contract_date": 1546387200000,
    "production_type": "SELF_MADE",
    "contact_person": "1",
    "label_template_id": "2dbae9e2-e5a2-492e-a987-edda887def58",
    "products_information": [
      {
        "order_line_id": "9a2cb3d6-eac8-45c7-9d02-5b82dc7ca0ef",
        "gtin": "04650117240996",
        "tn_ved_code": "6401",
        "emission_type": "LOCAL",
        "code_quantity": "12",
        "code_type": "PRINTABLE",
        "serial_num_type": "OPERATOR",
        "marking_type": "PRODUCT_PACKAGE",
        "order_line_status": "PRODUCTION"
      }
    ]
  }
}
```


Метод получения списка заказов

GET /api/v3/facade/order/all

Параметры запроса:

direction - порядок сортировки. Возможные значения "DESC" - по убыванию, "ASC" - по возрастанию. Опциональный параметр.

sort - код поля сортировки. Опциональный параметр.

limit - максимальное количество заказов в КМ в ответе. Опциональный параметр.

offset - номер страницы ответа, начальное значение 0. Опциональный параметр.

Заголовки ответа:

Content-Type: application/json

Тело ответа:

```
{
  "subOrders": [
    {
      "order_id": <GUID >, "a82e4f78-3dc5-43a0-9a36-4c8c2ec8eb20",
      "order_creation_date": < >, 1548582083721,
      "gtin_total_quantity": < gtin>,
      "code_total_quantity": < >,
      "delivery_type": < >, "PHYSICAL_DEVICE",
      "delivery_address": < >, "",
      "order_status": < >, "PRODUCTION",
      "production_type": < >, "SELF_MADE",
      "participant_inn": < >, "6820022500",
      "participant_name": < >
    }
  ]
}
```

Атрибутивный состав ответа:

subOrders - массив заказов текущего участника. Тип массив объектов

order_id - уникальный номер заказа в ИС МП. Тип строковый.

order_creation_date - дата создания заказа. Формат Unix time в миллисекундах. Тип числовой.

gtin_total_quantity - суммарное количество gtin по всем подзаказам. Тип числовой.

code_total_quantity - суммарное количество кодов маркировки по всем подзаказам. Тип числовой.

delivery_type - способ получения. Тип числовой.

delivery_address - адрес доставки. Тип числовой.

order_status - статус заказа КМ. Тип строковый.

production_type - способ изготовления СИ. Возможные значения "SELF_MADE" - самостоятельно, СЕМ - центр этикетирования и маркировки. Тип строковый.

participant_inn - ИНН участника. Тип строковый.

participant_name - Наименование организации участника. Тип строковый.

http статус ответа в случае успеха:

200 - ОК

Пример запроса:

```
curl 'https://ismp.crpt.ru/api/v3/api/v3/facade/order/all' -H
'Authorization: Bearer <>'
:
< HTTP/1.1 200
< Server: nginx/1.14.0
< Date: Sun, 27 Jan 2019 16:10:33 GMT
< Content-Type: application/json;charset=UTF-8
< Transfer-Encoding: chunked
< Connection: keep-alive
< X-Content-Type-Options: nosniff
< X-XSS-Protection: 1; mode=block
< Cache-Control: no-cache, no-store, max-age=0, must-revalidate
< Pragma: no-cache
< Expires: 0
< X-Frame-Options: DENY
<

* Connection #0 to host ismp.crpt.ru left intact
{"subOrders":[{"order_id":"a82e4f78-3dc5-43a0-9a36-4c8c2ec8eb20","order_
creation_date":1548582083721,"gtin_total_quantity":1,"code_total_quantit
y":12,"delivery_type":null,"delivery_address":null,"order_status":"PRODU
CTION","production_type":"SELF_MADE","participant_inn":"6820022500","par
ticipant_name":
"\\""}, {"order_id":"b6515cb9-ea90-47e1-a13f-b840c8d3c004","order_creatio
n_date":1548582063227,"gtin_total_quantity":1,"code_total_quantity":12,"
delivery_type":null,"delivery_address":null,"order_status":"PRODUCTION",
"production_type":"SELF_MADE","participant_inn":"6820022500","participan
t_name":
"\\""}, {"order_id":"9c570e70-da86-4195-b641-1bcf434e8338","order_creatio
n_date":1548582054265,"gtin_total_quantity":1,"code_total_quantity":12,"
delivery_type":null,"delivery_address":null,"order_status":"PRODUCTION",
"production_type":"SELF_MADE","participant_inn":"6820022500","participan
t_name":
"\\""}, {"order_id":"7534becf-f5a6-443e-a4de-9a5bc41dfd57","order_creatio
n_date":1548581669255,"gtin_total_quantity":1,"code_total_quantity":12,"
delivery_type":null,"delivery_address":null,"order_status":"PRODUCTION",
"production_type":"SELF_MADE","participant_inn":"6820022500","participan
t_name":
"\\""}, {"order_id":"d3927ac6-3426-486e-b730-19a10c5af1b2","order_creatio
n_date":1548578431094,"gtin_total_quantity":1,"code_total_quantity":11,"
delivery_type":null,"delivery_address":null,"order_status":"PRODUCTION",
"production_type":"SELF_MADE","participant_inn":"6820022500","participan
t_name":
"\\""}, {"order_id":"0bf6a8ad-7acc-4320-815c-a2046af3f8c3","order_creatio
n_date":1548252241232,"gtin_total_quantity":3,"code_total_quantity":21,"
delivery_type":null,"delivery_address":",
-", "order_status":"MK_PARTLY_EMITTED","production_type":"SELF_MADE","par
ticipant_inn":"6820022500","participant_name":
"
```

```
\\""}, {"order_id": "32323abd-00a9-4cc8-9039-30e50515ed85", "order_creatio
n_date": 1548250810947, "gtin_total_quantity": 161, "code_total_quantity": 17
3134, "delivery_type": "PHYSICAL_DEVICE", "delivery_address": "", "order_stat
us": "DRAFT", "production_type": "SELF_MADE", "participant_inn": "6820022500"
, "participant_name": ""
\\""}, {"order_id": "11597101-fb5b-4789-85a6-c3487d0ac132", "order_creatio
n_date": 1548250707462, "gtin_total_quantity": 137, "code_total_quantity": 14
8582, "delivery_type": "PHYSICAL_DEVICE", "delivery_address": "", "order_stat
us": "PROCESSING", "production_type": "SELF_MADE", "participant_inn": "682002
2500", "participant_name": ""
\\""}, {"order_id": "f808c46e-61fe-4819-afec-ed0cd1221557", "order_creatio
n_date": 1547640455639, "gtin_total_quantity": 150, "code_total_quantity": 37
2, "delivery_type": "PHYSICAL_DEVICE", "delivery_address": "", "order_status"
: "DRAFT", "production_type": "SELF_MADE", "participant_inn": "6820022500", "p
articipant_name": ""
\\""}, {"order_id": "b8cc3d5f-cd91-42ed-82ee-8b4d205241c6", "order_creatio
n_date": 1547545337678, "gtin_total_quantity": 150, "code_total_quantity": 37
2, "delivery_type": "PHYSICAL_DEVICE", "delivery_address": "", "order_status"
: "DRAFT", "production_type": "SELF_MADE", "participant_inn": "6820022500", "p
articipant_name": ""
\\""}], "total": 19}
```

Метод получения списка шаблонов этикеток участника

GET /api/v3/facade/label/template/all

Коды маркировки и товары

Метод получения списка КМ

GET /api/v3/facade/identifytools/listV2

Параметры запроса:

Название параметра	Тип данных	Описание
Фильтрация		
cis	Строка	Код маркировки (допускается как полное совпадение, так и частичное)
emissionDateFrom	Строка	Дата эмиссии, от. Задается в формате уууу-ММ-дд'T'HH:mm:ss.SSS'Z Пример: 2019-01-01T03:00:00.000Z
emissionDateTo	Строка	Дата эмиссии, до. Задается в формате уууу-ММ-дд'T'HH:mm:ss.SSS'Z Пример: 2019-01-10T03:00:00.000Z
gtin	Строка	Global Trade Item Number
sn	Строка	Серийный номер
tnVed10	Строка	10-значный Код ТН ВЭД ЕАЭС
producerInn	Строка	ИНН производителя
cisStatus	Массив строковых перечислений	Текущий статус КМа. Список доступных значений: EMITTED - эмитирован INTRODUCED - введен в оборот RETIRED - выбыл
emissionType	Строковое перечисление	Тип эмиссии. Список доступных значений: LOCAL - производство РФ FOREIGN - ввезен в РФ
Пагинация/Сортировка		
order	Строковое перечисление	Тип сортировки (ascending/descending). Список доступных значений: ASC - ascending DESC - descending
uit	Строка	Значение КМ-"точки отсчета", по которому сортируются записи.
orderedColumnValue	Строка	Значение столбца-"точки отсчета", по которому сортируются записи.
orderColumn	Строковое перечисление	Название столбца, по которому будет производиться сортировка. Список доступных значений: emd - дата эмиссии

pageDir	Строковое перечисление	Выбор направления (вперед/назад) для пагинации. Список доступных значений: NEXT - вперед PREV - назад
limit	Целое число	Максимальное количество записей, которое вернется в качестве ответа.

http статус ответа в случае успеха:

200 - ОК

Тело ответа:

```
{
  "results": [
    {
      "cis": <>, "010405915570452921wAwsi5ysFW27f2406405",
      "gtin": <GTIN>, "04059155704529",
      "tnVedEaes": <10- >, "6401990000",
      "tnVedEaesGroup": <4- >, "6405",
      "productName": < >, " , . 9-9-24302-21-019/6",
      "emissionDate": < >, "2019-01-23T08:24:21.048Z",
      "ownerInn": < >, "583713715202",
      "ownerName": < >, " ",
      "type": < >, "PRINTED",
      "markingType": < >, "ON_PACKAGE",
      "status": < >, "INTRODUCED",
      "emissionType": , "LOCAL"
    }
  ],
  "total": 68
}
```

Атрибутивный состав ответа:

results - список кодов маркировки

gtin - gtin товара

tnVedEaes - 10-значный Код ТН ВЭД ЕАЭС

tnVedEaesGroup - группа ТН ВЭД ЕАЭС

productName - наименование продукции

emissionDate - дата эмиссии

ownerInn - ИНН собственника товара

ownerName - наименование собственника товара

type - вид СИ (PRINTED, GLUED, HINGED)

markingType - вид маркировки (ON_PACKAGE, ON_PRODUCT, ON_PRODUCT_LABEL)

status - статус КМ (EMITTED - эмитирован, INTRODUCED - введен в оборот, RETIRED - выбыл)

emissionType - тип производства (LOCAL - производство РФ, FOREIGN - ввезен в РФ).

Пример запроса:

```

curl -X GET
'https://ismp.crpt.ru/api/v3/facade/identifytools/listV2?limit=10&order=
DESC&orderColumn=emd&uit=010469022978096621qSF7qjA3aRIqM2406402&orderedC
olumnValue=2019-01-23T08:42:17.593Z&pageDir=NEXT' -H 'Authorization:
Bearer <>'
:
< HTTP/1.1 200
< Server: nginx/1.14.0
< Date: Sun, 27 Jan 2019 09:41:23 GMT
< Content-Type: application/json;charset=UTF-8
< Transfer-Encoding: chunked
< Connection: keep-alive
< X-Content-Type-Options: nosniff
< X-XSS-Protection: 1; mode=block
< Cache-Control: no-cache, no-store, max-age=0, must-revalidate
< Pragma: no-cache
< Expires: 0
< X-Frame-Options: DENY
<
* Connection #0 to host ismp.crpt.ru left intact
{
  "results": [
    {
      "cis": "010405915570452921wAwwsi5ysFW27f2406405",
      "gtin": "04059155704529",
      "tnVedEaes": "6401990000",
      "tnVedEaesGroup": "6405",
      "productName": " , . 9-9-24302-21-019/6",
      "emissionDate": "2019-01-23T08:24:21.048Z",
      "ownerInn": "583713715202",
      "ownerName": " ",
      "participantName": " ",
      "participantInn": "583713715202",
      "type": "PRINTED",
      "markingType": "ON_PACKAGE",
      "status": "INTRODUCED",
      "lastDocId": "a55eb7db-1e38-42ae-8bca-f58b84807ec7",
      "emissionType": "LOCAL"
    }
  ]
}

```

Данный запрос вернет 10 (параметр limit) следующих (параметр pageDir) записей, начиная с даты эмиссии 2019-01-23T08:42:17.593Z (параметр orderedColumnValue) и кода маркировки 010469022978096621qSF7qjA3aRIqM2406402 (параметр uit). Сортировка будет производиться по дате эмиссии (параметр orderColumn) и (всегда) по самому коду маркировки (вне зависимости от переданных параметров) DESC'ом (параметр order). Значение даты эмиссии и кода маркировки должно быть взято из последней строки (для pageDir NEXT) и из первой для (pageDir PREV) для перемещения между страницами.

Метод получения информации о КМ

GET /api/v3/facade/identifytools/{cis}

Параметры url:

cis - код маркировки (строковое значение).

http статус ответа в случае успеха:

200 - ОК

Тело ответа:

```
{
  "cis": <>, "010405915570452921wAwsi5ysFW27f2406405",
  "gtin": <GTIN>, "04059155704529",
  "tnVedEaes": <10- >, "6401990000",
  "tnVedEaesGroup": <4- >, "6405",
  "productName": < >, " , . 9-9-24302-21-019/6",
  "emissionDate": < >, "2019-01-23T08:24:21.048Z",
  "ownerInn": < >, "583713715202",
  "ownerName": < >, " ",
  "type": < >, "PRINTED",
  "markingType": < >, "ON_PACKAGE",
  "status": < >, "INTRODUCED",
  "emissionType": , "LOCAL"
}
```

Атрибутивный состав ответа:

gtin - gtin товара

tnVedEaes - 10-значный Код ТН ВЭД ЕАЭС

tnVedEaesGroup - группа ТН ВЭД ЕАЭС

productName - наименование продукции

emissionDate - дата эмиссии

ownerInn - ИНН собственника товара

ownerName - наименование собственника товара

type - вид СИ (PRINTED, GLUED, HINGED)

markingType - вид маркировки (ON_PACKAGE, ON_PRODUCT, ON_PRODUCT_LABEL)

status - статус КМ (EMITTED - эмитирован, INTRODUCED - введен в оборот, RETIRED - выбыл)

emissionType - тип производства (LOCAL - производство РФ, FOREIGN - ввезен в РФ).

Пример запроса:

```

curl -X GET
'https://ismp.crpt.ru/api/v3/facade/identifytools/010405915570452921HXRN
JW15asPkB2406405' -H 'Authorization: Bearer <>'
:
< HTTP/1.1 200
< Server: nginx/1.14.0
< Date: Sun, 27 Jan 2019 09:41:23 GMT
< Content-Type: application/json;charset=UTF-8
< Transfer-Encoding: chunked
< Connection: keep-alive
< X-Content-Type-Options: nosniff
< X-XSS-Protection: 1; mode=block
< Cache-Control: no-cache, no-store, max-age=0, must-revalidate
< Pragma: no-cache
< Expires: 0
< X-Frame-Options: DENY
<
* Connection #0 to host ismp.crpt.ru left intact
  {
    "cis": "010405915570452921wAwsi5ysFW27f2406405",
    "gtin": "04059155704529",
    "tnVedEaes": "6401990000",
    "tnVedEaesGroup": "6405",
    "productName": " , . 9-9-24302-21-019/6",
    "emissionDate": "2019-01-23T08:24:21.048Z",
    "ownerInn": "583713715202",
    "ownerName": " ",
    "participantName": " ",
    "participantInn": "583713715202",
    "type": "PRINTED",
    "markingType": "ON_PACKAGE",
    "status": "INTRODUCED",
    "lastDocId": "a55eb7db-1e38-42ae-8bca-f58b84807ec7",
    "emissionType": "LOCAL"
  }

```

Метод получения списка товаров

GET /api/v3/facade/marked_products/listV2

Параметры запроса:

Название параметра	Тип данных	Описание
Фильтрация		
cis	Строка	Уникальный идентификатор (допускается как полное совпадение, так и частичное)

emissionDateFrom	Строка	Дата эмиссии, от. Задается в формате уууу-ММ-дд'Т'НН:мм:сс.SSS'Z Пример: 2019-01-01T03:00:00.000Z
emissionDateTo	Строка	Дата эмиссии, до. Задается в формате уууу-ММ-дд'Т'НН:мм:сс.SSS'Z Пример: 2019-01-10T03:00:00.000Z
gtin	Строка	Global Trade Item Number
sn	Строка	Серийный номер
tnVed10	Строка	10-значный Код ТН ВЭД ЕАЭС
producerInn	Строка	ИНН производителя
cisStatus	Массив строковых перечислений	Текущий статус товара. Список доступных значений: INTRODUCED - введен в оборот WAIT_SHIPMENT - ожидает отгрузки (приемку) RETIRED - выбыл (розничная реализация) LOAN_RETIRED - выбыл (договор рассрочки)
emissionType	Строковое перечисление	Тип эмиссии. Список доступных значений: LOCAL - производство РФ FOREIGN - ввезен в РФ
Пагинация/Сортировка		
order	Строковое перечисление	Тип сортировки (ascending/descending). Список доступных значений: ASC - ascending DESC - descending
uit	Строка	Значение КМ-"точки отсчета", по которому сортируются записи.
orderedColumnValue	Строка	Значение столбца-"точки отсчета", по которому сортируются записи.
orderColumn	Строковое перечисление	Название столбца, по которому будет производиться сортировка. Список доступных значений: emd - дата эмиссии
pageDir	Строковое перечисление	Выбор направления (вперед/назад) для пагинации. Список доступных значений: NEXT - вперед PREV - назад
limit	Целое число	Максимальное количество записей, которое вернется в качестве ответа.

Тело ответа:

```

{
  "results": [
    {
      "cis": "010463003407002921wskglE44Rlqym2406401",
      "gtin": "04630034070029",
      "tnvd": "6401100000",
      "tnvedGroup": "6401",
      "productName": " , . CRPT--99999, 52",
      "ownerName": " \ " \ ",
      "ownerInn": "7731369928",
      "producerName": " \ " \ ",
      "producerInn": "7731369928",
      "status": "INTRODUCED",
      "emissionDate": "2019-01-18T08:14:40.344Z",
      "lastDocId": "e76a1c45-c8ce-4961-84f1-1df47e5a7d35",
      "name": " , . CRPT--99999, 52",
      "brand": "",
      "model": "CRPT--99999",
      "country": " ",
      "productTypeDesc": "",
      "color": "",
      "materialDown": "",
      "productSize": "52",
      "materialUpper": "",
      "materialLining": "",
      "packageType": "BOX",
      "productType": "310000022"
    }
  ],
  "total": 1
}

```

Атрибутивный состав ответа:

results - список товаров

cis - Уникальный идентификатор товара

gtin - Global Trade Item Number

tnvd - Код ТН ВЭД ЕАЭС товара

tnvedGroup - Группа ТН ВЭД ЕАЭС (первые 4 символа)

productName - Наименование товара

ownerName - Собственник товара

ownerInn - ИНН собственника товара

producerName - Производитель/импортер товара

producerInn - ИНН производителя/импортера товара

status - статус товара: INTRODUCED - введен в оборот WAIT_SHIPMENT - ожидает отгрузки (приемку) RETIRED - выбыл (розничная реализация) LOAN_RETIRED - выбыл (договор рассрочки)

emissionDate - дата производства

emissionType - тип производства (LOCAL - производство РФ, FOREIGN - ввезен в РФ)

lastDocId - Код последнего созданного документа, включая заявку

name - Наименование товара

brand - Товарный бренд

model - Производитель товара (model из gs1)

prevCis - предыдущее значение кода маркировки (в случае, если товар был перемаркирован).

В результирующем JSON также может содержаться набор полей, специфичных для конкретного товара.

Пример запроса:



```
curl -X GET
'https://ismp.crpt.ru/api/v3/facade/identifytools/010405915570452921HXRN
JW15asPkb2406405' -H 'Authorization: Bearer <>'
:
< HTTP/1.1 200
< Server: nginx/1.14.0
< Date: Sun, 27 Jan 2019 09:41:23 GMT
< Content-Type: application/json;charset=UTF-8
< Transfer-Encoding: chunked
< Connection: keep-alive
< X-Content-Type-Options: nosniff
< X-XSS-Protection: 1; mode=block
< Cache-Control: no-cache, no-store, max-age=0, must-revalidate
< Pragma: no-cache
< Expires: 0
< X-Frame-Options: DENY
<
* Connection #0 to host ismp.crpt.ru left intact
{
  "results": [
    {
      "cis": "010463003407002921wskglE44Rlqym2406401",
      "gtin": "04630034070029",
      "tnvd": "6401100000",
      "tnvedGroup": "6401",
      "productName": " , . CRPT--99999, 52",
      "ownerName": " \" \" \"",
      "ownerInn": "7731369928",
      "producerName": " \" \" \"",
      "producerInn": "7731369928",
      "status": "INTRODUCED",
      "emissionDate": "2019-01-18T08:14:40.344Z",
      "lastDocId": "e76alc45-c8ce-4961-84f1-1df47e5a7d35",
      "name": " , . CRPT--99999, 52",
      "brand": "",
      "model": "CRPT--99999",
      "country": " ",
      "productTypeDesc": "",
      "color": "",
      "materialDown": "",
      "productSize": "52",
      "materialUpper": "",
      "materialLining": "",
      "packageType": "BOX",
      "productType": "310000022"
    }
  ],
  "total": 1
}
```

Данный запрос вернет 10 (параметр limit) следующих (параметр pageDir) записей, начиная с даты эмиссии 2019-01-23T08:42:17.593Z (параметр orderedColumnValue) и кода маркировки 010463003407002921wskg1E44R1qum2406401 (параметр uit). Сортировка будет производиться по дате эмиссии (параметр orderColumn) и (всегда) по самому коду маркировки (вне зависимости от переданных параметров) DESC'ом (параметр order). Значение даты эмиссии и кода маркировки должно быть взято из последней строки (для pageDir NEXT) и из первой для (pageDir PREV) для перемещения между страницами.

Получение информации о маркированном товаре

GET /api/v3/facade/marked_products/{uit}

Параметры url:

uit - уникальный идентификатор товара (строковое значение).

http статус ответа в случае успеха:

200 - ОК

Тело ответа:

```

{
  "cis": < >, "010463003407002921wskg1E44R1qym2406401",
  "gtin": <gtin>, "04630034070029",
  "tnvd": <10- >, "6401100000",
  "tnvedGroup": <4- > "6401",
  "productName": < >,
  "ownerName": < >,
  "ownerInn": < >, "7731369928",
  "producerName": < / >,
  "producerInn": < / >, "7731369928",
  "status": <c>, "INTRODUCED",
  "emissionDate": < >, "2019-01-18T08:14:40.344Z",
  "producedDate": < >, "2019-01-18T08:19:33.575Z",
  "type": < >, "PRINTED",
  "emissionType": < >, "LOCAL",
  "markingType": < >, "ON_PACKAGE",
  "prodOrderType": < >, "OPERATOR",
  "name": " , . CRPT--99999, 52",
  "brand": "",
  "model": "CRPT--99999",
  "certDoc": {
    "type": "CONFORMITY_CERT",
    "number": "122",
    "date": "2019-01-02T20:00:00.000Z"
  },
  "country": " ",
  "productTypeDesc": "",
  "color": "",
  "materialDown": "",
  "productSize": "52",
  "materialUpper": "",
  "materialLining": "",
  "packageType": "BOX",
  "productType": "310000022"
}

```

Атрибутивный состав ответа:

cis - Уникальный идентификатор товара

gtin - Global Trade Item Number

tnvd - Код ТН ВЭД ЕАЭС товара

tnvedGroup - Группа ТН ВЭД ЕАЭС (первые 4 символа)

productName - Наименование товара

ownerName - Собственник товара

ownerInn - ИНН собственника товара

producerName - Производитель/импортер товара

producerInn - ИНН производителя/импортера товара

status - статус товара: INTRODUCED - введен в оборот WAIT_SHIPMENT - ожидает отгрузки (приемку) RETIRED - выбыл (розничная реализация) LOAN_RETIRED - выбыл (договор рассрочки)

emissionDate - дата производства

producedDate - Дата ввода товара с СИ в оборот

emissionType - тип производства (LOCAL - производство РФ, FOREIGN - ввезен в РФ)

lastDocId - Код последнего созданного документа, включая заявку

name - Наименование товара

brand - Товарный бренд

model - Производитель товара (model из gs1)

prevCis - предыдущее значение кода маркировки (в случае, если товар был перемаркирован).

prodOrderType - Тип производственного заказа

type - тип КМ (PRINTED, GLUED, HINGED)

Пример запроса:

```
curl -X GET
'https://ismp.crpt.ru/api/v3/facade/marked_products/010463003407002921wskg1E44R1qym2406401' -H 'Authorization: Bearer <>'
:
< HTTP/1.1 200
< Server: nginx/1.14.0
< Date: Sun, 27 Jan 2019 09:41:23 GMT
< Content-Type: application/json;charset=UTF-8
< Transfer-Encoding: chunked
< Connection: keep-alive
< X-Content-Type-Options: nosniff
< X-XSS-Protection: 1; mode=block
< Cache-Control: no-cache, no-store, max-age=0, must-revalidate
< Pragma: no-cache
< Expires: 0
< X-Frame-Options: DENY
<
* Connection #0 to host ismp.crpt.ru left intact
{
  "cis": "010463003407002921wskg1E44R1qym2406401",
  "gtin": "04630034070029",
  "tnvd": "6401100000",
  "tnvedGroup": "6401",
  "productName": " , . CRPT--99999, 52",
  "ownerName": " \ " \ ",
  "ownerInn": "7731369928",
  "producerName": " \ " \ ",
  "producerInn": "7731369928",
  "status": "INTRODUCED",
  "emissionDate": "2019-01-18T08:14:40.344Z",
  "producedDate": "2019-01-18T08:19:33.575Z",
  "type": "PRINTED",
  "emissionType": "LOCAL",
```

```
"markingType": "ON_PACKAGE",
"prodOrderType": "OPERATOR",
"lastDocId": "e76a1c45-c8ce-4961-84f1-1df47e5a7d35",
"name": " , . CRPT--99999, 52",
"brand": "",
"model": "CRPT--99999",
"certDoc": {
  "type": "CONFORMITY_CERT",
  "number": "122",
  "date": "2019-01-02T20:00:00.000Z"
},
"country": " ",
"productTypeDesc": "",
"color": "",
"materialDown": "",
"productSize": "52",
"materialUpper": "",
"materialLining": "",
"packageType": "BOX",
"productType": "310000022"
```



```
}
```

Метод получения списка КИЗов по идентификатору

GET /api/v3/facade/cis/cis_list

Заголовок запроса:

Content-Type: application/json

параметры запроса:

cis (array[string]) - массив идентификаторов кизов

http статус ответа в случае успеха:

200 - ОК

Пример запроса:

```
curl -X GET "http://10.73.69.69:8095/api/v3/facade/cis/cis_list?cis=1&cis=2" -H "accept: */*"
```

Заголовки ответа:

Content-Type: application/json

Тело ответа:

```
{
  "010463003407001221NwAyLM5EYYQKB2406404": {
    "cis": "010463003407001221NwAyLM5EYYQKB2406404",
    "gtin": "04630034070012",
    "producerName": " \  \",
    "status": "EMITTED",
    "emissionDate": 1550507912681,
    "ownerName": " \  \",
    "productName": " , , 17",
    "statusEx": "EMITTED"
  }
}
```

Документы

Метод получения списка документов, ранее загруженных в ИС МП

GET /api/v3/facade/doc/listV2

Параметры запроса:

Название параметра	Тип данных	Описание
Фильтрация		
dateFrom	Строка	Дата документа, от. Задается в формате yyyy-MM-dd'T'HH:mm:ss.SSS'Z Пример: 2019-01-01T03:00:00.000Z

dateTo	Строка	Дата документа, до. Задается в формате уууу-ММ-ддТТНН:мм:сс.SSS'Z Пример: 2019-01-10T03:00:00.000Z
number	Строка	Номер документа.
documentStatus	Строка	Статус документа. Список доступных значений: IN_PROGRESS - проверяется CHECKED_OK - оформлен CHECKED_NOT_OK - ошибка при проверке PROCESSING_ERROR - ошибка при обработке UNDEFINED - не определен.
documentType	Массив строковых перечислений	Тип документа. Список доступных значений: LP_INTRODUCE_GOODS - ввод в оборот LP_INTRODUCE_GOODS_CSV - ввод в оборот через CSV-файл LP_INTRODUCE_GOODS_XML - ввод в оборот через XML-файл LP_SHIP_GOODS - отгрузка LP_SHIP_GOODS_CSV - отгрузка через CSV-файл LP_SHIP_GOODS_XML - отгрузка через XML-файл LP_ACCEPT_GOODS - приемка LP_ACCEPT_GOODS_XML - приемка через XML-файл
inputFormat	Boolean	true - если документ входящий, false - если исходящий
participantInn	Строка	ИНН участника товарооборота
Пагинация/Сортировка		
order	Строковое перечисление	Тип сортировки (ascending/descending). Список доступных значений: ASC - ascending DESC - descending
did	Строка	Значение номера документа-"точки отсчета", по которому сортируются записи.
orderedColumnValue	Строка	Значение столбца-"точки отсчета", по которому сортируются записи.
orderColumn	Строковое перечисление	Название столбца, по которому будет производиться сортировка. Список доступных значений: docDate - дата документа
pageDir	Строковое перечисление	Выбор направления (вперед/назад) для пагинации. Список доступных значений: NEXT - вперед PREV - назад
limit	Целое число	Максимальное количество записей, которое вернется в качестве ответа.

http статус ответа в случае успеха:

200 - ОК

Тело ответа:

```
{
  "results": [
    {
      "number": "b917dfb0-523d-41e0-9e64-e8bf0052c5bd",
      "docDate": "2019-01-18T06:45:35.630Z",
      "receivedAt": 1547793935630,
      "type": "LP_INTRODUCE_GOODS",
      "status": "CHECKED_OK",
      "senderName": " \ " \ " ,
    }
  ]
}
```

Атрибутивный состав ответа:

number - Номер документа.

docDate - Дата документа.

receivedAt - Unix-время получения документа.

type - Тип документа. LP_INTRODUCE_GOODS - ввод в оборот. LP_INTRODUCE_GOODS_CSV - ввод в оборот через CSV-файл. LP_INTRODUCE_GOODS_XML - ввод в оборот через XML-файл. LP_SHIP_GOODS - отгрузка. LP_SHIP_GOODS_CSV - отгрузка через CSV-файл. LP_SHIP_GOODS_XML - отгрузка через XML-файл. LP_ACCEPT_GOODS - приемка. LP_ACCEPT_GOODS_XML - приемка через XML-файл.

status - Статус документа. IN_PROGRESS - проверяется, CHECKED_OK - оформлен, CHECKED_NOT_OK - ошибка при проверке, PROCESSING_ERROR - ошибка при обработке, UNDEFINED - не определен.

senderName - наименование отправителя документа.

Пример запроса:

```
curl
'https://ismp.crpt.ru/api/v3/facade/doc/listV2?limit=10&order=DESC&order
Column=docDate&did=623136d3-7a9b-40c9-8ce3-8091e41f83aa&orderedColumnVal
ue=2019-01-28T09:30:40.136Z&pageDir=NEXT' -H 'content-type:
application/json' -H 'Authorization: Bearer <>'
:
< HTTP/1.1 200
< Server: nginx/1.14.0
< Date: Sun, 27 Jan 2019 18:10:33 GMT
< Content-Type: application/json;charset=UTF-8
< Transfer-Encoding: chunked
< Connection: keep-alive
< X-Content-Type-Options: nosniff
< X-XSS-Protection: 1; mode=block
< Cache-Control: no-cache, no-store, max-age=0, must-revalidate
< Pragma: no-cache
< Expires: 0
< X-Frame-Options: DENY
<

* Connection #0 to host ismp.crpt.ru left intact
{
  "results": [
    {
      "number": "b917dfb0-523d-41e0-9e64-e8bf0052c5bd",
      "docDate": "2019-01-18T06:45:35.630Z",
      "receivedAt": 1547793935630,
      "type": "LP_INTRODUCE_GOODS",
      "status": "CHECKED_OK",
      "senderName": " \ "  \ " ",
    }
  ]
}
```

Метод получения документа, ранее загруженного в ИС МП

GET /api/v3/facade/doc/{docId}/body

Параметры url:

docId - номер документа

http статус ответа в случае успеха:

200 - ОК

Тело ответа:

```

{
  "number": < >,
  "docDate": < >, "2019-01-18T08:19:33.575Z",
  "type": < >, "LP_INTRODUCE_GOODS",
  "status": < >, "CHECKED_OK",
  "senderName": < >,
  "body": < json>,
  "content": < >
}

```

number - Номер документа.

docDate - Дата документа.

receivedAt - Unix-время получения документа.

type - Тип документа. LP_INTRODUCE_GOODS - ввод в оборот. LP_INTRODUCE_GOODS_CSV - ввод в оборот через CSV-файл. LP_INTRODUCE_GOODS_XML - ввод в оборот через XML-файл. LP_SHIP_GOODS - отгрузка. LP_SHIP_GOODS_CSV - отгрузка через CSV-файл. LP_SHIP_GOODS_XML - отгрузка через XML-файл. LP_ACCEPT_GOODS - приемка. LP_ACCEPT_GOODS_XML - приемка через XML-файл.

status - Статус документа. IN_PROGRESS - проверяется, CHECKED_OK - оформлен, CHECKED_NOT_OK - ошибка при проверке, PROCESSING_ERROR - ошибка при обработке, UNDEFINED - не определен.

senderName - наименование отправителя документа.

body - содержимое документа в формате JSON. В этом JSON может содержаться набор полей, специфичных для конкретного документа.

content - содержимое документа в начальном виде (если документ был добавлен в систему посредством загрузки CSV-файла, то в поле content будет содержимое этого CSV-файла).

Пример запроса:

```

curl -X GET
'https://ismp.crpt.ru/api/v3/facade/doc/e76a1c45-c8ce-4961-84f1-1df47e5a7d35/body' -H 'content-type: application/json' -H 'Authorization: Bearer <>'
:
< HTTP/1.1 200
< Server: nginx/1.14.0
< Date: Sun, 27 Jan 2019 18:10:33 GMT
< Content-Type: application/json; charset=UTF-8
< Transfer-Encoding: chunked
< Connection: keep-alive
< X-Content-Type-Options: nosniff
< X-XSS-Protection: 1; mode=block
< Cache-Control: no-cache, no-store, max-age=0, must-revalidate
< Pragma: no-cache
< Expires: 0
< X-Frame-Options: DENY
<

* Connection #0 to host ismp.crpt.ru left intact
{

```

```
"number": "e76alc45-c8ce-4961-84f1-1df47e5a7d35",
"docDate": "2019-01-18T08:19:33.575Z",
"type": "LP_INTRODUCE_GOODS",
"status": "CHECKED_OK",
"senderName": " \ " \ ",
"downloadStatus": "CHECKED_OK",
"body": {
  "reg_date": "2019-01-18T08:19:31.967Z",
  "document_description": {
    "participant_inn": "7731369928",
    "marking_type": "PRODUCT_PACKAGE",
    "production_date": "2019-01-17T20:00:00.000Z",
    "production_type": "OWN_PRODUCTION",
    "producer_inn": "7731369928",
    "owner_inn": "7731369928"
  },
  "doc_type": "Promotion_Inform_Selfmade",
  "doc_id": "e76alc45-c8ce-4961-84f1-1df47e5a7d35",
  "products": [
    {
      "uitu_code": "",
      "uit_code": "010463003407002921wskg1E44R1qym2406401",
      "tnved_code": "6401100000",
      "producer_inn": "7731369928",
      "owner_inn": "7731369928",
      "certificate_document": "CONFORMITY_CERT",
      "certificate_document_number": "122",
      "certificate_document_date": "2019-01-02T20:00:00.000Z"
    }
  ]
},
"content":
"{\"doc_id\": \"e76alc45-c8ce-4961-84f1-1df47e5a7d35\", \"doc_type\": \"Pro
motion_Inform_Selfmade\", \"reg_date\": \"2019-01-18T08:19:31.967Z\", \"doc
ument_description\": {\"participant_inn\": \"7731369928\", \"marking_type\"
: \"PRODUCT_PACKAGE\", \"production_date\": \"2019-01-17T20:00:00.000Z\", \"
production_type\": \"OWN_PRODUCTION\", \"producer_inn\": \"7731369928\", \"o
wner_inn\": \"7731369928\"}, \"products\": [{\"uitu_code\": \"\", \"uit_code\"
: \"010463003407002921wskg1E44R1qym2406401\", \"tnved_code\": \"6401100000
\", \"producer_inn\": \"7731369928\", \"owner_inn\": \"7731369928\", \"certif
icate_document\": \"CONFORMITY_CERT\", \"certificate_document_number\": \"1
22\", \"certificate_document_date\": \"2019-01-02T20:00:00.000Z\"}]}"
```

```
}
```

Методы создания документов

Url запроса на создания конкретного типа документа:

Тип документа	URL метода создания документа
Ввод товара в оборот	POST /api/v3/lk/documents/send
Ввод товара в оборот (Импорт)	POST /api/v3/lk/import/send
Отгрузка товара	POST /api/v3/lk/documents/shipment/create
Отмена отгрузки товара	POST /api/v3/lk/documents/shipment/cancel
Приемка товара	POST /api/v3/lk/documents/acceptance/create
Вывод товара из оборота при реализации договора конечному потребителю	POST /api/v3/lk/receipt/send
Вывод товара из оборота при реализации договора рассрочки очередного платежа	POST /api/v3/lk/loan_receipt/send
Ввод в оборот товара, произведенного на территории РФ (производство сторонней организацией)	POST /api/v3/lk/documents/commissioning/contract/create
Ввод в оборот товара, полученного от физических лиц, не являющихся индивидуальными предпринимателями, в том числе при комиссионной торговле	POST /api/v3/lk/documents/commissioning/indi/create
Списание не нанесенного на товар КМ с/без СИ	POST /api/v3/lk/documents/km/cancellation/create
Списание нанесенного на товар КМ с СИ (без вывода товара из оборота)	POST /api/v3/lk/documents/km/cancellation/applied/create
Перемаркировка товара	POST /api/v3/lk/remarking/send

Заголовок запроса:

Content-Type: application/json

Тело запроса:

```
{
  "product_document": "< base64> ",
  "document_format": "< >"
  "signature": "< base64>"
}
```

Атрибутивный состав ответа:

product_document - документ о вводе товара в оборот, закодированный в base64. Описание документа приведено в п. "Приложение. Документ о вводе в оборот". Тип строковый.

document_format - формат документа о вводе товара в оборот, возможные значения "MANUAL" - json, "CSV" - csv, "XML" - xml. Тип строковый.

signature - открепленная УКЭП закодированный в base64. Документ необходимо подписывать до перевода его в base64. Тип строковый.

Заголовки ответа:

Content-Type: application/json

Тело ответа:

```
<GUID>
```

Атрибутивный состав ответа:

<GUID> - уникальный идентификатор документа в ИС МП. Тип строковый

http статус ответа в случае успеха:

200 - ОК

Пример запроса:

```
curl 'https://ismp.crpt.ru/api/v3/lk/documents/send' -H 'content-type:
application/json' -H 'Authorization: Bearer <>'
--data-binary '{ "product_document": "< base64>",
"document_format": "MANUAL", "signature": "< base64>" }'
:
< HTTP/1.1 200
< Server: nginx/1.14.0
< Date: Sun, 27 Jan 2019 18:10:33 GMT
< Content-Type: application/json; charset=UTF-8
< Transfer-Encoding: chunked
< Connection: keep-alive
< X-Content-Type-Options: nosniff
< X-XSS-Protection: 1; mode=block
< Cache-Control: no-cache, no-store, max-age=0, must-revalidate
< Pragma: no-cache
< Expires: 0
< X-Frame-Options: DENY
<

* Connection #0 to host ismp.crpt.ru left intact
9abd3d41-76bc-4542-a88e-b1f7be8130b5
```

Метод создания черновика документа

POST /api/v3/lk/documents/draft

Заголовок запроса:

Тело запроса:


```
{
  "date_saved": "2019-02-13T07:59:05.785Z",
  "document_body": "string",
  "document_type": "UNKNOWN",
  "organisation_inn": "string",
  "registration_date": "2019-02-13T07:59:05.785Z",
  "user_id": 0
}
```

Атрибутивный состав запроса все поля обязательные:

date_saved - дата сохранения

document_body - тело документа. Тип строковый

document_type - тип документа. Тип строковый

organisation_inn - ИНН организации. Тип строковый

registration_date - Дата создания.

user_id - идентификатор пользователя. Тип числовой

http статус ответа в случае успеха:

200 - ОК

Заголовки ответа:

Content-Type: application/json

Тело ответа:

```
{
  "date_saved": "2019-02-13T08:18:50.932Z",
  "document_body": "string",
  "document_type": "UNKNOWN",
  "document_uuid": "string",
  "draft_uuid": "string",
  "errors": "string",
  "organisation_inn": "string",
  "registration_date": "2019-02-13T08:18:50.932Z",
  "user_id": 0
}
```

Метод получения черновика документа

GET /api/v3/lk/documents/draft/{id}

http статус ответа в случае успеха:

200 - ОК

Заголовки ответа:

Content-Type: application/json

Тело ответа:

```
{
  "date_saved": "2019-02-13T10:28:36.520Z",
  "document_body": "string",
  "document_type": "UNKNOWN",
  "document_uuid": "string",
  "draft_uuid": "string",
  "errors": "string",
  "organisation_inn": "string",
  "registration_date": "2019-02-13T10:28:36.521Z",
  "user_id": 0
}
```

http статус ответа в случае успеха - 200

Ответ 204 - Черновик не найден

Метод обновления черновика документа

PUT /api/v3/lk/documents/draft/{id}

Заголовок запроса:

Content-Type: application/json

параметр запроса:

id - идентификатор черновика

Тело запроса:

```
{
  "date_saved": "2019-02-13T07:59:05.785Z",
  "document_body": "string",
  "document_type": "UNKNOWN",
  "organisation_inn": "string",
  "registration_date": "2019-02-13T07:59:05.785Z",
  "user_id": 0
}
```

Атрибутивный состав запроса все поля обязательные:

date_saved - дата сохранения

document_body - тело документа. Тип строковый

document_type - тип документа. Тип строковый

organisation_inn - ИНН организации. Тип строковый

registration_date - Дата создания.

user_id - идентификатор пользователя. Тип числовой

http статус ответа в случае успеха:

200 - ОК

Заголовки ответа:

Content-Type: application/json

Тело ответа:

```
{
  "date_saved": "2019-02-13T10:58:47.990Z",
  "document_body": "string",
  "document_type": "UNKNOWN",
  "document_uuid": "string",
  "draft_uuid": "string",
  "errors": "string",
  "organisation_inn": "string",
  "registration_date": "2019-02-13T10:58:47.990Z",
  "user_id": 0
}
```

Метод удаления черновика документа

DELETE /api/v3/lk/documents/draft/{id}

Параметры запроса:

id - id черновика

http статус ответа в случае успеха:

200 - OK

Метод поиска и фильтрации черновика документа

GET /api/v3/lk/documents/draft/search

http статус ответа в случае успеха:

200 - OK

Параметры запроса:

id - идентификатор черновика. Тип UUID

registerDateStart - Дата регистрации с.

registerDateEnd - Дата регистрации по.

dateSavedStart - Дата последнего сохранения с.

dateSavedEnd - Дата последнего сохранения по. Пример: 2019-04-08T16:41:44.521Z

userId - Идентификатор пользователя. Тип числовой

organisationInn - ИНН организации. Тип строковый

documentType - Тип документа. Тип строковый

orderColumn - Поле для сортировки. Тип строковый

orderDirection - Тип сортировки.ASC/DESC

limit - Размер раздела. Тип числовой.

page - Номер раздела. Тип числовой.

onDelete - Флаг на удаление. true/false.

Пример запроса:

```
http://10.73.69.69:8120/api/v3/lk/documents/draft/search?limit=10&orderDirection=ASC&page=2&userId=1
```

Тело ответа:

```
[
  {
    "draft_uuid": "0b35d630-47e6-45df-b0e0-12fe04540042",
    "registration_date": "2019-02-13T21:53:47.708Z",
    "date_saved": "2019-02-13T21:53:47.708Z",
    "user_id": 2,
    "organisation_inn": "7724211288",
    "document_type": "UNKNOWN",
    "document_uuid": "string",
    "errors": "string"
  }
]
```

Метод получения списка КИЗов по идентификатору

GET /api/v3/facade/cis/cis_list

Заголовок запроса:

Content-Type: application/json

параметры запроса:

cis (array[string]) - массив идентификаторов кизов

http статус ответа в случае успеха:

200 - OK

Пример запроса:

```
curl -X GET "http://stable.fashion.crpt.ru/api/v3/facade/cis/cis_list?cis=010463003759004321fffaabbssuasfasfasfasf240A77A&cis=010463003759004321fffaaccssuasfasfasfasf240A77A" -H "accept: */*"
```

Заголовки ответа:

Content-Type: application/json

Тело ответа:

```
{
  "010463003407001221NwAyLM5EYYQKB2406404": {
    "cis": "010463003407001221NwAyLM5EYYQKB2406404",
    "gtin": "04630034070012",
    "producerName": " \ " \ " ",
    "status": "EMITTED",
    "emissionDate": 1550507912681,
    "ownerName": " \ " \ " ",
    "productName": " , , 17",
    "statusEx": "EMITTED"
  }
}
```

Добавление заявки на регистрацию УОТ

POST api/v3/lk/registration/send

Тело запроса:

```
{
  "product_document": "< base64>",
  "document_format": "< >",
  "signature": "< base64>"
}
```

Атрибутивный состав ответа:

product_document - документ о приемке, закодированный в base64. Описание документа приведено в п. "Приложение. Документ о регистрации УОТ". Тип строковый.

document_format - формат документа о приемке, возможные значения "MANUAL" - json, "CSV" - csv, "XML" - xml. Тип строковый.

signature - открепленная УКЭП закодированный в base64. Документ необходимо подписывать до перевода его в base64. Тип строковый.

http статус ответа в случае успеха:

200 - ОК

Заголовки ответа:

Content-Type: application/json

Тело ответа:

```
{
  "string": "<document_id>"
}
```

Пример запроса:

```
curl 'https://ismp.crpt.ru/api/v3/lk/registration/send' -H
'content-type: application/json' -H 'Authorization: Bearer <>'
--data-binary '{ "product_document": "< base64>",
"document_format": "MANUAL", "signature": "< base64>" }'
:
< HTTP/1.1 200
< Server: nginx/1.14.0
< Date: Sun, 27 Jan 2019 18:10:33 GMT
< Content-Type: application/json; charset=UTF-8
< Transfer-Encoding: chunked
< Connection: keep-alive
< X-Content-Type-Options: nosniff
< X-XSS-Protection: 1; mode=block
< Cache-Control: no-cache, no-store, max-age=0, must-revalidate
< Pragma: no-cache
< Expires: 0
< X-Frame-Options: DENY
<
```

Регистрация участника

Добавление заявки на регистрацию УОТ

POST api/v3/lk/registration/send

Тело запроса:

```
{
  "product_document": "< base64>",
  "document_format": "< >",
  "signature": "< base64>"
}
```

Атрибутивный состав ответа:

product_document - документ о приемке, закодированный в base64. Описание документа приведено в п. "Приложение. Документ о регистрации УОТ". Тип строковый.

document_format - формат документа о приемке, возможные значения "MANUAL" - json, "CSV" - csv, "XML" - xml. Тип строковый.

signature - открепленная УКЭП закодированный в base64. Документ необходимо подписывать до перевода его в base64. Тип строковый.

http статус ответа в случае успеха:

200 - ОК

Заголовки ответа:

Content-Type: application/json

Тело ответа:

```
{
  "string": "<document_id>"
}
```

Пример запроса:

```
curl 'https://ismp.crpt.ru/api/v3/lk/registration/send' -H
'content-type: application/json' -H 'Authorization: Bearer <>'
--data-binary '{ "product_document": "< base64>",
"document_format": "MANUAL", "signature": "< base64>" }'
:
< HTTP/1.1 200
< Server: nginx/1.14.0
< Date: Sun, 27 Jan 2019 18:10:33 GMT
< Content-Type: application/json; charset=UTF-8
< Transfer-Encoding: chunked
< Connection: keep-alive
< X-Content-Type-Options: nosniff
< X-XSS-Protection: 1; mode=block
< Cache-Control: no-cache, no-store, max-age=0, must-revalidate
< Pragma: no-cache
< Expires: 0
< X-Frame-Options: DENY
<
```

Добавление нового пользователя в реестр ИС МП

POST api/v3/lk/registration/app-user/add/send

Тело запроса:

```
{
  "product_document": "< base64>",
  "document_format": "< >",
  "signature": "< base64>"
}
```

Атрибутивный состав ответа:

product_document - документ о добавлении пользователей, закодированный в base64. Описание документа приведено в п. "Приложение. Документ о добавлении пользователей". Тип строковый.

document_format - формат документа о добавлении пользователей, возможные значения "MANUAL" - json, "XML" - xml. Тип строковый.

signature - открепленная УКЭП закодированный в base64. Документ необходимо подписывать до перевода его в base64. Тип строковый.

http статус ответа в случае успеха:

200 - OK

Заголовки ответа:

Content-Type: application/json

Тело ответа:

```
{
  "string": "<document_id>"
}
```

Пример запроса:

```
curl 'https://ismp.crpt.ru/api/v3/lk/registration/app-user/add/send' -H
'content-type: application/json' -H 'Authorization: Bearer <>'
--data-binary '{ "product_document": "< base64>",
"document_format": "MANUAL", "signature": "< base64>" }'
:
< HTTP/1.1 200
< Server: nginx/1.14.0
< Date: Sun, 27 Jan 2019 18:10:33 GMT
< Content-Type: application/json;charset=UTF-8
< Transfer-Encoding: chunked
< Connection: keep-alive
< X-Content-Type-Options: nosniff
< X-XSS-Protection: 1; mode=block
< Cache-Control: no-cache, no-store, max-age=0, must-revalidate
< Pragma: no-cache
< Expires: 0
< X-Frame-Options: DENY
<
```

Добавление сертификатов пользователям

POST api/v3/lk/registration/app-user/cert/send

Тело запроса:

```
{
  "product_document": "< base64>",
  "document_format": "< >",
  "signature": "< base64>"
}
```

Атрибутивный состав ответа:

product_document - документ о добавлении сертификатов, закодированный в base64. Описание документа приведено в п. "Приложение."

Документ о добавлении сертификатов пользователям". Тип строковый.

document_format - формат документа о добавлении сертификатов, возможные значения "MANUAL" - json, "XML" - xml. Тип строковый.

signature - открепленная УКЭП закодированный в base64. Документ необходимо подписывать до перевода его в base64. Тип строковый.

http статус ответа в случае успеха:

200 - ОК

Заголовки ответа:

Content-Type: application/json

Тело ответа:

```
{
  "string": "<document_id>"
}
```

Пример запроса:

```
curl 'https://ismp.crpt.ru/api/v3/lk/registration/app-user/cert/send' -H
'content-type: application/json' -H 'Authorization: Bearer <>'
--data-binary '{ "product_document": "< base64>",
"document_format": "MANUAL", "signature": "< base64>" }'
:
< HTTP/1.1 200
< Server: nginx/1.14.0
< Date: Sun, 27 Jan 2019 18:10:33 GMT
< Content-Type: application/json; charset=UTF-8
< Transfer-Encoding: chunked
< Connection: keep-alive
< X-Content-Type-Options: nosniff
< X-XSS-Protection: 1; mode=block
< Cache-Control: no-cache, no-store, max-age=0, must-revalidate
< Pragma: no-cache
< Expires: 0
< X-Frame-Options: DENY
<
```

Блокировка пользователя

POST api/v3/lk/registration/app-user/block/send

Тело запроса:

```
{
  "product_document": "< base64>",
  "document_format": "< >",
  "signature": "< base64>"
}
```

Атрибутивный состав ответа:

product_document - документ по блокировке, закодированный в base64. Описание документа приведено в п. "Приложение. Документ по блокировке/возобновлению/удалению пользователей". Тип строковый.

document_format - формат документа по блокировке пользователя, возможные значения "MANUAL" - json, "XML" - xml. Тип строковый.

signature - открепленная УКЭП закодированный в base64. Документ необходимо подписывать до перевода его в base64. Тип строковый.

http статус ответа в случае успеха:

200 - ОК

Заголовки ответа:

Content-Type: application/json

Тело ответа:

```
{
  "string": "<document_id>"
}
```

Пример запроса:

```
curl 'https://ismp.crpt.ru/api/v3/lk/registration/app-user/block/send'
-H 'content-type: application/json' -H 'Authorization: Bearer <>'
--data-binary '{ "product_document": "< base64>",
"document_format": "MANUAL", "signature": "< base64>" }'
:
< HTTP/1.1 200
< Server: nginx/1.14.0
< Date: Sun, 27 Jan 2019 18:10:33 GMT
< Content-Type: application/json; charset=UTF-8
< Transfer-Encoding: chunked
< Connection: keep-alive
< X-Content-Type-Options: nosniff
< X-XSS-Protection: 1; mode=block
< Cache-Control: no-cache, no-store, max-age=0, must-revalidate
< Pragma: no-cache
< Expires: 0
< X-Frame-Options: DENY
<
```

Возобновление пользователя

POST api/v3/lk/registration/app-user/resume/send

Тело запроса:

```
{
  "product_document": "< base64>",
  "document_format": "< >",
  "signature": "< base64>"
}
```

Атрибутивный состав ответа:

product_document - документ о возобновлению, закодированный в base64. Описание документа приведено в п. "Приложение. Документ по блокировке/возобновлению/удалению пользователей". Тип строковый.

document_format - формат документа по возобновлению пользователя, возможные значения "MANUAL" - json, "XML" - xml. Тип строковый.

signature - открепленная УКЭП закодированный в base64. Документ необходимо подписывать до перевода его в base64. Тип строковый.

http статус ответа в случае успеха:

200 - ОК

Заголовки ответа:

Content-Type: application/json

Тело ответа:

```
{
  "string": "<document_id>"
}
```

Пример запроса:

```
curl 'https://ismp.crpt.ru/api/v3/lk/registration/app-user/resume/send'
-H 'content-type: application/json' -H 'Authorization: Bearer <>'
--data-binary '{ "product_document": "< base64>",
"document_format": "MANUAL", "signature": "< base64>" }'
:
< HTTP/1.1 200
< Server: nginx/1.14.0
< Date: Sun, 27 Jan 2019 18:10:33 GMT
< Content-Type: application/json; charset=UTF-8
< Transfer-Encoding: chunked
< Connection: keep-alive
< X-Content-Type-Options: nosniff
< X-XSS-Protection: 1; mode=block
< Cache-Control: no-cache, no-store, max-age=0, must-revalidate
< Pragma: no-cache
< Expires: 0
< X-Frame-Options: DENY
<
```

Удаление пользователя

POST api/v3/lk/registration/app-user/remove/send

Тело запроса:

```
{
  "product_document": "< base64>",
  "document_format": "< >",
  "signature": "< base64>"
}
```

Атрибутивный состав ответа:

product_document - документ по удалению, закодированный в base64. Описание документа приведено в п. "Приложение. Документ по блокировке/возобновлению/удалению пользователей". Тип строковый.

document_format - формат документа по удалению пользователя, возможные значения "MANUAL" - json, "XML" - xml. Тип строковый.

signature - открепленная УКЭП закодированный в base64. Документ необходимо подписывать до перевода его в base64. Тип строковый.

http статус ответа в случае успеха:

200 - ОК

Заголовки ответа:

Content-Type: application/json

Тело ответа:

```
{
  "string": "<document_id>"
}
```

Пример запроса:

```
curl 'https://ismp.crpt.ru/api/v3/lk/registration/app-user/remove/send'
-H 'content-type: application/json' -H 'Authorization: Bearer <>'
--data-binary '{ "product_document": "< base64>",
"document_format": "MANUAL", "signature": "< base64>" }'
:
< HTTP/1.1 200
< Server: nginx/1.14.0
< Date: Sun, 27 Jan 2019 18:10:33 GMT
< Content-Type: application/json; charset=UTF-8
< Transfer-Encoding: chunked
< Connection: keep-alive
< X-Content-Type-Options: nosniff
< X-XSS-Protection: 1; mode=block
< Cache-Control: no-cache, no-store, max-age=0, must-revalidate
< Pragma: no-cache
< Expires: 0
< X-Frame-Options: DENY
<
```

Списание нанесенных КМ

POST /api/v3/lk/documents/km/cancellation/applied/create

Тело запроса:

```
{
  "product_document": "< base64>",
  "document_format": "< >",
  "signature": "< base64>"
}
```

Атрибутивный состав ответа:

product_document - документ по удалению, закодированный в base64. Описание документа приведено в п. "Приложение. Документ по списанию КМ без СИ". Тип строковый.

document_format - формат документа по удалению пользователя, возможные значения "MANUAL" - json, "XML" - xml. Тип строковый.

signature - открепленная УКЭП закодированный в base64. Документ необходимо подписывать до перевода его в base64. Тип строковый.

http статус ответа в случае успеха:

200 - ОК

Заголовки ответа:

Content-Type: application/json

Тело ответа:

```
{
  "string": "<document_id>"
}
```

Пример запроса:

```
curl
'http://lk-doc-api.apps.qa.lp.crpt.tech/api/v3/lk/documents/km/cancellation/applied/create' -H 'content-type: application/json' -H
'Authorization: Bearer <>'
--data-binary '{ "product_document": "< base64>",
"document_format": "MANUAL", "signature": "< base64>" }'
:
< HTTP/1.1 200
< Server: nginx/1.14.0
< Date: Sun, 27 Jan 2019 18:10:33 GMT
< Content-Type: application/json; charset=UTF-8
< Transfer-Encoding: chunked
< Connection: keep-alive
< X-Content-Type-Options: nosniff
< X-XSS-Protection: 1; mode=block
< Cache-Control: no-cache, no-store, max-age=0, must-revalidate
< Pragma: no-cache
< Expires: 0
< X-Frame-Options: DENY
<
```

Списание не нанесенных КМ

POST api/v3/lk/documents/km/cancellation/create

Тело запроса:

```
{
  "product_document": "< base64>",
  "document_format": "< >",
  "signature": "< base64>"
}
```

Атрибутивный состав ответа:

product_document - документ по удалению, закодированный в base64. Описание документа приведено в п. "Приложение. Документ по списанию КМ без СИ". Тип строковый.

document_format - формат документа по удалению пользователя, возможные значения "MANUAL" - json, "XML" - xml. Тип строковый.

signature - открепленная УКЭП закодированный в base64. Документ необходимо подписывать до перевода его в base64. Тип строковый.

http статус ответа в случае успеха:

200 - ОК

Заголовки ответа:

Content-Type: application/json

Тело ответа:

```
{
  "string": "<document_id>"
}
```

Пример запроса:

Справочник товаров GS1

Метод получения списка gtin участника оборота товара по ИНН

GET /api/v3/product/search

Параметры запроса:

limit - максимальное количество заказов в КМ в ответе. Опциональный параметр.

offset - номер страницы ответа, начальное значение 0. Опциональный параметр.

http статус ответа в случае успеха:

200 - ОК

Заголовки ответа:

Content-Type: application/json

Тело ответа:

```

{
  "errorCode": < >,
  "results": < > [
    {
      "id": <Id >, 30056,
      "name": < > "001079-11", 16",
      "brand": < >, "brand": "",
      "gtin": "<GTIN >, "04690257113712",
      "packageType": < >, "BOX",
      "inn":< >, "5011017647",
      "model": "string", "001079-11",
      "publicationDate": < >, 1537304400000,
      "country":< >, "",
      "productTypeDesc":< >, "",
      "color":<>, "",
      "materialDown": < >"",
      "materialUpper":< >, "",
      "materialLining":< >, "",
      "productType": "310000013",
      "innerUnitCount": 1,
    }
  ],
  "total": < >
}

```

Атрибутивный состав ответа:

id – Уникальный номер товара. Тип числовой.

name - Наименование товара. Тип строковый.

brand - Наименование бренда. Тип строковый.

gtin - GTIN товара. Тип строковый.

packageType - Тип упаковки. Тип строковый.

inn - ИНН Российского производителя. Тип строковый.

model – Модель производителя. Тип строковый.

publicationDate - Дата публикации. Формат Unix time в миллисекундах. Тип числовой.

country - Страна производства. Тип строковый

productTypeDesc - Наименование на этикетке. Тип строковый.

color – Цвет. Тип строковый.

materialDown - Материал низа. Тип строковый.

materialUpper - Материал верха. Тип строковый.

materialLining - Материал подкладки. Тип строковый.

productType – Тип продукта. Тип строковый.

innerUnitCount – Число экземпляров товара. Тип числовой.

http статус ответа в случае успеха:

200 - OK

Пример запроса:

```
curl -X GET
'https://ismp.crpt.ru/api/v3/product/search?limit=10&offset=0&page=0&pre
fix=6401,6402,6403,6404,6405' -H 'Authorization: Bearer <>'
:
< HTTP/1.1 200
< Server: nginx/1.14.0
< Date: Sun, 27 Jan 2019 18:10:33 GMT
< Content-Type: application/json;charset=UTF-8
< Transfer-Encoding: chunked
< Connection: keep-alive
< X-Content-Type-Options: nosniff
< X-XSS-Protection: 1; mode=block
< Cache-Control: no-cache, no-store, max-age=0, must-revalidate
< Pragma: no-cache
< Expires: 0
< X-Frame-Options: DENY
<

* Connection #0 to host ismp.crpt.ru left intact
{
  "results": [
    {
      "id": 30056,
      "name": "001079-11  , 16",
      "gtin": "04690257113712",
      "brand": "",
      "packageType": "BOX",
      "innerUnitCount": 1,
      "model": "001079-11",
      "publicationDate": 1537304400000,
      "inn": "5011017647",
      "country": "",
      "productTypeDesc": "",
      "color": "",
      "materialDown": "",
      "materialUpper": "",
      "materialLining": "",
      "productType": "310000013"
    },
    {
      "id": 30045,
      "name": "001079-12  , 16",
      "gtin": "04690257114078",
      "brand": "",
      "packageType": "BOX",
      "innerUnitCount": 1,
```

```
"model": "001079-12",
"publicationDate": 1537304400000,
"inn": "5011017647",
"country": "",
"productTypeDesc": "",
"color": "",
"materialDown": "",
"materialUpper": "",
"materialLining": "",
"productType": "310000013"
},
{
  "id": 30059,
  "name": "001079-12  , 18",
  "gtin": "04690257114108",
  "brand": "",
  "packageType": "BOX",
  "innerUnitCount": 1,
  "model": "001079-12",
  "publicationDate": 1537304400000,
  "inn": "5011017647",
  "country": "",
  "productTypeDesc": "",
  "color": "",
  "materialDown": "",
  "materialUpper": "",
  "materialLining": "",
  "productType": "310000013"
}
],
```

```
"total": 11210,  
"errorCode": null  
}
```

Метод получения информации о товаре по gtin

GET api/v3/product/info

Параметры запроса:

gtins – список GTIN.

http статус ответа в случае успеха:

200 - ОК

Пример запроса:

```

curl -X GET
'https://ismp.crpt.ru/api/v3/product/api/v3/product/info?gtins=046902571
13712,04690257113712' -H 'Authorization: Bearer <>'
:
< HTTP/1.1 200
< Server: nginx/1.14.0
< Date: Sun, 27 Jan 2019 18:10:33 GMT
< Content-Type: application/json;charset=UTF-8
< Transfer-Encoding: chunked
< Connection: keep-alive
< X-Content-Type-Options: nosniff
< X-XSS-Protection: 1; mode=block
< Cache-Control: no-cache, no-store, max-age=0, must-revalidate
< Pragma: no-cache
< Expires: 0
< X-Frame-Options: DENY
<

* Connection #0 to host ismp.crpt.ru left intact
{
  "errorCode": < >,
  "results": < > [
    {
      "id": <Id >, 30056,
      "name": < > "001079-11", 16",
      "brand": < >, "brand": "",
      "gtin": "<GTIN >, 04690257113712",
      "packageType": < >, "BOX",
      "inn":< >, "5011017647",
      "innerUnitCount": 1,
      "model": "string", "001079-11",
      "publicationDate": < >, 1537304400000,
      "country":< >, "",
      "productTypeDesc":< >, "",
      "color":<>, "",
      "materialDown": < >"",
      "materialUpper":< >, "",
      "materialLining":< >, "",
      "productType": "310000013",
    }
  ],
  "total": < >
}

```

Метод получения списка 10-ти значных кодов ТН ВЭД

GET /api/v3/facade/tnved/search

Параметры запроса:

prefix - массив 4-х значных кодов ТН ВЭД, используемых в качестве префиксов 10-ти значных кодов ТН ВЭД в ответе

limit - максимальное количество заказов в КМ в ответе. Опциональный параметр.

offset - номер страницы ответа, начальное значение 0. Опциональный параметр.

Заголовки ответа:

Content-Type: application/json

Тело ответа:

```
{
  "records": [
    {
      "code": <10-    >, "6401100000",
      "description": < >
    },
    {
      "code": <10-    >, "6401100000",
      "description": < >
    }
  ],
  "total": 71
}
```

Атрибутивный состав ответа:

records - список 10-ти значных кодов ТН ВЭД, удовлетворяющих условиям запроса. Тип массив.

code - 10-ти значный код ТН ВЭД. Тип строковый.

description - текстовое описание кода ТН ВЭД. Тип строковый.

http статус ответа в случае успеха:

200 - ОК

Пример запроса:

```

curl -X GET
'https://ismp.crpt.ru/api/v3/facade/tnved/search?limit=10&offset=0&page=
0&prefix=6401,6402,6403,6404,6405' -H 'Authorization: Bearer <>'
:
< HTTP/1.1 200
< Server: nginx/1.14.0
< Date: Sun, 27 Jan 2019 18:10:33 GMT
< Content-Type: application/json;charset=UTF-8
< Transfer-Encoding: chunked
< Connection: keep-alive
< X-Content-Type-Options: nosniff
< X-XSS-Protection: 1; mode=block
< Cache-Control: no-cache, no-store, max-age=0, must-revalidate
< Pragma: no-cache
< Expires: 0
< X-Frame-Options: DENY
<

* Connection #0 to host ismp.crpt.ru left intact
{
  "records": [
    {
      "code": "6401100000",
      "description": "
    . -
    },
    {
      "code": "6401921000",
      "description": "
    . - - -
    }
  ],
  "total": 71
}

```

Реестр СУЗ (станция управления заказами)

Метод для добавления СУЗ (пользовательский)

POST /api/v3/facade/suz/registry

Заголовок запроса:

Content-Type: application/json

Тело запроса:

```
{
  "document": "< base64>",
  "signature": "< base64>"
}
```

Атрибутивный состав запроса:

document - документ с информацией о СУЗ, закодированный в base64. Описание документа приведено в п. "Приложение. Документ с информацией о СУЗ". Тип строковый

signature - открепленная УКЭП закодированный в base64. Документ необходимо подписывать до перевода его в base64. Тип строковый.

Заголовки ответа:

Content-Type: application/json

Тело ответа:

```
{
  "suz_id": "<GUID>",
  "suz_status": "< >"
}
```

Атрибутивный состав ответа:

suz_id - Идентификатор СУЗ, тип строковый в формате GUID

suz_status - Статус СУЗ, возможные значения "ACTIVE" - запись СУЗ активна, "DELETED" - запись СУЗ удалена. Тип строковый.

http статус ответа в случае успеха:

201 - Created

http статус ответа в случае ошибки:

400, 422 - при ошибках в атрибутивном составе тела запроса

401 - Требуется авторизация

500 - произошла внутренняя ошибка сервера

Пример запроса:

```
curl -X POST 'https://ismp.crpt.ru/api/v3/facade/suz/registry' -H
'content-type: application/json' -H 'Authorization: Bearer <>'
--data-binary '{ "document": "< base64>", "signature": "< base64>"}'
:
< HTTP/1.1 201
< Server: nginx/1.14.0
< Date: Sun, 27 Jan 2019 09:41:23 GMT
< Content-Type: application/json; charset=UTF-8
< Transfer-Encoding: chunked
< Connection: keep-alive
< X-Content-Type-Options: nosniff
< X-XSS-Protection: 1; mode=block
< Cache-Control: no-cache, no-store, max-age=0, must-revalidate
< Pragma: no-cache
< Expires: 0
< X-Frame-Options: DENY
<
* Connection #0 to host ismp.crpt.ru left intact
{"suz_id": "a82e4f78-3dc5-43a0-9a36-5c8c2ec8eb20", "suz_status": "ACTIVE" }
```

Метод для удаления СУЗ (пользовательский)

DELETE /api/v3/facade/suz/registry/{id записи СУЗ}

Заголовок запроса:

Content-Type: application/json

Тело запроса пустое

Заголовки ответа:

Content-Type: application/json

Тело ответа пустое

http статус ответа в случае успеха:

200 - Ok

http статус ответа в случае ошибки:

401 - Требуется авторизация

422 - при ошибках в атрибутивном составе тела запроса

500 - произошла внутренняя ошибка сервера

Пример запроса:


```
curl -X DELETE
'https://ismp.crpt.ru/api/v3/facade/suz/registry/a82e4f78-3dc5-43a0-9a36-5c8c2ec8eb20' -H 'content-type: application/json' -H 'Authorization: Bearer <>'
:
< HTTP/1.1 200
< Server: nginx/1.14.0
< Date: Sun, 27 Jan 2019 09:41:23 GMT
< Content-Type: application/json;charset=UTF-8
< Transfer-Encoding: chunked
< Connection: keep-alive
< X-Content-Type-Options: nosniff
< X-XSS-Protection: 1; mode=block
< Cache-Control: no-cache, no-store, max-age=0, must-revalidate
< Pragma: no-cache
< Expires: 0
< X-Frame-Options: DENY
<
* Connection #0 to host ismp.crpt.ru left intact
```

Метод для получения текущего активного СУЗ (пользовательский)

GET /api/v3/facade/suz/registry

Заголовок запроса:

Content-Type: application/json

Тело запроса пустое

Заголовки ответа:

Content-Type: application/json

Тело ответа:

```
{
  "suz_id": "<GUID>",
  "suz_status": "< >",
  "suz_work_status": "< >",
  "certificate_serial_number": "< >",
  "certificate_owner_full_name": "< >",
  "certificate_authority_name": "< >",
  "certificate_authority_inn": "< >",
  "error_description": "< >",
  "create_date": "< >"
}
```

Атрибутивный состав ответа:

suz_id - Идентификатор СУЗ, тип строковый в формате GUID

suz_status - Статус СУЗ, возможные значения "ACTIVE" - запись СУЗ активна, "DELETED" - запись СУЗ удалена. Тип строковый.

suz_work_status - Статус работы СУЗ, возможные значения "OK" - СУЗ работает в штатном режиме, "ERROR" - в СУЗ произошла ошибка. Тип строковый.

certificate_serial_number - Серийный номер сертификата. Тип строковый.

certificate_owner_full_name - ФИО владельца сертификата. Тип строковый.

certificate_authority_name - Имя удостоверяющего центра выдавшего сертификат. Тип строковый.

certificate_authority_inn - ИНН удостоверяющего центра выдавшего сертификат. Тип строковый.

error_description - Информация об ошибке, если статус работы СУЗ ошибочный (suz_work_status == ERROR). Тип строковый.

create_date - Дата создания записи СУЗ. Формат Unix time в миллисекундах. Тип числовой.

http статус ответа в случае успеха:

200 - Ok

http статус ответа в случае ошибки:

400 - при ошибках в атрибутивном составе тела запроса

401 - Требуется авторизация

500 - произошла внутренняя ошибка сервера

Пример запроса:

```
curl 'https://ismp.crpt.ru/api/v3/facade/suz/registry' -H 'content-type:
application/json' -H 'Authorization: Bearer <>'
:
< HTTP/1.1 200
< Server: nginx/1.14.0
< Date: Sun, 27 Jan 2019 09:41:23 GMT
< Content-Type: application/json;charset=UTF-8
< Transfer-Encoding: chunked
< Connection: keep-alive
< X-Content-Type-Options: nosniff
< X-XSS-Protection: 1; mode=block
< Cache-Control: no-cache, no-store, max-age=0, must-revalidate
< Pragma: no-cache
< Expires: 0
< X-Frame-Options: DENY
<
* Connection #0 to host ismp.crpt.ru left intact
{"suz_id":"a82e4f78-3dc5-43a0-9a36-5c8c2ec8eb20","suz_status":"ACTIVE",
"suz_work_status":"OK","certificate_serial_number":"123456789",
"certificate_authority_name":"","certificate_authority_inn":"1234567890",
"certificate_owner_full_name":"","create_date":"1548582083721"}
```

Реестр регистраторов эмиссии

Метод получения списка регистраторов эмиссии

GET /api/v3/facade/registrar/

http статус ответа в случае успеха:

200 - ОК

Заголовки ответа:

Content-Type: application/json

Тело ответа:

```
"emission_registrars": [  
  {  
    "registrar_id": <GUID >,  
    "65f9bf16-0ae7-43ae-8b08-5c51c49636b1",  
    "ip_address": <IP >, "8.9.10.202",  
    "command_port": < >, "2021",  
    "notification_port": < >, "2022",  
    "address": < >  
    "serial_number": < >  
  }  
]  
}
```

Атрибутивный состав ответа:

registrar_id - уникальный идентификатор РЭ в ИС МП. Тип строковый.

ip_address - ip РЭ. Тип строковый.

command_port - командный порт РЭ. Тип строковый.

notification_port - нотификационный порт РЭ. Тип строковый.

address - почтовый адрес места установки РЭ. Тип строковый.

serial_number - серийный номер РЭ. Тип строковый.

http статус ответа в случае успеха:

200 - ОК

Пример запроса:

```
curl -X GET 'https://ismp.crpt.ru/api/v3/facade/registrar/' -H
'Authorization: Bearer <>'
:
< HTTP/1.1 200
< Server: nginx/1.14.0
< Date: Sun, 27 Jan 2019 18:10:33 GMT
< Content-Type: application/json;charset=UTF-8
< Transfer-Encoding: chunked
< Connection: keep-alive
< X-Content-Type-Options: nosniff
< X-XSS-Protection: 1; mode=block
< Cache-Control: no-cache, no-store, max-age=0, must-revalidate
< Pragma: no-cache
< Expires: 0
< X-Frame-Options: DENY
<

* Connection #0 to host ismp.crpt.ru left intact
{
  "emission_registrars": [
    {
      "registrar_id": "65f9bf16-0ae7-43ae-8b08-5c51c49636b1",
      "ip_address": "8.23.2.22",
      "command_port": "2021",
      "notification_port": "2022",
      "address": "",
      "serial_number": "00068911"
    }
  ]
}
```

Шаблоны этикеток

Метод получения списка шаблонов

GET /api/v3/facade/label/template/all

http статус ответа в случае успеха:

200 - ОК

Заголовки ответа:

Content-Type: application/json

Тело ответа:

```
{
  "label_templates": [
    {
      "label_template_id": <GUID >,
      "2dbae9e2-e5a2-492e-a987-edda887def58",
      "label_template_name": <>,
      "label_template_status": < >,
      "label_template_create_date": < Unix time ms>, 1548250658508,
      "label_template_update_date": < Unix time ms>, 1548250660419,
      "status_change_description": < >,
      "label_template_type": < >, "CUSTOM"
    }
  ]
}
```

Атрибутивный состав ответа:

label_templates - список шаблонов. Тип массив объектов.

label_template_name - наименования шаблона. Тип строковый.

label_template_status - статус валидации шаблона в ИС МП. Тип строковый.

label_template_create_date - время получения запроса на создание шаблона в ИС МП. Формат unix time в миллисекундах. Тип числовой.

label_template_update_date - время получения запроса на обновления шаблона в ИС МП. Формат unix time в миллисекундах. Тип числовой.

status_change_description - причина изменения статуса. Тип строковый.

label_template_type - тип шаблона. Тип строковый.

http статус ответа в случае успеха:

200 - ОК

Пример запроса:

```
curl -X GET '/api/v3/facade/label/template/all' -H 'Authorization:
Bearer <>'
:
< HTTP/1.1 200
< Server: nginx/1.14.0
< Date: Sun, 27 Jan 2019 18:10:33 GMT
< Content-Type: application/json;charset=UTF-8
< Transfer-Encoding: chunked
< Connection: keep-alive
< X-Content-Type-Options: nosniff
< X-XSS-Protection: 1; mode=block
< Cache-Control: no-cache, no-store, max-age=0, must-revalidate
< Pragma: no-cache
< Expires: 0
< X-Frame-Options: DENY
<

* Connection #0 to host ismp.crpt.ru left intact
{
  "emission_registrars": [
    {
      "registrar_id": "65f9bf16-0ae7-43ae-8b08-5c51c49636b1",
      "ip_address": "8.23.2.22",
      "command_port": "2021",
      "notification_port": "2022",
      "address": "",
      "serial_number": "00068911"
    }
  ]
}
```

Справочник товаров GS1

Метод получения списка gtin участника оборота товара по ИНН

GET /api/v3/product/search

Параметры запроса:

limit - максимальное количество заказов в КМ в ответе. Опциональный параметр.

offset - номер страницы ответа, начальное значение 0. Опциональный параметр.

http статус ответа в случае успеха:

200 - ОК

Заголовки ответа:

Content-Type: application/json

Тело ответа:

```

{
  "errorCode": < >,
  "results": < > [
    {
      "id": <Id >, 30056,
      "name": < > "001079-11", 16",
      "brand": < >, "brand": "",
      "gtin": "<GTIN >, "04690257113712",
      "packageType": < >, "BOX",
      "inn":< >, "5011017647",
      "model": "string", "001079-11",
      "publicationDate": < >, 1537304400000,
      "country":< >, "",
      "productTypeDesc":< >, "",
      "color":<>, "",
      "materialDown": < >"",
      "materialUpper":< >, "",
      "materialLining":< >, "",
      "productType": "310000013",
      "innerUnitCount": 1,
    }
  ],
  "total": < >
}

```

Атрибутивный состав ответа:

id – Уникальный номер товара. Тип числовой.

name - Наименование товара. Тип строковый.

brand - Наименование бренда. Тип строковый.

gtin - GTIN товара. Тип строковый.

packageType - Тип упаковки. Тип строковый.

inn - ИНН Российского производителя. Тип строковый.

model – Модель производителя. Тип строковый.

publicationDate - Дата публикации. Формат Unix time в миллисекундах. Тип числовой.

country - Страна производства. Тип строковый

productTypeDesc - Наименование на этикетке. Тип строковый.

color – Цвет. Тип строковый.

materialDown - Материал низа. Тип строковый.

materialUpper - Материал верха. Тип строковый.

materialLining - Материал подкладки. Тип строковый.

productType – Тип продукта. Тип строковый.

innerUnitCount – Число экземпляров товара. Тип числовой.

http статус ответа в случае успеха:

200 - OK

Пример запроса:

```
curl -X GET
'https://ismp.crpt.ru/api/v3/product/search?limit=10&offset=0&page=0&pre
fix=6401,6402,6403,6404,6405' -H 'Authorization: Bearer <>'
:
< HTTP/1.1 200
< Server: nginx/1.14.0
< Date: Sun, 27 Jan 2019 18:10:33 GMT
< Content-Type: application/json;charset=UTF-8
< Transfer-Encoding: chunked
< Connection: keep-alive
< X-Content-Type-Options: nosniff
< X-XSS-Protection: 1; mode=block
< Cache-Control: no-cache, no-store, max-age=0, must-revalidate
< Pragma: no-cache
< Expires: 0
< X-Frame-Options: DENY
<

* Connection #0 to host ismp.crpt.ru left intact
{
  "results": [
    {
      "id": 30056,
      "name": "001079-11    , 16",
      "gtin": "04690257113712",
      "brand": "",
      "packageType": "BOX",
      "innerUnitCount": 1,
      "model": "001079-11",
      "publicationDate": 1537304400000,
      "inn": "5011017647",
      "country": "",
      "productTypeDesc": "",
      "color": "",
      "materialDown": "",
      "materialUpper": "",
      "materialLining": "",
      "productType": "310000013"
    },
    {
      "id": 30045,
      "name": "001079-12    , 16",
      "gtin": "04690257114078",
      "brand": "",
      "packageType": "BOX",
      "innerUnitCount": 1,
```

```
"model": "001079-12",
"publicationDate": 1537304400000,
"inn": "5011017647",
"country": "",
"productTypeDesc": "",
"color": "",
"materialDown": "",
"materialUpper": "",
"materialLining": "",
"productType": "310000013"
},
{
  "id": 30059,
  "name": "001079-12  , 18",
  "gtin": "04690257114108",
  "brand": "",
  "packageType": "BOX",
  "innerUnitCount": 1,
  "model": "001079-12",
  "publicationDate": 1537304400000,
  "inn": "5011017647",
  "country": "",
  "productTypeDesc": "",
  "color": "",
  "materialDown": "",
  "materialUpper": "",
  "materialLining": "",
  "productType": "310000013"
}
],
```

```
"total": 11210,  
"errorCode": null  
}
```

Метод получения информации о товаре по gtin

GET api/v3/product/info

Параметры запроса:

gtins – список GTIN.

http статус ответа в случае успеха:

200 - ОК

Пример запроса:

```
curl -X GET
'https://ismp.crpt.ru/api/v3/product/api/v3/product/info?gtins=046902571
13712,04690257113712' -H 'Authorization: Bearer <>'
:
< HTTP/1.1 200
< Server: nginx/1.14.0
< Date: Sun, 27 Jan 2019 18:10:33 GMT
< Content-Type: application/json;charset=UTF-8
< Transfer-Encoding: chunked
< Connection: keep-alive
< X-Content-Type-Options: nosniff
< X-XSS-Protection: 1; mode=block
< Cache-Control: no-cache, no-store, max-age=0, must-revalidate
< Pragma: no-cache
< Expires: 0
< X-Frame-Options: DENY
<

* Connection #0 to host ismp.crpt.ru left intact
{
  "errorCode": < >,
  "results": < > [
    {
      "id": <Id >, 30056,
      "name": < > "001079-11", 16",
      "brand": < >, "brand": "",
      "gtin": "<GTIN >, 04690257113712",
      "packageType": < >, "BOX",
      "inn":< >, "5011017647",
      "innerUnitCount": 1,
      "model": "string", "001079-11",
      "publicationDate": < >, 1537304400000,
      "country":< >, "",
      "productTypeDesc":< >, "",
      "color":<>, "",
      "materialDown": < >"",
      "materialUpper":< >, "",
      "materialLining":< >, "",
      "productType": "310000013",
    }
  ],
  "total": < >
}
```

Метод получения доверенности

GET /api/v3/delegation/{id}

http статус ответа в случае успеха:

200 - ОК

Параметры запроса

id - идентификатор доверенности. Доверенность должна быть не удалена.

Заголовки ответа:

Content-Type: application/json

Тело запроса:

Файл pdf - необязательное поле

Поле delegation:

```
{
  "participant_inn": "string",
  "participant_name": "string",
  "representative_name": "string",
  "serial_number": "string",
  "valid_from": "string",
  "valid_to": "string"
}
```

Тело ответа:

```
{
  "id": 0,
  "participant_inn": "string",
  "participant_name": "string",
  "representative_name": "string",
  "serial_number": "string",
  "valid_from": "string",
  "valid_to": "string"
}
```

Атрибутивный состав запроса и ответа:

id - уникальный идентификатор. Тип Long.

serial_number - Номер доверенности. Тип строковый.

participant_name - Имя участника. Тип строковый.

participant_inn - ИНН участника. Тип строковый.

valid_from - Период действия, с. Тип дата. Паттерн ууу-ММ-дд'T'HH:mm:ss.SSS'Z'

valid_to - Период действия, по. Тип дата. Паттерн ууу-ММ-дд'T'HH:mm:ss.SSS'Z'.

representative_name - Имя доверенного лица.

http статус ответа в случае успеха:

200 - ОК

Пример запроса:

```
curl -X GET "http://localhost:8131/api/v3/delegation/1" -H "accept: */*"
-H "authorization: Bearer <ТОКЕН>
```

Пример ответа:

```
{
  "id": 1,
  "serial_number": "12",
  "participant_name": " ",
  "participant_inn": "7724211288",
  "valid_from": "2018-09-01T09:12:12.000Z",
  "valid_to": "2019-09-01T09:12:12.000Z",
  "representative_name": " "
```

Метод поиска по доверенностям

GET /api/v3/delegation/search

http статус ответа в случае успеха:

200 - ОК

Заголовки ответа:

Content-Type: application/json

Параметры запроса:

```
limit -
page -
document_name -
order_column -
valid -
representative_name -
valid_to - ,
valid_from - ,
participant_inn -
participant_name -
serial_number -
```

Тело ответа:

```
{
  "last": true,
  "result": [
    {
      "id": 0,
      "participant_inn": "string",
      "participant_name": "string",
      "representative_name": "string",
      "serial_number": "string",
      "valid_from": "string",
      "valid_to": "string"
    }
  ],
  "total": 0
}
```

Атрибутивный состав ответа:

id - уникальный идентификатор. Тип Long.

serial_number - Номер доверенности. Тип строковый.

participant_name - Имя участника. Тип строковый.

participant_inn - ИНН участника. Тип строковый.

valid_from - Период действия, с. Тип дата. Паттерн ууу-MM-dd'T'HH:mm:ss.SSS'Z'

valid_to - Период действия, по. Тип дата. Паттерн ууу-MM-dd'T'HH:mm:ss.SSS'Z'.

representative_name - Имя доверенного лица.

http статус ответа в случае успеха:

200 - ОК

Пример запроса:

```
curl -X GET
"http://localhost:8131/api/v3/delegation/search?serialNumber=1" -H
"accept: */*" -H "authorization: Bearer <ТОКЕН>
```

Пример ответа:

```
{
  "result": [
    {
      "id": 1,
      "serial_number": "12",
      "participant_id": null,
      "participant_name": "string",
      "participant_inn": "7724211288",
      "valid_from": "2018-10-01T09:12:12.000Z",
      "valid_to": "2019-10-01T09:12:12.000Z",
      "representative_name": "string"
    },
    {
      "id": 2,
      "serial_number": "12",
      "participant_id": null,
      "participant_name": " ",
      "participant_inn": "7724211288",
      "valid_from": "2018-09-01T09:12:12.000Z",
      "valid_to": "2019-09-01T09:12:12.000Z",
      "representative_name": " "
    }
  ],
  "total": 2,
  "last": true
}
```

Метод получения PDF файла доверенности

GET /api/v3/delegation/{id}/pdf

http статус ответа в случае успеха:

200 - ОК

Параметры запроса:

id - уникальный идентификатор, удаляемой доверенности. Тип Long.

Заголовки ответа:


```
cache-control: no-cache, no-store, max-age=0, must-revalidate
content-disposition: inline; filename=delegation.pdf
content-type: application/pdf
date: Fri, 15 Mar 2019 12:46:44 GMT
expires: 0
pragma: no-cache
transfer-encoding: chunked
x-content-type-options: nosniff
x-frame-options: DENY
x-xss-protection: 1; mode=block
```

Тело ответа:

```
<PDF >
```

http статус ответа в случае успеха:

200 - ОК

Пример запроса:

```
curl -X GET "http://localhost:8131/api/v3/delegation/1/pdf" -H "accept:
*/*" -H "authorization: Bearer <ТОКЕН>
```

Пример ответа:

```
{
  "id": 1,
  "serial_number": "12",
  "participant_name": "string",
  "participant_inn": "7724211288",
  "valid_from": "2018-10-01T09:12:12.000Z",
  "valid_to": "2019-10-01T09:12:12.000Z",
  "representative_name": "string"
}
```

Приложение. Документ заказа КМ

Формат JSON

Пример:

```

{
  "contract_number": "021118",
  "contract_date": 1541030400000,
  "operator_contract_date": "01.01.2019",
  "production_type": "SELF_MADE",
  "contact_person": "__XML",
  "delivery_address": "",
  "products_information": [
    {
      "order_line_id": "39d5d3bd-497e-402d-bfb0-fee6f7517940",
      "gtin": "04670028681596",
      "tn_ved_code": "6401",
      "emission_type": "LOCAL",
      "code_quantity": "1",
      "code_type": "PRINTABLE",
      "serial_num_type": "OPERATOR",
      "serial_numbers": [
        "8coAIyEsQNizH"
      ],
      "marking_type": "PRODUCT_PACKAGE"
    }
  ],
  "registrar_id": 1,
  "label_template_id": "2dbae9e2-e5a2-492e-a987-edda887def58"
}

```

Атрибутивный состав:

contract_number - номер договора с эмитентом. Тип строковый.

contract_date - дата договора с эмитентом. Формат даты Unix time в ms. **contract_date** - дата договора с эмитентом. Формат даты Unix time в ms. Поле устаревшее, в следующих релизах поддерживаться не будет, требуется использовать поле **operator_contract_date**

operator_contract_date - дата договора с эмитентом. Формат даты строковый: 'DD.MM.YYY'.

production_type - способ изготовления СИ. Возможные значения "SELF_MADE" - самостоятельно, СЕМ - центр этикетирования и маркировки. Тип строковый.

contact_person - контактное лицо. Тип строковый.

delivery_address - адрес доставки. Тип числовой.

order_line_id - уникальный номер подзаказа. Тип строковый.

gtin - 14-ти значный GS1 код товара Если код товара меньше 14 символов, то его необходимо дополнить ведущими нулями. Тип строковый.

tn_ved_code - 4-х код ТН ВЭД. Тип строковый.

emission_type - способ выпуска товара. Возможные значения "LOCAL" - производство в РФ, "FOREIGN" - импорт. Тип строковый.

code_quantity - количество КМ в подзаказе. Тип числовой.

code_type - вид СИ. Возможные значения "PRINTABLE" - печатаемый, "GLUEABLE" - клеевой, "MOUNTABLE" - навесной. Тип строковый.

serial_num_type - способ формирования серийного номера. Возможные значения "OPERATOR" - генерируется оператором, "SELF_MADE" - самостоятельно (не рекомендованно к использованию). Тип строковый.

marking_type - вид маркировки. Возможные значения "PRODUCT_PACKAGE" - на потребительскую упаковку, PRODUCT - на товар, PRODUCT_LABEL - на товарный ярлык. Тип строковый.

serial_numbers - Список серийных номеров. Тип строковый.

Формат XML

xsd схема mkio_order.xsd:

```
<?xml version="1.0" encoding="UTF-8"?>
<xs:schema elementFormDefault="qualified"
xmlns:xs="http://www.w3.org/2001/XMLSchema">
  <xs:include schemaLocation="LP_base_types.xsd"/>
  <xs:element name="MKIO_Order" type="MKIO_Order"/>
  <xs:complexType name="MKIO_Order">
    <xs:annotation>
      <xs:documentation> </xs:documentation>
    </xs:annotation>
    <xs:sequence>
      <xs:element name="TRADE_PARTICIPANT_INN"
type="TRADE_PARTICIPANT_INN_type">
        <xs:annotation>
          <xs:documentation> </xs:documentation>
        </xs:annotation>
      </xs:element>
      <xs:element name="OPERATOR_CONTRACT_NUM"
type="document_number_200_type">
        <xs:annotation>
          <xs:documentation> </xs:documentation>
        </xs:annotation>
      </xs:element>
      <xs:element name="OPERATOR_CONTRACT_DATE" type="date_type">
        <xs:annotation>
          <xs:documentation> </xs:documentation>
        </xs:annotation>
      </xs:element>
      <xs:element name="PREP_METHOD">
        <xs:annotation>
          <xs:documentation>
            1 -
            2 -
          </xs:documentation>
        </xs:annotation>
        <xs:simpleType>
          <xs:restriction base="xs:int">
            <xs:enumeration value="1"/>
            <xs:enumeration value="2"/>
          </xs:restriction>
        </xs:simpleType>
      </xs:element>
      <xs:element name="RECEPTION_METHOD">
        <xs:annotation>
          <xs:documentation>
```

```

        1 -
        2 -
    </xs:documentation>
</xs:annotation>
<xs:simpleType>
    <xs:restriction base="xs:int">
        <xs:enumeration value="1"/>
        <xs:enumeration value="2"/>
    </xs:restriction>
</xs:simpleType>
</xs:element>
<xs:element name="LABEL_TEMPLATE_ID" type="guid_type">
    <xs:annotation>
        <xs:documentation>ID </xs:documentation>
    </xs:annotation>
</xs:element>
<xs:element name="LABELLING_CENTER_NAME" minOccurs="0">
    <xs:annotation>
        <xs:documentation> </xs:documentation>
    </xs:annotation>
    <xs:simpleType>
        <xs:restriction base="xs:string">
            <xs:maxLength value="1000"/>
        </xs:restriction>
    </xs:simpleType>
</xs:element>
<xs:element name="LABELLING_CENTER_INN"
type="LABELLING_CENTER_INN_type" minOccurs="0">
    <xs:annotation>
        <xs:documentation> </xs:documentation>
    </xs:annotation>
</xs:element>
<xs:element name="LABELLING_CENTER_CONTRACT_NUM"
minOccurs="0">
    <xs:annotation>
        <xs:documentation> </xs:documentation>
    </xs:annotation>
    <xs:simpleType>
        <xs:restriction base="xs:string">
            <xs:maxLength value="1000"/>
        </xs:restriction>
    </xs:simpleType>
</xs:element>
<xs:element name="LABELLING_CENTER_CONTRACT_DATE"
type="date_type" minOccurs="0">
    <xs:annotation>
        <xs:documentation> </xs:documentation>
    </xs:annotation>
</xs:element>
<xs:element name="CONTACT">

```



```

base="xs:string">
    <xs:length value="4"/>
    <xs:pattern
value="[0-9]{4}"/>
    </xs:restriction>
    </xs:simpleType>
</xs:element>
<xs:element name="RELEASE_METHOD">
    <xs:annotation>
        <xs:documentation>
            1 -
            2 -
        </xs:documentation>
    </xs:annotation>
    <xs:simpleType>
        <xs:restriction
base="xs:int">
            <xs:enumeration
value="1"/>
            <xs:enumeration
value="2"/>
        </xs:restriction>
    </xs:simpleType>
</xs:element>
<xs:element name="MKIO_QUANTITY"
type="xs:long" minOccurs="0">
    <xs:annotation>
        <xs:documentation> . "
            s" 1 -
            .
        </xs:documentation>
    </xs:annotation>
</xs:element>
<xs:element name="IO_TYPE">
    <xs:annotation>
        <xs:documentation>
            1 -
            2 -
            3 -
        </xs:documentation>
    </xs:annotation>
    <xs:simpleType>
        <xs:restriction
base="xs:int">
            <xs:enumeration
value="1"/>
            <xs:enumeration
value="2"/>
            <xs:enumeration
value="3"/>

```

```

        </xs:restriction>
    </xs:simpleType>
</xs:element>
<xs:element name="S_ORG_METHOD">
    <xs:annotation>
        <xs:documentation>          s
            1 -
            2 -
        </xs:documentation>
    </xs:annotation>
    <xs:simpleType>
        <xs:restriction
            <xs:enumeration
            <xs:enumeration

        </xs:restriction>
    </xs:simpleType>
</xs:element>
<xs:element name="MARKING_TYPE">
    <xs:annotation>
        <xs:documentation>
            1 -
            2 -
            3 -
        </xs:documentation>
    </xs:annotation>
    <xs:simpleType>
        <xs:restriction
            <xs:enumeration
            <xs:enumeration
            <xs:enumeration

        </xs:restriction>
    </xs:simpleType>
</xs:element>
<xs:element name="S_NUMS"
    minOccurs="0">
    <xs:annotation>
        <xs:documentation> .
            "          s" 1 -
            .
        </xs:documentation>
    </xs:annotation>
    <xs:complexType>
        <xs:sequence>

```

```

name="SERIAL_NUMBER" maxOccurs="unbounded">
    <xs:element
        <xs:annotation>
            </xs:annotation>
            <xs:simpleType>
                <xs:restriction
                    base="xs:string">
                        <xs:length
                            value="13"/>
                        <xs:pattern
                            value="[a-zA-Z0-9]{13}"/>
                    </xs:restriction>
                </xs:simpleType>
            </xs:element>
        </xs:sequence>
    </xs:complexType>
</xs:element>
</xs:sequence>
</xs:complexType>
<xs:unique name="PRODUCT_DETAILS_unique">
    <xs:selector xpath="PRODUCT_DETAILS"/>
    <xs:field xpath="GTIN"/>
    <xs:field xpath="TNVED_CODE"/>
    <xs:field xpath="SERIAL_NUMBER"/>
</xs:unique>
</xs:element>
</xs:sequence>
<xs:attribute name="action_id" type="xs:int" use="required"

```



```

fixed="01" />
  </xs:complexType>
</xs:schema>

```

xsd cxema LP_base_types.xsd:

```

<?xml version="1.0" encoding="UTF-8"?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema"
elementFormDefault="qualified" attributeFormDefault="unqualified">
  <xs:simpleType name="IPv4Address_type">
    <xs:annotation>
      <xs:documentation>IPv4.          ( 0 255), .
</xs:documentation>
    </xs:annotation>
    <xs:restriction base="xs:string">
      <xs:pattern
value="((1?[0-9]?[0-9]|2[0-4][0-9]|25[0-5])\.){3}(1?[0-9]?[0-9]|2[0-4][0
-9]|25[0-5])"/>
    </xs:restriction>
  </xs:simpleType>
  <xs:simpleType name="IPv6Address_type">
    <xs:annotation>
      <xs:documentation>IPv6.          ( ), . </xs:documentation>
    </xs:annotation>
    <xs:restriction base="xs:string">
      <xs:pattern value="([A-Fa-f0-9]{1,4}:){7}[A-Fa-f0-9]{1,4}"
/>
    </xs:restriction>
  </xs:simpleType>
  <xs:simpleType name="string200_type">
    <xs:annotation>
      <xs:documentation> ,      200 </xs:documentation>
    </xs:annotation>
    <xs:restriction base="xs:string">
      <xs:minLength value="1"/>
      <xs:maxLength value="200"/>
    </xs:restriction>
  </xs:simpleType>
  <xs:simpleType name="string1000_type">
    <xs:annotation>
      <xs:documentation>      1000 </xs:documentation>
    </xs:annotation>
    <xs:restriction base="xs:string">
      <xs:minLength value="1"/>
      <xs:maxLength value="1000"/>
    </xs:restriction>
  </xs:simpleType>

```

```

<xs:simpleType name="TRADE_PARTICIPANT_INN_type">
  <xs:annotation>
    <xs:documentation> </xs:documentation>
  </xs:annotation>
  <xs:restriction base="xs:string">
    <xs:minLength value="10"/>
    <xs:maxLength value="12"/>
    <xs:pattern
value="((([0-9]{1}[1-9]{1}|[1-9]{1}[0-9]{1})[0-9]{8})|((([0-9]{1}[1-9]{1}|
[1-9]{1}[0-9]{1})[0-9]{10}))/>
    </xs:restriction>
  </xs:simpleType>
  <xs:simpleType name="LABELLING_CENTER_INN_type">
    <xs:annotation>
      <xs:documentation> </xs:documentation>
    </xs:annotation>
    <xs:restriction base="xs:string">
      <xs:length value="10"/>
      <xs:pattern
value="([0-9]{1}[1-9]{1}|[1-9]{1}[0-9]{1})[0-9]{8}"/>
      </xs:restriction>
    </xs:simpleType>
    <xs:simpleType name="date_type">
      <xs:annotation>
        <xs:documentation> ..</xs:documentation>
      </xs:annotation>
      <xs:restriction base="xs:string">
        <xs:length value="10"/>
        <xs:pattern
value="(0[1-9]{1}|[1-2]{1}[0-9]{1}|3[0-1]{1})\.(0[1-9]{1}|1[0-2]{1})\.(1
[0-9]{3}|20[0-9]{2}))/>
        </xs:restriction>
      </xs:simpleType>
      <xs:complexType name="fias_address_type">
        <xs:annotation>
          <xs:documentation> </xs:documentation>
        </xs:annotation>
        <xs:sequence>
          <xs:element name="aoguid">
            <xs:annotation>
              <xs:documentation> </xs:documentation>
            </xs:annotation>
            <xs:simpleType>
              <xs:restriction base="xs:string">
                <xs:length value="36"/>
                <xs:pattern
value="([a-fA-F0-9]{8}-[a-fA-F0-9]{4}-[a-fA-F0-9]{4}-[a-fA-F0-9]{4}-[a-f
A-F0-9]{12}))/>
                </xs:restriction>
              </xs:simpleType>
            </xs:element>
          </xs:sequence>
        </xs:complexType>
      </xs:simpleType>
    </xs:simpleType>
  </xs:sequence>
</xs:simpleType>

```

```

</xs:element>
<xs:element name="houseguid" minOccurs="0">
  <xs:annotation>
    <xs:documentation> </xs:documentation>
    <xs:documentation> </xs:documentation>
  </xs:annotation>
  <xs:simpleType>
    <xs:restriction base="xs:string">
      <xs:length value="36"/>
      <xs:pattern
value="([a-zA-F0-9]{8}-[a-zA-F0-9]{4}-[a-zA-F0-9]{4}-[a-zA-F0-9]{4}-[a-f
A-F0-9]{12})"/>
    </xs:restriction>
  </xs:simpleType>
</xs:element>
<xs:element name="flat" minOccurs="0">
  <xs:annotation>
    <xs:documentation></xs:documentation>
    <xs:documentation> </xs:documentation>
  </xs:annotation>
  <xs:simpleType>
    <xs:restriction base="xs:string">
      <xs:minLength value="1"/>
      <xs:maxLength value="20"/>
    </xs:restriction>
  </xs:simpleType>
</xs:element>
</xs:sequence>
</xs:complexType>
<xs:simpleType name="datetimeoffset">
  <xs:restriction base="xs:dateTime">
    <xs:minInclusive value="0001-01-01T00:00:00Z"/>
    <xs:pattern
value="((00[1-9])|(00[1-9][0-9])|(0[1-9][0-9]{2})|([1-9][0-9]{3}))-(0[
1-9])|(1[012]))-(0[1-9])|([12][0-9])|(3[01]))T((01[0-9])|(2[0-3]))(:[
0-5][0-9]){2}(\.[0-9]+)?(((+)((0[0-9])|(1[0-3]))(:[0-5][0-9])|14:00
)|Z)"/>
  </xs:restriction>
</xs:simpleType>
<xs:simpleType name="document_number_200_type">
  <xs:annotation>
    <xs:documentation> , 200 </xs:documentation>
  </xs:annotation>
  <xs:restriction base="xs:string">
    <xs:minLength value="1"/>
    <xs:maxLength value="200"/>
  </xs:restriction>
</xs:simpleType>
<xs:simpleType name="turnover_type_enum">
  <xs:annotation>

```

```

        <xs:documentation>
            1 - -
            2 -
            3 -
            4 -
        </xs:documentation>
    </xs:annotation>
    <xs:restriction base="xs:int">
        <xs:enumeration value="1"/>
        <xs:enumeration value="2"/>
        <xs:enumeration value="3"/>
        <xs:enumeration value="4"/>
    </xs:restriction>
</xs:simpleType>
<xs:simpleType name="gsl_uit_type">
    <xs:annotation>
        <xs:documentation> (31 38 )</xs:documentation>
    </xs:annotation>
    <xs:restriction base="xs:string">
        <xs:minLength value="31"/>
        <xs:maxLength value="38"/>
        <xs:pattern value="[A-Za-z0-9]{31}|[A-Za-z0-9]{38}"/>
    </xs:restriction>
</xs:simpleType>
<xs:simpleType name="gsl_uitu_type">
    <xs:annotation>
        <xs:documentation> </xs:documentation>
    </xs:annotation>
    <xs:restriction base="xs:string">
        <xs:length value="18"/>
        <xs:pattern value="[A-Za-z0-9]{18}"/>
    </xs:restriction>
</xs:simpleType>
<xs:simpleType name="price_type">
    <xs:annotation>
        <xs:documentation>/</xs:documentation>
    </xs:annotation>
    <xs:restriction base="xs:decimal">
        <xs:totalDigits value="19"/>
        <xs:fractionDigits value="2"/>
        <xs:minInclusive value="0"/>
    </xs:restriction>
</xs:simpleType>

<xs:simpleType name="guid_type">
    <xs:annotation>
        <xs:documentation>
            GUID
        </xs:documentation>
    </xs:annotation>

```

```

    <xs:restriction base="xs:string">
      <xs:length value="36"/>
      <xs:pattern
value="([a-zA-F0-9]{8}-[a-zA-F0-9]{4}-[a-zA-F0-9]{4}-[a-zA-F0-9]{4}-[a-zA-F0-9]{12})"/>
    </xs:restriction>
  </xs:simpleType>
  <xs:simpleType name="token_type">
    <xs:annotation>
      <xs:documentation>

          </xs:documentation>
        </xs:annotation>
        <xs:restriction base="xs:string">
          <xs:length value="15"/>
          <xs:pattern value="([A-Za-z0-9]{15})"/>
        </xs:restriction>
      </xs:simpleType>
      <xs:simpleType name="action_type">
        <xs:annotation>
          <xs:documentation>
            1 -
            2 -
            3 -
            4 -
            5 -
            6 -
            7 - ( )
            8 - /
            9 -
            10 -
            11 -
            12 -
          </xs:documentation>
        </xs:annotation>
        <xs:restriction base="xs:int">
          <xs:enumeration value="1"/>
          <xs:enumeration value="2"/>
          <xs:enumeration value="3"/>
          <xs:enumeration value="4"/>
          <xs:enumeration value="5"/>
          <xs:enumeration value="6"/>
          <xs:enumeration value="7"/>
          <xs:enumeration value="8"/>
          <xs:enumeration value="9"/>
          <xs:enumeration value="10"/>
          <xs:enumeration value="11"/>
          <xs:enumeration value="12"/>
        </xs:restriction>
      </xs:simpleType>

```

```
<xs:simpleType name="doc_type_enum">
  <xs:annotation>
    <xs:documentation>
      1 -
      2 -   ()
      3 -
      4 -
      5 -
      6 - -
      7 -
    </xs:documentation>
  </xs:annotation>
  <xs:restriction base="xs:int">
    <xs:enumeration value="1"/>
    <xs:enumeration value="2"/>
    <xs:enumeration value="3"/>
    <xs:enumeration value="4"/>
    <xs:enumeration value="5"/>
    <xs:enumeration value="6"/>
    <xs:enumeration value="7"/>
  </xs:restriction>
</xs:simpleType>
<xs:simpleType name="email_address">
  <xs:annotation>
    <xs:documentation> </xs:documentation>
  </xs:annotation>
  <xs:restriction base="xs:string">
    <xs:pattern
value="([0-9a-zA-Z]([-.\w]*[0-9a-zA-Z])*@[0-9a-zA-Z]([-\\w]*[0-9a-zA-Z]\.
)+[a-zA-Z]{2,9})"/>
```

```
    </xs:restriction>
  </xs:simpleType>
</xs:schema>
```

Пример:

```
<MKIO_Order action_id="10">
  <TRADE_PARTICIPANT_INN>7724211288</TRADE_PARTICIPANT_INN>
  <OPERATOR_CONTRACT_NUM>2123</OPERATOR_CONTRACT_NUM>
  <OPERATOR_CONTRACT_DATE>14.10.2018</OPERATOR_CONTRACT_DATE>
  <PREP_METHOD>1</PREP_METHOD>

  <LABEL_TEMPLATE_ID>c30ae458-f9fa-4111-b932-d72928638b8b</LABEL_TEMPLATE_ID>
  <CONTACT></CONTACT>
  <DELIVERY_ADDRESS></DELIVERY_ADDRESS>
  <PRODUCT_DETAILS>
    <PRODUCT>
      <GTIN>04630037591316</GTIN>
      <TNVED_CODE>6405</TNVED_CODE>
      <RELEASE_METHOD>1</RELEASE_METHOD>
      <MKIO_QUANTITY>2</MKIO_QUANTITY>
      <IO_TYPE>1</IO_TYPE>
      <S_ORG_METHOD>1</S_ORG_METHOD>
      <S_NUMS>
        <SERIAL_NUMBER>8coAIyEsQNiZH</SERIAL_NUMBER>
        <SERIAL_NUMBER>NqCPkoGYc6BsN</SERIAL_NUMBER>
      </S_NUMS>
      <MARKING_TYPE>1</MARKING_TYPE>
    </PRODUCT>
  </PRODUCT_DETAILS>
</MKIO_Order>
```

S_NUM - Список серийных номеров через запятую

Формат CSV

Пример:

```
  ;      ;      ; ; ; ; ; ;      ;      ; ;
7724211288;12345;2018-06-13;1;;2bc368af-3bdf-4abd-8df7-ae84498c5c00;;;;;
; ;

GTIN;      ;      ; ; ;      s;      s;
04059155704529;6405;1;10;1;2;;1
```

Ролевая модель

Добавление подписанной заявки на добавление роли пользователя

POST api/v3/lk/user/role/

```
{
  "product_document": "< base64> ",
  "document_format": "< > ",
  "signature": "< base64> "
}
```

Атрибутивный состав запроса::

product_document - документ о приемке, закодированный в base64. Описание документа приведено в п. "Приложение. Документ по созданию роли"

document_format - формат документа о приемке, возможные значения "MANUAL" - json, "XML" - xml. Тип строковый.

signature - открепленная УКЭП закодированный в base64. Документ необходимо подписывать до перевода его в base64. Тип строковый.

Пример запроса:

```
curl -X POST "http://lk-doc-api.apps.qa.lp.crpt.tech/api/v3/lk/user/role/" -H "accept: */*" -H "Content-Type: application/json" -d
"{\"document_format\": \"MANUAL\", \"product_document\": \"<Документ в base64>\", \"signature\": \"<Открепленная подпись в base64>\"}"
```

http статус ответа в случае успеха:

200 - OK

Заголовки ответа:

Content-Type: application/json

Тело ответа:

```
{
  "string": "<document_id>"
}
```

Добавление подписанной заявки на изменение роли пользователя

PUT api/v3/lk/user/role/

```
{
  "product_document": "< base64> ",
  "document_format": "< > ",
  "signature": "< base64> "
}
```

Атрибутивный состав запроса::

product_document - документ о приемке, закодированный в base64. Описание документа приведено в п. "Приложение. Документ по изменению роли"

document_format - формат документа о приемке, возможные значения "MANUAL" - json, "XML" - xml. Тип строковый.

signature - открепленная УКЭП закодированный в base64. Документ необходимо подписывать до перевода его в base64. Тип строковый.

Пример запроса:

```
curl -X PUT "http://lk-doc-api.apps.qa.lp.crpt.tech/api/v3/lk/user/role/" -H "accept: */*" -H "Content-Type: application/json" -d {"document_format": "MANUAL", "product_document": "<Документ в base64>", "signature": "<Открепленная подпись в base64>"}
```

http статус ответа в случае успеха:

200 - ОК

Заголовки ответа:

Content-Type: application/json

Тело ответа:

```
{
  "string": "<document_id>"
}
```

Добавление подписанной заявки на удаление роли пользователя

DELETE api/v3/lk/user/role/

```
{
  "product_document": "< base64>",
  "document_format": "< >",
  "signature": "< base64>"
}
```

Атрибутивный состав запроса::

product_document - документ о приемке, закодированный в base64. Описание документа приведено в п. "Приложение. Документ по удалению роли"

document_format - формат документа о приемке, возможные значения "MANUAL" - json, "XML" - xml. Тип строковый.

signature - открепленная УКЭП закодированный в base64. Документ необходимо подписывать до перевода его в base64. Тип строковый.

Пример запроса:

```
curl -X DELETE "http://lk-doc-api.apps.qa.lp.crpt.tech/api/v3/lk/user/role/" -H "accept: */*" -H "Content-Type: application/json" -d {"document_format": "MANUAL", "product_document": "<Документ в base64>", "signature": "<Открепленная подпись в base64>"}
```

http статус ответа в случае успеха:

200 - ОК

Заголовки ответа:

Content-Type: application/json

Тело ответа:

```
{
  "string": "<document_id>"
}
```

Добавление подписанной заявки на присвоению роли пользователю

PATCH api/v3/lk/user/role/

```
{
  "product_document": "< base64>",
  "document_format": "< >",
  "signature": "< base64>"
}
```

Атрибутивный состав запроса::

product_document - документ о приемке, закодированный в base64. Описание документа приведено в п. "Приложение. Документ по присвоению роли"

document_format - формат документа о приемке, возможные значения "MANUAL" - json, "XML" - xml. Тип строковый.

signature - открепленная УКЭП закодированный в base64. Документ необходимо подписывать до перевода его в base64. Тип строковый.

Пример запроса:

```
curl -X PATCH "http://lk-doc-api.apps.qa.lp.crpt.tech/api/v3/lk/user/role/" -H "accept: */*" -H "Content-Type: application/json" -d
{"document_format": "MANUAL", "product_document": "<Документ в base64>", "signature": "<Открепленная подпись в base64>"}
```

http статус ответа в случае успеха:

200 - ОК

Заголовки ответа:

Content-Type: application/json

Тело ответа:

```
{
  "string": "<document_id>"
}
```

Получение списка ролей участника

GET /api/v3/auth/user-role/search

Параметры запроса:

id - Идентификатор роли. Тип числовой.

name - Наименование роли. тип строковый.

organisationInn - ИНН участника. Тип строковый.

organisationId - Идентификатор участника. Тип числовой.

createDateFrom - Дата создания, с. Пример 2019-04-19T13:39:16.125Z

createDateTo - Дата создания, по. Пример 2019-04-19T13:39:16.125Z

orgRoleId - Идентификатор роли участника. Тип числовой.

orderColumn - Поле для сортировки. Тип строковый.

orderDirection - Тип сортировки. ASC/DESC

limit - Размер раздела. Тип числовой.

page - Номер раздела. Тип числовой.

```
{
  "last": true,
  "result": [
    {
      "create_date": "2019-04-19T13:39:16.125Z",
      "description": "string",
      "domain_objects": [
        {
          "description": "string",
          "entries": [
            "READ"
          ],
          "id": 0,
          "name": "string"
        }
      ],
      "id": 0,
      "name": "string",
      "parent_role_id": 0
    }
  ],
  "total": 0
}
```

Пример запроса:

```
curl -X GET
"http://localhost:8080/api/v3/auth/user-role/search?organisationInn=12"
-H "accept: */*" -H "authorization: Bearer <TOKEN>"
```

Регистрация доменных объектов

POST /api/v3/auth/domain/

Тело запроса:

```
{
  "description": " ",
  "entries": ["CREATE"],
  "name": "CRPT-LK-DOC-API.BLOCKING-CONTROLLER"
}
```

Атрибутивный состав запроса:

description - Описание доменного объекта. Тип строковый. Необязательное

entries - Набор входов. Тип строковый массив. Не меньше одного

name - Наименование доменного объекта. Тип строковый. Обязательное.

Тело запроса:

```
{
  "description": "string",
  "entries": ["READ"],
  "id": 0,
  "name": "string"
}
```

Пример запроса:

```
curl -X POST "http://localhost:8080/api/v3/auth/domain/" -H "accept: */*" -H "authorization: Bearer <ТОКЕН>" -H "Content-Type: application/json" -d "{ \"description\": \"string\", \"entries\": [ \"READ\" ], \"id\": 0, \"name\": \"string\"}"
```

Обновление доменных объектов

PUT /api/v3/auth/domain/

Тело запроса:

```
{
  "id": 0,
  "description": " ",
  "entries": ["CREATE"],
  "name": "CRPT-LK-DOC-API.BLOCKING-CONTROLLER"
}
```

Атрибутивный состав запроса:

id - Идентификатор доменного объекта.

description - Описание доменного объекта. Тип строковый. Необязательное

entries - Набор входов. Тип строковый массив. Не меньше одного

name - Наименование доменного объекта. Тип строковый. Обязательное.

Пример запроса:

```
curl -X PUT "http://localhost:8080/api/v3/auth/domain/" -H "accept: */*"
-H "authorization: Bearer <ТОКЕН>" -H "Content-Type: application/json"
-d "{\"description\": \"string\", \"entries\": [ \"READ\" ], \"id\": 0,
\"name\": \"string\"}"
```

Удаление доменных объектов

```
DELETE /api/v3/auth/domain/{id}
```

параметр запроса - id - идентификатор доменного объекта

Пример запроса:

```
curl -X DELETE "http://localhost:8080/api/v3/auth/domain/12" -H "accept:
*/*" -H "authorization: Bearer <ТОКЕН>"
```

Приложение. Документ по созданию роли

Формат JSON

Пример JSON:

```
{
  "sender_inn": "458955694527",
  "user_role"
    "description": "",
    "name": "Shippment",
    "domain_objects": [
      {
        "name": "CRPT-LK-DOC-API.SHIPMENT-REQUEST-CONTROLLER",
        "description": " ",
        "entries": [ "CREATE" ]
      },
      {
        "name": "CRPT-FACADE.DOC-CONTROLLER.LIST.SHIPMENT",
        "description": " ",
        "entries": [ "READ" ]
      }
    ]
}
```

Атрибутивный состав

sender_inn - ИНН отправителя. Обязательное

user_role :

description - Описание роли. Тип строковый. Необязательное.

name - Наименование роли. Тип строковый. Обязательное.

domain_objects :

name - Наименование доменного объекта. Тип строковый. Обязательное.

description - Описание доменного объекта. Тип строковый. Обязательное.

entries - маска объекта. Тип строковый. Не меньше одного. Возможные значения: READ, CREATE, WRITE, DELETE, ADMINISTRATION, DOWNLOAD

Формат XML

Пример XML:

```
<create_user_role action_id="3">
  <sender_inn>7703799090</sender_inn>
  <user_role>
    <description>Role description</description>
    <name>Role name</name>
    <domain_objects>
      <name>Domain name</name>
      <description>What it does</description>
      <entries>WRITE</entries>
      <entries>READ</entries>
    </domain_objects>
  </user_role>
</create_user_role>
```

Схема XSD:

```

<?xml version="1.0" encoding="windows-1251"?>
<xs:schema attributeFormDefault="unqualified"
elementFormDefault="qualified"
xmlns:xs="http://www.w3.org/2001/XMLSchema">
  <xs:include schemaLocation="../LP_base_types.xsd"/>
  <xs:element name="create_user_role">
    <xs:complexType>
      <xs:sequence>
        <xs:element type="TRADE_PARTICIPANT_INN_type" name="sender_inn">
          <xs:annotation>
            <xs:documentation> </xs:documentation>
          </xs:annotation>
        </xs:element>
        <xs:element name="user_role">
          <xs:complexType>
            <xs:sequence>
              <xs:element type="xs:string" name="description" minOccurs="0"/>
              <xs:element type="xs:string" name="name"/>
              <xs:element name="domain_objects" maxOccurs="unbounded">
                <xs:complexType>
                  <xs:sequence>
                    <xs:element type="xs:string" name="name"/>
                    <xs:element type="xs:string" name="description" minOccurs="0"/>
                    <xs:element type="xs:string" name="entries"
maxOccurs="unbounded"/>
                  </xs:sequence>
                </xs:complexType>
              </xs:element>
            </xs:sequence>
          </xs:complexType>
        </xs:element>
      </xs:sequence>
    </xs:complexType>
    <xs:attribute name="action_id" type="xs:int" use="required"
fixed="3"/>
  </xs:element>
</xs:schema>

```

Приложение. Документ по изменению роли

Формат JSON

Пример JSON:

```

{
  "sender_inn": "458955694527",
  "user_role"
    "id": 100,
    "description": "",
    "name": "Shippment",
    "domain_objects": [
      {
        "name": "CRPT-LK-DOC-API.SHIPMENT-REQUEST-CONTROLLER",
        "description": " ",
        "entries": [ "CREATE" ]
      },
      {
        "name": "CRPT-FACADE.DOC-CONTROLLER.LIST.SHIPMENT",
        "description": " ",
        "entries": [ "READ" ]
      }
    ]
  }
}

```

Атрибутивный состав

sender_inn - ИНН отправителя, Обязательное

user_role :

id - Идентификатор роли. Обязательное

description - Описание роли. Тип строковый. Необязательное.

name - Наименование роли. Тип строковый. Обязательное.

domain_objects :

name - Наименование доменного объекта. Тип строковый. Обязательное.

description - Описание доменного объекта. Тип строковый. Обязательное.

entries - маска объекта. Тип строковый. Не меньше одного. READ, CREATE, WRITE, DELETE, ADMINISTRATION, DOWNLOAD

Формат XML

Пример XML:


```
<create_user_role action_id="15">
  <sender_inn>7703799090</sender_inn>
  <user_role>
    <id>100</id>
    <description>Role description</description>
    <name>Role name</name>
    <domain_objects>
      <name>Domain name</name>
      <description>What it does</description>
      <entries>WRITE</entries>
      <entries>READ</entries>
    </domain_objects>
  </user_role>
</create_user_role>
```

Cxema XSD:

```

<?xml version="1.0" encoding="windows-1251"?>
<xs:schema attributeFormDefault="unqualified"
elementFormDefault="qualified"
xmlns:xs="http://www.w3.org/2001/XMLSchema">
  <xs:include schemaLocation="../LP_base_types.xsd"/>
  <xs:element name="create_user_role">
    <xs:complexType>
      <xs:sequence>
        <xs:element type="TRADE_PARTICIPANT_INN_type" name="sender_inn">
          <xs:annotation>
            <xs:documentation> </xs:documentation>
          </xs:annotation>
        </xs:element>
        <xs:element name="user_role">
          <xs:complexType>
            <xs:sequence>
              <xs:element type="xs:long" name="id" minOccurs="0"/>
              <xs:element type="xs:string" name="description" minOccurs="0"/>
              <xs:element type="xs:string" name="name"/>
              <xs:element name="domain_objects" maxOccurs="unbounded">
                <xs:complexType>
                  <xs:sequence>
                    <xs:element type="xs:string" name="name"/>
                    <xs:element type="xs:string" name="description" minOccurs="0"/>
                    <xs:element type="xs:string" name="entries"
maxOccurs="unbounded"/>
                  </xs:sequence>
                </xs:complexType>
              </xs:element>
            </xs:sequence>
          </xs:complexType>
        </xs:element>
      </xs:sequence>
    </xs:complexType>
    <xs:attribute name="action_id" type="xs:int" use="required"
fixed="3"/>
  </xs:element>
</xs:schema>

```

Приложение. Документ по удалению роли

Формат JSON

Пример JSON:

```
{
  "sender_inn": "458955694527",
  "user_role" {
    "id": 100,
  }
}
```

Атрибутивный состав

sender_inn - ИНН отправителя. Обязательное

user_role :

id - Идентификатор роли. Обязательное

Формат XML

Пример XML:

```
<create_user_role action_id="15">
  <sender_inn>7703799090</sender_inn>
  <user_role>
    <id>100</id>
  </user_role>
</create_user_role>
```

Схема XSD:

```

<?xml version="1.0" encoding="windows-1251"?>
<xs:schema attributeFormDefault="unqualified"
elementFormDefault="qualified"
xmlns:xs="http://www.w3.org/2001/XMLSchema">
  <xs:include schemaLocation="../LP_base_types.xsd"/>
  <xs:element name="create_user_role">
    <xs:complexType>
      <xs:sequence>
        <xs:element type="TRADE_PARTICIPANT_INN_type" name="sender_inn">
          <xs:annotation>
            <xs:documentation> </xs:documentation>
          </xs:annotation>
        </xs:element>
        <xs:element name="user_role">
          <xs:complexType>
            <xs:sequence>
              <xs:element type="xs:long" name="id" minOccurs="0"/>
            </xs:sequence>
          </xs:complexType>
        </xs:element>
      </xs:sequence>
      <xs:attribute name="action_id" type="xs:int" use="required"
fixed="3"/>
    </xs:complexType>
  </xs:element>
</xs:schema>

```

Приложение. Документ по присвоению роли

Формат JSON

Пример JSON:

```

{
  "sender_inn": "458955694527",
  "assignee_user_ids": [1, 2],
  "user_role_id": 1
}

```

Атрибутивный состав:

sender_inn - Инн отправителя. Тип строковый. Обязательное.

assignee_user_ids - Множество идентификаторов пользователей для присвоения роли. Тип числовой. Не меньше одного.

user_role_id - Идентификатор роли для присвоения. Тип числовой. Обязательное.

Формат XML

Пример XML:

```
<?xml version='1.0' encoding='UTF-8' standalone='yes'?>
<assign_user_role action_id="3.1" version="1">
  <sender_inn>458955694527</sender_inn>
  <assignee_user_id>1</assignee_user_id>
  <user_role_id>1</user_role_id>
</assign_user_role>
```

Схема XSD:

```

<?xml version="1.0" encoding="windows-1251"?>
<xs:schema attributeFormDefault="unqualified"
elementFormDefault="qualified"
xmlns:xs="http://www.w3.org/2001/XMLSchema">
  <xs:include schemaLocation="../LP_base_types.xsd"/>
  <xs:element name="assign_user_role">
    <xs:complexType>
      <xs:sequence>
        <xs:element type="TRADE_PARTICIPANT_INN_type"
name="sender_inn">
          <xs:annotation>
            <xs:documentation> </xs:documentation>
          </xs:annotation>
        </xs:element>
        <xs:element name="assignee_user_id">
          <xs:annotation>
            <xs:documentation>ID </xs:documentation>
          </xs:annotation>
          <xs:simpleType>
            <xs:restriction base="xs:long">
              <xs:minInclusive value="1"/>
            </xs:restriction>
          </xs:simpleType>
        </xs:element>
        <xs:element name="user_role_id">
          <xs:annotation>
            <xs:documentation>ID </xs:documentation>
          </xs:annotation>
          <xs:simpleType>
            <xs:restriction base="xs:long">
              <xs:minInclusive value="1"/>
            </xs:restriction>
          </xs:simpleType>
        </xs:element>
      </xs:sequence>
      <xs:attribute name="action_id" type="xs:decimal"
use="required" fixed="3.1"/>
      <xs:attribute name="version" type="xs:int" use="required"
fixed="1"/>
    </xs:complexType>
  </xs:element>
</xs:schema>

```

Приложение. Документ с информацией о СУЗ

Формат JSON

Пример:

```
{
  "certificate_serial_number": "< >",
  "certificate_owner_full_name": "< >",
  "certificate_authority_name": "< >",
  "certificate_authority_inn": "< >",
}
```

Атрибутивный состав:

certificate_serial_number - Серийный номер сертификата. Тип строковый.

certificate_owner_full_name - ФИО владельца сертификата. Тип строковый.

certificate_authority_name - Имя удостоверяющего центра выдавшего сертификат. Тип строковый.

certificate_authority_inn - ИНН удостоверяющего центра выдавшего сертификат. Тип строковый.

Приложение. Документ о вводе в оборот

Формат JSON

Пример:

```
{
  "document_description" : {
    "participant_inn": "6820022500",
    "marking_type": "PRODUCT_PACKAGE",
    "production_date": "2012-11-22T20:13:10.002Z",
    "producer_inn": "6820022500",
    "owner_inn": "6820022500",
    "production_type": "OWN_PRODUCTION"
  },
  "doc_type": "Promotion_Inform_Selfmade",
  "products": [
    {
      "production_date": "2012-11-22T20:13:10.002Z",
      "uit_code": "33333",
      "uitu_code": "123456789123456789",
      "tnved_code": "6401100000",
      "marking_type": "PRODUCT_PACKAGE",
      "producer_inn": "467780979386",
      "owner_inn": "467780979386",
      "certificate_document": "CONFORMITY_DECLAR",
      "certificate_document_number": "FFFFFF",
      "certificate_document_date": "2014-11-22T20:13:10.002Z"
    }
  ]
}
```

Атрибутивный состав:

document_description - общие параметры документа. Тип объект.

participant_inn - ИНН участника оборота товаров. Тип строковый.

producer_inn - ИНН производителя. Тип строка.

production_type - способ производства. Возможные значения OWN_PRODUCTION - собственное производство, CONTRACT_PRODUCTION - производство товара по договору. Тип строковый.

production_date - дата производства. Тип строка.

marking_type - вид маркировки. Возможные значения PRODUCT_PACKAGE - на потребительскую упаковку, PRODUCT - на товар, PRODUCT_LABEL - на товарный ярлык. Тип строковый.

products - массив КМ для ввода в оборот. Тип массив.

uit_code - вводимый в оборот КМ. Тип строковый.

uitu_code - уникальный идентификатор транспортной упаковки

tnved_code - 10-ти значный код ТН ВЭД

certificate_document - вид документа обязательной сертификации. Возможный значения CONFORMITY_CERT - сертификат соответствия, CONFORMITY_DECLAR - декларация соответствия. Тип строковый.

certificate_document_number - номер документа обязательной сертификации. Тип строковый.

certificate_document_date - дата документа обязательной сертификации. Тип строковый.

Формат XML

xsd схема:

```
<?xml version="1.0" encoding="UTF-8"?>
<xs:schema elementFormDefault="qualified"
xmlns:xs="http://www.w3.org/2001/XMLSchema">
  <xs:element name="Vvod" type="Vvod"/>
  <xs:complexType name="Vvod">
    <xs:annotation>
      <xs:documentation>      ( )</xs:documentation>
    </xs:annotation>
    <xs:sequence>
      <xs:element name="DocType">
        <xs:simpleType>
          <xs:restriction base="xs:string">
            <xs:enumeration value="Promotion_Inform_Selfmade"/>
          </xs:restriction>
        </xs:simpleType>
      </xs:element>
      <xs:element name="Participant_INN">
        <xs:simpleType>
          <xs:restriction base="xs:string">
            <xs:minLength value="10"/>
            <xs:maxLength value="12"/>
            <xs:pattern
value="((([0-9]{1}[1-9]{1}|[1-9]{1}[0-9]{1})[0-9]{8})|((([0-9]{1}[1-9]{1}|
[1-9]{1}[0-9]{1})[0-9]{10}))/>
          </xs:restriction>
        </xs:simpleType>
      </xs:element>
    </xs:sequence>
  </xs:complexType>
</xs:schema>
```



```

<xs:element name="Product_Date" minOccurs="0">
  <xs:simpleType>
    <xs:restriction base="xs:date"/>
  </xs:simpleType>
</xs:element>
<xs:element name="Producer_INN">
  <xs:simpleType>
    <xs:restriction base="xs:string">
      <xs:maxLength value="12"/>
      <xs:minLength value="10"/>
      <xs:pattern
value="(((0-9){1}[1-9]{1}|[1-9]{1}[0-9]{1})[0-9]{8})|(((0-9){1}[1-9]{1}|
[1-9]{1}[0-9]{1})[0-9]{10})"/>
    </xs:restriction>
  </xs:simpleType>
</xs:element>
<xs:element name="INNOwner">
  <xs:simpleType>
    <xs:restriction base="xs:string">
      <xs:minLength value="10"/>
      <xs:maxLength value="12"/>
      <xs:pattern
value="(((0-9){1}[1-9]{1}|[1-9]{1}[0-9]{1})[0-9]{8})|(((0-9){1}[1-9]{1}|
[1-9]{1}[0-9]{1})[0-9]{10})"/>
    </xs:restriction>
  </xs:simpleType>
</xs:element>
<xs:element name="Production_Order">
  <xs:annotation>
    <xs:documentation>Production_Order
1 -
2 - </xs:documentation>
  </xs:annotation>
  <xs:simpleType>
    <xs:restriction base="xs:int">
      <xs:enumeration value="1"/>
      <xs:enumeration value="2"/>
    </xs:restriction>
  </xs:simpleType>
</xs:element>
<xs:element name="MarkingType">
  <xs:annotation>
    <xs:documentation>
1 -
2 -
3 - </xs:documentation>
  </xs:annotation>
  <xs:simpleType>
    <xs:restriction base="xs:int">
      <xs:enumeration value="1"/>
    </xs:restriction>
  </xs:simpleType>
</xs:element>

```

```

                <xs:enumeration value="2"/>
                <xs:enumeration value="3"/>
            </xs:restriction>
        </xs:simpleType>
    </xs:element>
    <xs:element name="Products_List">
        <xs:complexType>
            <xs:sequence>
                <xs:element name="PRODUCT" maxOccurs="unbounded">
                    <xs:complexType>
                        <xs:sequence>
                            <xs:element name="Product_Date">
                                <xs:simpleType>
                                    <xs:restriction
base="xs:date">
                                        </xs:restriction>
                                    </xs:simpleType>
                                </xs:element>
                                <xs:element name="UIT" minOccurs="0">
                                    <xs:simpleType>
                                        <xs:restriction base="xs:string">
                                            <xs:minLength value="31"/>
                                            <xs:maxLength value="38"/>
                                            <xs:pattern value="[A-Za-z0-9]{31}|[A-Za-z0-9]{38}"/>
                                        </xs:restriction>
                                    </xs:simpleType>
                                </xs:element>
                                <xs:element name="UITU" minOccurs="0">
                                    <xs:simpleType>
                                        <xs:restriction
base="xs:string">
                                            <xs:length value="18"/>
                                        </xs:restriction>
                                    </xs:simpleType>
                                </xs:element>
                                <xs:element name="CodeTNVED">
                                    <xs:simpleType>
                                        <xs:restriction
base="xs:string">
                                            <xs:length value="10"/>
                                        </xs:restriction>
                                    </xs:simpleType>
                                </xs:element>
                                <xs:element name="MarkingType">
                                    <xs:annotation>
                                        <xs:documentation>
                                            1 -
                                            2 -
                                            3 - </xs:documentation>
                                        </xs:annotation>
                                </xs:element>
                            </xs:sequence>
                        </xs:complexType>
                    </xs:element>
                </xs:sequence>
            </xs:complexType>
        </xs:element>
    </xs:sequence>
</xs:complexType>
</xs:element name="Products_List">
</xs:element>

```

```

        <xs:simpleType>
            <xs:restriction
base="xs:int">
                <xs:enumeration
value="1"/>
                <xs:enumeration
value="2"/>
                <xs:enumeration
value="3"/>
            </xs:restriction>
        </xs:simpleType>
    </xs:element>
    <xs:element name="CertificateDoc">
        <xs:annotation>
            <xs:documentation>
                1 -
                2 -
                3 - </xs:documentation>
            </xs:annotation>
        <xs:simpleType>
            <xs:restriction
base="xs:int">
                <xs:enumeration
value="1"/>
                <xs:enumeration
value="2"/>
                <xs:enumeration
value="3"/>
            </xs:restriction>
        </xs:simpleType>
    </xs:element>
    <xs:element name="CertificateNumber">
        <xs:simpleType>
            <xs:restriction
base="xs:string"/>
        </xs:simpleType>
    </xs:element>
    <xs:element name="CertificateDate">
        <xs:simpleType>
            <xs:restriction
base="xs:date"/>
        </xs:simpleType>
    </xs:element>
</xs:sequence>
</xs:complexType>
</xs:element>
</xs:sequence>
</xs:complexType>
</xs:element>
</xs:sequence>

```

```
<xs:attribute name="action_id" type="xs:int" use="required"
fixed="05"/>
</xs:complexType>
```

```
</xs:schema>
```

Пример:

```
<?xml version="1.0" encoding="utf-8"?>
<Vvod xsi:noNamespaceSchemaLocation="schema.xsd" action_id="05"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance">
  <DocType>Promotion_Inform_Selfmade</DocType>
  <Participant_INN>0156328626</Participant_INN>
  <Product_Date>2007-10-26</Product_Date>
  <Producer_INN>8837133178</Producer_INN>
  <INNOwner>8407866816</INNOwner>
  <Production_Order>2</Production_Order>
  <MarkingType>2</MarkingType>
  <Products_List>
    <!--1 or more repetitions!-->
    <PRODUCT>
      <Product_Date>2003-01-01</Product_Date>
      <UIT>wpTBMVjbBsv5jGXRgf0yWh6Wf63aFYFT7rRo8y</UIT>
      <UITU>ABCDEFGHIJKLMNQPQR</UITU>
      <CodeTNVED>ABCDEFGHIJ</CodeTNVED>
      <MarkingType>2</MarkingType>
      <CertificateDoc>2</CertificateDoc>
      <CertificateNumber>string</CertificateNumber>
      <CertificateDate>2017-05-15</CertificateDate>
    </PRODUCT>
  </Products_List>
</Vvod>
```

Формат CSV

Пример:

```
' ' ' ' ' ' ' '
( ),2721169293,12.12.2012,2721169293,2721169293, , ,
' ' ' ' ' ' ' '
' ' ' ' ' ' ' '
0469025711403043EjYJqFDEoA56409,046902571140305OEKfTO2HoWFL6409,98767894
3,21.12.2015, , ,564534,12.12.2013
```

Приложение. Документ о вводе в оборот (Импорт)

Формат JSON

Пример:

```
{
  "doc_type": "GOODSIMPORT",
  "document_description": {
    "participant_inn": "467780979386",
    "declaration_date": "2019-12-31T21:00:00.000Z",
    "declaration_number": "decl num",
    "customs_code": "customs code",
    "decision_code": 1
  },
  "products": [
    {
      "uit_code": "33333",
      "uitu_code": "123456789123456789",
      "tnved_code": "0123456789",
      "certificate_document": "CONFORMITY_DECLAR",
      "certificate_document_number": "FFFFFF",
      "certificate_document_date": "2014-11-22T21:00:00.000Z",
    }
  ]
}
```

Атрибутивный состав:

document_description - общие параметры документа. Тип объект.

declaration_date - Дата декларации товара. Тип строковый.

declaration_number - Номер декларации товара. Тип строковый.

customs_code - Код таможенного органа. Тип строковый.

decision_code - Код приятного решения. Тип числовой.

participant_inn - ИНН участника оборота товаров. Тип строковый.

products - массив КМ для ввода в оборот. Тип массив.

uit_code - вводимый в оборот КМ. Тип строковый.

uitu_code - уникальный идентификатор транспортной упаковки

tnved_code - 10-ти значный код ТН ВЭД

certificate_document - вид документа обязательной сертификации. Возможный значения CONFORMITY_CERT - сертификат соответствия, CONFORMITY_DECLAR - декларация соответствия. Тип строковый.

certificate_document_number - номер документа обязательной сертификации. Тип строковый.

certificate_document_date - дата документа обязательной сертификации. Тип строковый.

Формат XML

xsd схема:

```
<?xml version="1.0" encoding="UTF-8"?>
```

```

<xs:schema elementFormDefault="qualified"
xmlns:xs="http://www.w3.org/2001/XMLSchema">
  <xs:include schemaLocation="../LP_base_types.xsd"/>
  <xs:element name="Vvod_Imp" type="Vvod_Imp"/>
  <xs:complexType name="Vvod_Imp">
    <xs:annotation>
      <xs:documentation>      (</xs:documentation>
    </xs:annotation>
    <xs:sequence>
      <xs:element name="DocType">
        <xs:simpleType>
          <xs:restriction base="xs:string">
            <xs:enumeration value="GOODSIMPORT"/>
          </xs:restriction>
        </xs:simpleType>
      </xs:element>
      <xs:element name="TRADE_PARTICIPANT_INN"
type="TRADE_PARTICIPANT_INN_type">
        <xs:annotation>
          <xs:documentation> </xs:documentation>
        </xs:annotation>
      </xs:element>
      <xs:element name="Declaration_Date" type="date_type">
        <xs:annotation>
          <xs:documentation> </xs:documentation>
        </xs:annotation>
      </xs:element>
      <xs:element name="Declaration_Number">
        <xs:annotation>
          <xs:documentation> </xs:documentation>
        </xs:annotation>
        <xs:simpleType>
          <xs:restriction base="xs:string">
            <xs:minLength value="1"/>
            <xs:maxLength value="255"/>
          </xs:restriction>
        </xs:simpleType>
      </xs:element>
      <xs:element name="Customs_Code">
        <xs:annotation>
          <xs:documentation> </xs:documentation>
        </xs:annotation>
        <xs:simpleType>
          <xs:restriction base="xs:string">
            <xs:minLength value="1"/>
            <xs:maxLength value="255"/>
          </xs:restriction>
        </xs:simpleType>
      </xs:element>
      <xs:element name="Decision_Code">

```

```

    <xs:annotation>
      <xs:documentation>
        10 -
        11 -           ,
          ' ' ' ' ' '
            12 13
        12 -           , 121
        13 -           , 122
        14 -           , 123

        20 -
      </xs:documentation>
    </xs:annotation>
    <xs:simpleType>
      <xs:restriction base="xs:int">
        <xs:enumeration value="10"/>
        <xs:enumeration value="11"/>
        <xs:enumeration value="12"/>
        <xs:enumeration value="13"/>
        <xs:enumeration value="14"/>
        <xs:enumeration value="20"/>
      </xs:restriction>
    </xs:simpleType>
  </xs:element>
  <xs:element name="Products_List">
    <xs:complexType>
      <xs:sequence>
        <xs:element name="PRODUCT"
maxOccurs="unbounded">
          <xs:complexType>
            <xs:sequence>
              <xs:element name="UIT"
type="gs1_uit_type" minOccurs="0">
                <xs:annotation>

<xs:documentation></xs:documentation>
              </xs:annotation>
            </xs:element>
            <xs:element name="UITU"
type="gs1_uitu_type" minOccurs="0">
              <xs:annotation>

<xs:documentation></xs:documentation>
            </xs:annotation>
          </xs:element>
          <xs:element name="Certificate_Doc">
            <xs:annotation>
              <xs:documentation>

```



```

1 -
2 -
    </xs:documentation>
</xs:annotation>
<xs:simpleType>
    <xs:restriction
base="xs:int">
        <xs:enumeration
value="1"/>
        <xs:enumeration
value="2"/>
    </xs:restriction>
</xs:simpleType>
</xs:element>
<xs:element
name="Certificate_Doc_Number">
    <xs:annotation>
        <xs:documentation>
        </xs:documentation>
    </xs:annotation>
    <xs:simpleType>
        <xs:restriction
base="xs:string">
            <xs:minLength
value="1"/>
            <xs:maxLength
value="255"/>
        </xs:restriction>
    </xs:simpleType>
</xs:element>
<xs:element
name="Certificate_Doc_Date" type="date_type">
    <xs:annotation>
        <xs:documentation>
        </xs:documentation>
    </xs:annotation>
</xs:element>
<xs:element name="TNVED_Code">
    <xs:simpleType>
        <xs:restriction
base="xs:string">
            <xs:length value="10"/>
        </xs:restriction>
    </xs:simpleType>
</xs:element>
</xs:sequence>
</xs:complexType>
</xs:element>
</xs:sequence>
</xs:complexType>

```

```

        </xs:element>
    </xs:sequence>
    <xs:attribute name="action_id" type="xs:string" use="required"
fixed="5.1"/>
    <xs:attribute name="version" type="xs:int" use="required"
fixed="1"/>
</xs:complexType>
</xs:schema>

```

Пример:

```

<?xml version="1.0" encoding="utf-8"?>
<Vvod_Imp action_id="5.1" version="1">
    <DocType>GOODSIMPORT</DocType>
    <TRADE_PARTICIPANT_INN>1234567890</TRADE_PARTICIPANT_INN>
    <Declaration_Date>01.01.2019</Declaration_Date>
    <Declaration_Number>string</Declaration_Number>
    <Customs_Code>string</Customs_Code>
    <Decision_Code>11</Decision_Code>
    <!--<Country_Code>string</Country_Code>-->
    <!--Optional:-->
    <Currency>string</Currency>

    <Products_List>
        <!--1 or more repetitions:-->
        <PRODUCT>
            <!--Optional:-->
            <UIT>stringstringstringstringstringstringst</UIT>
            <!--Optional:-->
            <UITU>stringstringstring</UITU>
            <Certificate_Doc>1</Certificate_Doc>
            <Certificate_Doc_Number>string</Certificate_Doc_Number>
            <Certificate_Doc_Date>01.01.2019</Certificate_Doc_Date>
            <TNVED_Code>1234567890</TNVED_Code>
        </PRODUCT>
    </Products_List>
</Vvod_Imp>

```

Формат CSV

Пример:

```
' ' ' ' ' ' ' '
( ),7728551528, ,12.12.2017,1234,111,10,1.0,,
,,,,,,,,,
' ' ' ' ' ' ' '
0104630033880100211AREwAwLETM7g240ffd0,123456789123456789,6401921000,
,12,23.09.2018
```

Приложение. Отгрузка

Формат JSON

Пример:

```
{
  "document_num": "784378438",
  "document_date": "2018-10-13T00:00:00.000Z",
  "receiver_inn": "7731369928",
  "receiver": "",
  "sender_inn": "007731369928",
  "sender": "\\\"\\\"",
  "owner_inn": "7731369928",
  "owner": "",
  "request_type": "SHIPMENT",
  "turnover_type": "SALE",
  "transfer_date": "2018-10-13T00:00:00.000Z",
  "pdf": "string",
  "products": [
    {
      "uitu_code": "",
      "uit_code": "010463003759131621qSyEMozA3oJjP2406401",
      "product_description": ".,.3F671-0606-A382B",
      "product_cost": 1000,
      "product_tax": 10
    }
  ]
}
```



После успешной валидации документа по отгрузке его статус меняется на WAIT_ACCEPTANCE [Ожидает приёмку], после Приёмки, статус документа переходит в CHECKED_OK.

Атрибутивный состав:

request_type - тип документа. Должен быть "SHIPMENT". Тип строковый.

document_num - номер первичного документа. Тип строковый.

document_date - дата первичного документа. Тип строковый.

transfer_date - дата передачи маркированных товаров. Тип строковый.

receiver_inn - ИНН получателя. Тип строковый.

receiver - наименование получателя. Тип строковый.

owner_inn - ИНН получателя. Тип строковый.

owner - наименование получателя. Тип строковый.

turnover_type - тип отгрузки. Возможные значения SALE - продажа-покупка, COMMISSION - комиссия, AGENT - агент, CONTRACT - передача маркированных товаров по договорам подряда. Тип строковый.

products - массив КМ для ввода в оборот. Тип массив.

uit_code - КМ товара

uitu_code - КМ упаковки

product_description - наименования продукта

product_cost - цена

product_tax - НДС

Формат XML

xsd схема:

```
<?xml version="1.0" encoding="UTF-8"?>
<!-- edited with XMLSpy v2009 (http://www.altova.com) by Admin (EMBRACE) -->
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema"
elementFormDefault="qualified" attributeFormDefault="unqualified">
  <xs:element name="shipment_signs" type="shipment_signs"/>
  <xs:simpleType name="TRADE_PARTICIPANT_INN_type">
    <xs:annotation>
      <xs:documentation> </xs:documentation>
    </xs:annotation>
    <xs:restriction base="xs:string">
      <xs:minLength value="10"/>
      <xs:maxLength value="12"/>
      <xs:pattern
value="((([0-9]{1}[1-9]{1}|[1-9]{1}[0-9]{1})[0-9]{8})|((([0-9]{1}[1-9]{1}|
[1-9]{1}[0-9]{1})[0-9]{10}))/>
    </xs:restriction>
  </xs:simpleType>
  <xs:simpleType name="datetimeoffset">
    <xs:restriction base="xs:dateTime">
      <xs:minInclusive value="0001-01-01T00:00:00Z"/>
      <xs:pattern
value="((000[1-9])|(00[1-9][0-9])|(0[1-9][0-9]{2})|([1-9][0-9]{3}))-(0[
1-9])|(1[012]))-(0[1-9])|([12][0-9])|(3[01]))T((([01][0-9])|(2[0-3]))(:[
0-5][0-9]){2}(\.[0-9]+)?(([-+])(((0[0-9])|(1[0-3]))(:[0-5][0-9]))|14:00
)|Z)"/>
    </xs:restriction>
  </xs:simpleType>
```

```

    <xs:simpleType name="document_number_200_type">
<xs:annotation>
    <xs:documentation> ,    200 </xs:documentation>
</xs:annotation>
<xs:restriction base="xs:string">
    <xs:minLength value="1"/>
    <xs:maxLength value="200"/>
</xs:restriction>
</xs:simpleType>
    <xs:simpleType name="date_type">
<xs:annotation>
    <xs:documentation>    ..</xs:documentation>
</xs:annotation>
<xs:restriction base="xs:string">
    <xs:length value="10"/>
    <xs:pattern
value="(0[1-9]{1}|[1-2]{1}[0-9]{1}|3[0-1]{1})\.(0[1-9]{1}|1[0-2]{1})\.(1
[0-9]{3}|20[0-9]{2})"/>
</xs:restriction>
</xs:simpleType>
    <xs:simpleType name="turnover_type_enum">
<xs:annotation>
    <xs:documentation>
1 - -
2 -
3 -
4 -
    </xs:documentation>
</xs:annotation>
<xs:restriction base="xs:int">
    <xs:enumeration value="1"/>
    <xs:enumeration value="2"/>
    <xs:enumeration value="3"/>
    <xs:enumeration value="4"/>
</xs:restriction>
</xs:simpleType>
    <xs:simpleType name="gs1_uit_type">
<xs:annotation>
    <xs:documentation>    </xs:documentation>
</xs:annotation>
<xs:restriction base="xs:string">
    <xs:length value="31"/>
    <xs:pattern value="[A-Za-z0-9]{31}" />
</xs:restriction>
</xs:simpleType>
    <xs:simpleType name="price_type">
<xs:annotation>
<xs:documentation></xs:documentation>
</xs:annotation>
<xs:restriction base="xs:decimal">

```

```
<xs:totalDigits value="16"/>
<xs:fractionDigits value="2"/>
<xs:minInclusive value="0"/>
  </xs:restriction>
</xs:simpleType>
<xs:complexType name="shipment_signs">
  <xs:annotation>
    <xs:documentation> </xs:documentation>
  </xs:annotation>
  <xs:sequence>
    <xs:element name="subject_inn" type="TRADE_PARTICIPANT_INN_type">
      <xs:annotation>
        <xs:documentation> </xs:documentation>
      </xs:annotation>
    </xs:element>
    <xs:element name="recipient_inn" type="TRADE_PARTICIPANT_INN_type">
      <xs:annotation>
        <xs:documentation> </xs:documentation>
      </xs:annotation>
    </xs:element>
    <xs:element name="owner_inn" type="TRADE_PARTICIPANT_INN_type">
      <xs:annotation>
        <xs:documentation> </xs:documentation>
      </xs:annotation>
    </xs:element>
    <xs:element name="transfer_date" type="date_type">
      <xs:annotation>
        <xs:documentation> </xs:documentation>
      </xs:annotation>
    </xs:element>
    <xs:element name="move_doc_num" type="document_number_200_type">
      <xs:annotation>
        <xs:documentation> : </xs:documentation>
      </xs:annotation>
    </xs:element>
    <xs:element name="move_doc_date" type="date_type">
      <xs:annotation>
        <xs:documentation> : </xs:documentation>
      </xs:annotation>
    </xs:element>
    <xs:element name="turnover_type" type="turnover_type_enum">
      <xs:annotation>
        <xs:documentation> </xs:documentation>
      </xs:annotation>
    </xs:element>
    <xs:element name="order_details">
      <xs:annotation>
        <xs:documentation> </xs:documentation>
      </xs:annotation>
    </xs:element>
  </xs:sequence>
</xs:complexType>
```

```

<xs:sequence>
  <xs:element name="signs">
    <xs:complexType>
      <xs:sequence>
        <xs:choice>
          <xs:element name="sign_uit" maxOccurs="unbounded">
            <xs:complexType>
              <xs:sequence>
                <xs:element name="uit" type="gs1_uit_type">
                  <xs:annotation>
                    <xs:documentation> </xs:documentation>
                  </xs:annotation>
                </xs:element>
              </xs:sequence>
              <xs:attribute name="cost" type="price_type" use="optional">
                <xs:annotation>
                  <xs:documentation> </xs:documentation>
                </xs:annotation>
              </xs:attribute>
              <xs:attribute name="vat_value" type="price_type"
use="optional">
                <xs:annotation>
                  <xs:documentation> ( )</xs:documentation>
                </xs:annotation>
              </xs:attribute>
            </xs:complexType>
          </xs:element>
          <xs:element name="sign_uitu" maxOccurs="unbounded">
            <xs:complexType>
              <xs:sequence>
                <xs:element name="uitu" type="gs1_uitu_type">
                  <xs:annotation>
                    <xs:documentation> </xs:documentation>
                  </xs:annotation>
                </xs:element>
              </xs:sequence>
              <xs:attribute name="cost" type="price_type" use="optional">
                <xs:annotation>
                  <xs:documentation> </xs:documentation>
                </xs:annotation>
              </xs:attribute>
              <xs:attribute name="vat_value" type="price_type"
use="optional">
                <xs:annotation>
                  <xs:documentation> ()</xs:documentation>
                </xs:annotation>
              </xs:attribute>
            </xs:complexType>
          </xs:element>
        </xs:choice>
      </xs:sequence>
    </xs:element>
  </xs:sequence>

```

```
    </xs:sequence>
  </xs:complexType>
  <xs:unique name="sign_uit_unique">
    <xs:selector xpath="sign"/>
    <xs:field xpath="sign_uit"/>
  </xs:unique>
  <xs:unique name="sign_uitu_unique">
    <xs:selector xpath="sign"/>
    <xs:field xpath="sign_uit"/>
  </xs:unique>
</xs:element>
</xs:sequence>
</xs:complexType>
<xs:unique name="sign_uit_unique_in_details">
  <xs:selector xpath="signs/sign"/>
  <xs:field xpath="sign_uitu"/>
</xs:unique>
</xs:element>
</xs:sequence>
  <xs:attribute name="action_id" type="xs:int" use="required"
fixed="10"/>
  <xs:attribute name="version" type="xs:decimal" use="required"
```



```
fixed="1.02"/>
  </xs:complexType>
</xs:schema>
```

Пример:

```
<shipment_signs action_id="10" version="1.02">
  <subject_inn>7724211288</subject_inn>
  <recipient_inn>7731369928</recipient_inn>
  <owner_inn>7724211288</owner_inn>
  <transfer_date>30.01.2018</transfer_date>
  <move_doc_num>12345</move_doc_num>
  <move_doc_date>30.01.2018</move_doc_date>
  <turnover_type>1</turnover_type>
  <order_details>
    <signs><!-- //-->
      <sign_uit cost="1000.000000000000"
vat_value="10.000000000000">
        <uit>010463003759026521H7LNyi59XoL84</uit>
      </sign_uit>
      <sign_uit cost="2000.000000000000"
vat_value="20.000000000000">
        <uit>010463003759026521H7LNyi59XoL83</uit>
      </sign_uit>
    </signs>
    <signs><!-- //-->
      <sign_uitu cost="1000.000000000000"
vat_value="10.000000000000">
        <uitu>123456789123456789</uitu>
      </sign_uitu>
    </signs>
  </order_details>
</shipment_signs>
```

Формат CSV

Пример:

```
','','','','','','','','',''
,7731369928, ,7731369928, ,7724211288,123456,12.01.2019,24.09.2018
11:33,-,1.02
/,, ,,,,,,
010469022978096621tER1N5qDnnnck2406402,"33030,5","270,85",,,,,,
```

Приложение. Документ об отмене отгрузки товара

Формат JSON

Пример:

```
{
  "participant_inn": "7731369928",
  "shipment_number": "1dc9d25f-8e76-451f-9def-ea151c2285fc"
}
```

Атрибутивный состав:

participant_inn - ИНН участника оборота товаров. Тип строковый.

shipment_number - номер отменяемого документа об отгрузки. Тип строковый.

Приложение. Документ о приемке товара

Формат JSON

Пример:

```
{
  "request_type": "ACCEPTANCE",
  "release_order_number": "57250cce-52cf-408b-99ef-39cc239f60fd",
  "products": [
    {
      "uit_code": "010463003407001221gZ1RSRRsefaPt2406402",
      "product_description": " , , 17",
      "product_cost": 100,
      "product_tax": 10
    }
  ],
  "document_date": "2018-12-31T21:00:00.000Z",
  "transfer_date": "2019-01-01T21:00:00.000Z",
  "document_number": "434",
  "trade_sender_inn": "7724211288",
  "trade_owner_inn": "7724211288",
  "trade_sender_name": " \\\"\\\" ",
  "trade_owner_name": " ",
  "trade_recipient_inn": "7728551777",
  "turnover_type": "COMMISSION",
  "acceptance_date": "2019-01-01T21:00:00.000Z"
}
```

Атрибутивный состав:

request_type - тип документа. Должен быть "ACCEPTANCE".

release_order_number - номер заявки на отгрузку

products - массив с описанием принимаемого товара. Тип массив.

uit_code - код товара. Тип строковый.

product_cost - цена. Тип числовой.

product_tax - НДС. Тип числовой.

product_description - наименование товара. Тип строковый.

document_date - дата первичного документа. Формат Unix time в ms. Тип числовой.

transfer_date - дата отгрузки товара. Формат Unix time в ms. Тип числовой.

document_number - номер первичного документа. Тип строковый.

trade_sender_inn - ИНН отправителя. Тип строковый.

trade_owner_inn - ИНН получателя. Тип строковый.

trade_sender_name - наименование организации отправителя. Тип строковый.

trade_owner_name - наименование организации получателя. Тип строковый.

turnover_type - тип отгрузки. Возможные значения SALE - продажа-покупка, COMMISSION - комиссия, AGENT - агент, CONTRACT - передача маркированных товаров по договорам подряда. Тип строковый.

acceptance_date - дата приемки товара. Тип числовой.

Формат XML

xsd схема:

```
<?xml version="1.0" encoding="UTF-8"?>
<!-- edited with XMLSpy v2009 (http://www.altova.com) by Admin (EMBRACE)
-->
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema"
elementFormDefault="qualified" attributeFormDefault="unqualified">
  <xs:element name="reception_signs" type="reception_signs"/>
  <xs:simpleType name="TRADE_PARTICIPANT_INN_type">
    <xs:annotation>
      <xs:documentation> </xs:documentation>
    </xs:annotation>
    <xs:restriction base="xs:string">
      <xs:minLength value="10"/>
      <xs:maxLength value="12"/>
      <xs:pattern
value="((([0-9]{1}[1-9]{1}|[1-9]{1}[0-9]{1})[0-9]{8})|((([0-9]{1}[1-9]{1}|
[1-9]{1}[0-9]{1})[0-9]{10}))/>
    </xs:restriction>
  </xs:simpleType>
  <xs:simpleType name="datetimeoffset">
    <xs:restriction base="xs:dateTime">
      <xs:minInclusive value="0001-01-01T00:00:00Z"/>
      <xs:pattern
value="((000[1-9])|(00[1-9][0-9])|(0[1-9][0-9]{2})|([1-9][0-9]{3}))-(00[
```

```

1-9])|(1[012]))-((0[1-9])|([12][0-9])|(3[01]))T(([01][0-9])|(2[0-3]))(:[
0-5][0-9]){2}(\.[0-9]+)?(([-+])(((0[0-9])|(1[0-3]))(:[0-5][0-9]))|14:00
)|Z)"/>
    </xs:restriction>
  </xs:simpleType>
  <xs:simpleType name="document_number_200_type">
    <xs:annotation>
      <xs:documentation> ,      200 </xs:documentation>
    </xs:annotation>
    <xs:restriction base="xs:string">
      <xs:minLength value="1"/>
      <xs:maxLength value="200"/>
    </xs:restriction>
  </xs:simpleType>
  <xs:simpleType name="date_type">
    <xs:annotation>
      <xs:documentation> ..</xs:documentation>
    </xs:annotation>
    <xs:restriction base="xs:string">
      <xs:length value="10"/>
      <xs:pattern
value="(0[1-9]{1}|[1-2]{1}[0-9]{1}|3[0-1]{1})\.(0[1-9]{1}|1[0-2]{1})\.(1
[0-9]{3}|20[0-9]{2})"/>
    </xs:restriction>
  </xs:simpleType>
  <xs:simpleType name="guid_type">
    <xs:annotation>
      <xs:documentation>
        GUID
      </xs:documentation>
    </xs:annotation>
    <xs:restriction base="xs:string">
      <xs:length value="36"/>
      <xs:pattern
value="([a-fA-F0-9]{8}-[a-fA-F0-9]{4}-[a-fA-F0-9]{4}-[a-fA-F0-9]{4}-[a-f
A-F0-9]{12})"/>
    </xs:restriction>
  </xs:simpleType>
  <xs:simpleType name="turnover_type_enum">
    <xs:annotation>
      <xs:documentation>
        1 - -
        2 -
        3 -
        4 -
      </xs:documentation>
    </xs:annotation>
    <xs:restriction base="xs:int">
      <xs:enumeration value="1"/>
      <xs:enumeration value="2"/>

```

```

        <xs:enumeration value="3"/>
        <xs:enumeration value="4"/>
    </xs:restriction>
</xs:simpleType>
<xs:simpleType name="gsl_uit_type">
    <xs:annotation>
        <xs:documentation> </xs:documentation>
    </xs:annotation>
    <xs:restriction base="xs:string">
        <xs:length value="31"/>
        <xs:pattern value="[A-Za-z0-9]{31}"/>
    </xs:restriction>
</xs:simpleType>
<xs:simpleType name="price_type">
    <xs:annotation>
        <xs:documentation></xs:documentation>
    </xs:annotation>
    <xs:restriction base="xs:decimal">
        <xs:totalDigits value="16"/>
        <xs:fractionDigits value="2"/>
        <xs:minInclusive value="0"/>
    </xs:restriction>
</xs:simpleType>
<xs:complexType name="reception_signs">
    <xs:annotation>
        <xs:documentation> </xs:documentation>
    </xs:annotation>
    <xs:sequence>
        <xs:element name="session_ui" type="guid_type">
            <xs:annotation>
                <xs:documentation>ID </xs:documentation>
            </xs:annotation>
        </xs:element>
        <xs:element name="subject_inn" type="TRADE_PARTICIPANT_INN_type">
            <xs:annotation>
                <xs:documentation> </xs:documentation>
            </xs:annotation>
        </xs:element>
        <xs:element name="recipient_inn" type="TRADE_PARTICIPANT_INN_type">
            <xs:annotation>
                <xs:documentation> </xs:documentation>
            </xs:annotation>
        </xs:element>
        <xs:element name="owner_inn" type="TRADE_PARTICIPANT_INN_type">
            <xs:annotation>
                <xs:documentation> </xs:documentation>
            </xs:annotation>
        </xs:element>
        <xs:element name="sender_date" type="datetimeoffset">
            <xs:annotation>

```

```

    <xs:documentation>    </xs:documentation>
  </xs:annotation>
</xs:element>
<xs:element name="recipient_date" type="datetimeoffset">
  <xs:annotation>
    <xs:documentation>    </xs:documentation>
  </xs:annotation>
</xs:element>
<xs:element name="transfer_date" type="date_type">
  <xs:annotation>
    <xs:documentation>    </xs:documentation>
  </xs:annotation>
</xs:element>
<xs:element name="move_doc_num" type="document_number_200_type">
  <xs:annotation>
    <xs:documentation>    :    </xs:documentation>
  </xs:annotation>
</xs:element>
<xs:element name="move_doc_date" type="date_type">
  <xs:annotation>
    <xs:documentation>    :    </xs:documentation>
  </xs:annotation>
</xs:element>
<xs:element name="turnover_type" type="turnover_type_enum">
  <xs:annotation>
    <xs:documentation>    </xs:documentation>
  </xs:annotation>
</xs:element>
<xs:element name="order_details">
  <xs:annotation>
    <xs:documentation>    </xs:documentation>
  </xs:annotation>
  <xs:complexType>
    <xs:sequence>
      <xs:element name="signs">
        <xs:complexType>
          <xs:sequence>
            <xs:choice>
              <xs:element name="sign_uit" maxOccurs="unbounded">
                <xs:complexType>
                  <xs:sequence>
                    <xs:element name="uit" type="gs1_uit_type">
                      <xs:annotation>
                        <xs:documentation>    </xs:documentation>
                      </xs:annotation>
                    </xs:element>
                  </xs:sequence>
                </xs:complexType>
              <xs:attribute name="accept_type" type="xs:boolean"
use="required">
                <xs:annotation>

```

```

        <xs:documentation> </xs:documentation>
    </xs:annotation>
</xs:attribute>
<xs:attribute name="cost" type="price_type" use="optional">
    <xs:annotation>
        <xs:documentation> </xs:documentation>
    </xs:annotation>
</xs:attribute>
<xs:attribute name="vat_value" type="price_type"
use="optional">
    <xs:annotation>
        <xs:documentation> ( )</xs:documentation>
    </xs:annotation>
</xs:attribute>
</xs:complexType>
</xs:element>
<xs:element name="sign_uitu" maxOccurs="unbounded">
    <xs:complexType>
        <xs:sequence>
            <xs:element name="uitu" type="gsl_uitu_type">
                <xs:annotation>
                    <xs:documentation> </xs:documentation>
                </xs:annotation>
            </xs:element>
        </xs:sequence>
        <xs:attribute name="accept_type" type="xs:boolean"
use="required">
            <xs:annotation>
                <xs:documentation> </xs:documentation>
            </xs:annotation>
        </xs:attribute>
        <xs:attribute name="cost" type="price_type" use="optional">
            <xs:annotation>
                <xs:documentation> </xs:documentation>
            </xs:annotation>
        </xs:attribute>
        <xs:attribute name="vat_value" type="price_type"
use="optional">
            <xs:annotation>
                <xs:documentation> ( )</xs:documentation>
            </xs:annotation>
        </xs:attribute>
    </xs:complexType>
</xs:element>
</xs:choice>
</xs:sequence>
</xs:complexType>
<xs:unique name="sign_uit_unique">
    <xs:selector xpath="sign"/>
    <xs:field xpath="sign_uit"/>

```

```
</xs:unique>
<xs:unique name="sign_uitu_unique">
  <xs:selector xpath="sign"/>
  <xs:field xpath="sign_uit"/>
</xs:unique>
</xs:element>
</xs:sequence>
</xs:complexType>
<xs:unique name="sign_uit_unique_in_details">
  <xs:selector xpath="signs/sign"/>
  <xs:field xpath="sign_uitu"/>
</xs:unique>
</xs:element>
</xs:sequence>
<xs:attribute name="action_id" type="xs:int" use="required"
fixed="11"/>
<xs:attribute name="version" type="xs:decimal" use="required"
```



```
fixed="1.02"/>
  </xs:complexType>
</xs:schema>
```

Пример:

```
<reception_signs action_id="11" version="1.02">
  <session_ui>57250cce-52cf-408b-99ef-39cc239f60fd</session_ui>
  <subject_inn>7724211288</subject_inn>
  <recipient_inn>7728551777</recipient_inn>
  <owner_inn>7724211288</owner_inn>
  <sender_date>2014-09-19T03:18:33.000Z</sender_date>
  <recipient_date>2014-09-19T03:18:33.000Z</recipient_date>
  <transfer_date>01.01.2019</transfer_date>
  <move_doc_num>434</move_doc_num>
  <move_doc_date>01.01.2016</move_doc_date>
  <turnover_type>2</turnover_type>
  <order_details>
    <signs>
      <sign_uit cost="1000.000000000000"
vat_value="10.000000000000" accept_type="true">
        <uit>010463003759026521H7LNyi59XoL84</uit>
      </sign_uit>
      <sign_uit cost="1000.000000000000"
vat_value="10.000000000000" accept_type="true">
        <uit>010463003759026521H7LNyi59XoL84</uit>
      </sign_uit>
    </signs>
    <signs>
      <sign_uitu cost="1000.000000000000"
vat_value="10.000000000000" accept_type="true">
        <uitu>123456789123456789</uitu>
      </sign_uitu>
      <sign_uitu cost="1000.000000000000"
vat_value="10.000000000000" accept_type="true">
        <uitu>123456789123456789</uitu>
      </sign_uitu>
    </signs>
  </order_details>
</reception_signs>
```

Приложение. Документ по регистрации УОТ

Формат JSON:

```

{
  "organisation_type": "TRADE_PARTICIPANT",
  "inn": "7724211299",
  "organisation_name": "",
  "user_name": "",
  "email": "gladilin@mail.com",
  "cert_number": "FF",
  "trust_center_inn": "7967176620"
}

```

Атрибутивный состав (Все поля обязательные):

organisation_type - тип организации "TRADE_PARTICIPANT" <Участник оборота товаров>, "PRODUCER" <Производитель товара>, "IMPORTER" <Импортер товара>, "WHOLESALE" <Оптовая торговля>, "RETAIL" <Розничная торговля>, "IS_MP_OPERATOR" ("Оператор ИС МП"), "CEM" <LABELLING_CENTER>, "OGV" <Орган государственной власти>; Тип строковый

inn - ИНН организации. Тип строковый.

organisation_name - Имя организации. Тип строковый.

user_name - ФИО пользователя. Тип строковый.

email - адрес электронной почты. Тип строковый.

cert_number - номер сертификата. Тип строковый.

trust_center_inn - ИНН удостоверяющего центра.

Формат XML

xsd схема:

```

<?xml version="1.0" encoding="UTF-8"?>
<xs:schema elementFormDefault="qualified"
xmlns:xs="http://www.w3.org/2001/XMLSchema">
  <xs:element name="RegReq" type="RegReq"/>
  <xs:simpleType name="TRADE_PARTICIPANT_INN_type">
    <xs:annotation>
      <xs:documentation> </xs:documentation>
    </xs:annotation>
    <xs:restriction base="xs:string">
      <xs:minLength value="10"/>
      <xs:maxLength value="12"/>
      <xs:pattern
value="((([0-9]{1}[1-9]{1}|[1-9]{1}[0-9]{1})[0-9]{8})|((([0-9]{1}[1-9]{1}|
[1-9]{1}[0-9]{1})[0-9]{10}))/>
      </xs:restriction>
    </xs:simpleType>
    <xs:simpleType name="datetimeoffset">
      <xs:restriction base="xs:dateTime">
        <xs:minInclusive value="0001-01-01T00:00:00Z"/>
        <xs:pattern
value="((000[1-9])|(00[1-9][0-9])|(0[1-9][0-9]{2})|([1-9][0-9]{3}))-(0[
1-9])|(1[012]))-(0[1-9])|([12][0-9])|(3[01]))T((01[0-9])|(2[0-3]))(:[

```

```

0-5][0-9]){2}(\.[0-9]+)?((([+-](((0[0-9])|(1[0-3]))(:[0-5][0-9]))|14:00)
)|Z)"/>
    </xs:restriction>
</xs:simpleType>
<xs:simpleType name="document_number_200_type">
    <xs:annotation>
        <xs:documentation> ,      200 </xs:documentation>
    </xs:annotation>
    <xs:restriction base="xs:string">
        <xs:minLength value="1"/>
        <xs:maxLength value="200"/>
    </xs:restriction>
</xs:simpleType>
<xs:simpleType name="date_type">
    <xs:annotation>
        <xs:documentation> ..</xs:documentation>
    </xs:annotation>
    <xs:restriction base="xs:string">
        <xs:length value="10"/>
        <xs:pattern
value="(0[1-9]{1}|[1-2]{1}[0-9]{1}|3[0-1]{1})\.(0[1-9]{1}|1[0-2]{1})\.(1
[0-9]{3}|20[0-9]{2})"/>
    </xs:restriction>
</xs:simpleType>
<xs:simpleType name="guid_type">
    <xs:annotation>
        <xs:documentation>
            GUID
        </xs:documentation>
    </xs:annotation>
    <xs:restriction base="xs:string">
        <xs:length value="36"/>
        <xs:pattern
value="([a-fA-F0-9]{8}-[a-fA-F0-9]{4}-[a-fA-F0-9]{4}-[a-fA-F0-9]{4}-[a-f
A-F0-9]{12})"/>
    </xs:restriction>
</xs:simpleType>
<xs:simpleType name="turnover_type_enum">
    <xs:annotation>
        <xs:documentation>
            1 - -
            2 -
            3 -
            4 -
        </xs:documentation>
    </xs:annotation>
    <xs:restriction base="xs:int">
        <xs:enumeration value="1"/>
        <xs:enumeration value="2"/>
        <xs:enumeration value="3"/>

```

```

        <xs:enumeration value="4"/>
    </xs:restriction>
</xs:simpleType>
<xs:simpleType name="gsl_uit_type">
    <xs:annotation>
        <xs:documentation> </xs:documentation>
    </xs:annotation>
    <xs:restriction base="xs:string">
        <xs:length value="31"/>
        <xs:pattern value="[A-Za-z0-9]{31}"/>
    </xs:restriction>
</xs:simpleType>
<xs:complexType name="RegReq">
    <xs:annotation>
        <xs:documentation> </xs:documentation>
    </xs:annotation>
    <xs:sequence>
        <xs:element name="PARTICIPANT_TYPE">
            <xs:annotation>
                <xs:documentation>
                    1 -
                    2 -
                    3 -
                    4 -
                    5 - </xs:documentation>
            </xs:annotation>
            <xs:simpleType>
                <xs:restriction base="xs:int">
                    <xs:enumeration value="1"/>
                    <xs:enumeration value="2"/>
                    <xs:enumeration value="3"/>
                    <xs:enumeration value="4"/>
                </xs:restriction>
            </xs:simpleType>
        </xs:element>
        <xs:element name="PARTICIPANT_INN" type="TRADE_PARTICIPANT_INN_type">
            <xs:annotation>
                <xs:documentation></xs:documentation>
            </xs:annotation>
        </xs:element>
        <xs:element name="PARTICIPANT_NAME">
            <xs:annotation>
                <xs:documentation> </xs:documentation>
            </xs:annotation>
            <xs:simpleType>
                <xs:restriction base="xs:string">
                    <xs:maxLength value="1000"/>
                </xs:restriction>
            </xs:simpleType>
        </xs:element>

```

```
<xs:element name="PARTICIPANT_AUTH">
  <xs:annotation>
    <xs:documentation> , </xs:documentation>
  </xs:annotation>
  <xs:simpleType>
    <xs:restriction base="xs:string">
      <xs:maxLength value="1000"/>
    </xs:restriction>
  </xs:simpleType>
</xs:element>
<xs:element name="PARTICIPANT_EMAIL">
  <xs:annotation>
    <xs:documentation> </xs:documentation>
  </xs:annotation>
  <xs:simpleType>
    <xs:restriction base="xs:string">
      <xs:maxLength value="1000"/>
    </xs:restriction>
  </xs:simpleType>
</xs:element>
<xs:element name="cert_number">
  <xs:annotation>
    <xs:documentation> </xs:documentation>
  </xs:annotation>
  <xs:simpleType>
    <xs:restriction base="xs:string">
      <xs:maxLength value="500"/>
    </xs:restriction>
  </xs:simpleType>
</xs:element>
<xs:element name="trust_center_inn"
type="TRADE_PARTICIPANT_INN_type">
  <xs:annotation>
    <xs:documentation> </xs:documentation>
  </xs:annotation>
</xs:element>
</xs:sequence>
<xs:attribute name="action_id" type="xs:int" use="required"
fixed="00"/>
<xs:attribute name="version" type="xs:int" use="required" fixed="3"/>
</xs:complexType>
```

```
</xs:schema>
```

Пример xml:

```
<?xml version="1.0" encoding="utf-8"?>
<RegReq action_id="00" version="3">
  <PARTICIPANT_TYPE>1</PARTICIPANT_TYPE>
  <PARTICIPANT_INN>7724211299</PARTICIPANT_INN>
  <PARTICIPANT_NAME></PARTICIPANT_NAME>
  <PARTICIPANT_AUTH> </PARTICIPANT_AUTH>
  <PARTICIPANT_EMAIL>gladilin@mail.com</PARTICIPANT_EMAIL>
  <cert_number>42</cert_number>
  <trust_center_inn>7967176620</trust_center_inn>
</RegReq>
```

Приложение. Документ по блокировке/возобновлению/удалению пользователей

Формат JSON:

```
{
  "participant_inn": "5031076070"
  "subject_persons": [
    { "user_id": 33, "user_name": " " },
    { "user_id": 33, "user_name": " " }
  ]
}
```

Атрибутивный состав (Все поля обязательные):

participant_inn - ИНН участника, подписавшего заявку. Тип строковый.

user_id - ID пользователя. Тип long.

user_name - Имя пользователя. Тип Строковый.

Формат XML

xsd схема:

```
<?xml version="1.0" encoding="UTF-8"?>
<xs:schema elementFormDefault="qualified"
  xmlns:xs="http://www.w3.org/2001/XMLSchema">

  <xs:element name="adding_user" type="adding_user"/>
  <xs:simpleType name="TRADE_PARTICIPANT_INN_type">
```

```

<xs:annotation>
  <xs:documentation> </xs:documentation>
</xs:annotation>
<xs:restriction base="xs:string">
  <xs:minLength value="10"/>
  <xs:maxLength value="12"/>
  <xs:pattern
value="((( [0-9]{1}[1-9]{1}|[1-9]{1}[0-9]{1}))[0-9]{8})|((( [0-9]{1}[1-9]{1}|
[1-9]{1}[0-9]{1}))[0-9]{10})"/>
  </xs:restriction>
</xs:simpleType>
<xs:complexType name="adding_user">
  <xs:annotation>
    <xs:documentation> , / </xs:documentation>
  </xs:annotation>
  <xs:sequence>
    <xs:element name="participant_inn"
type="TRADE_PARTICIPANT_INN_type">
      <xs:annotation>
        <xs:documentation> ///</xs:documentation>
      </xs:annotation>
    </xs:element>

    <xs:sequence minOccurs="1" maxOccurs="unbounded">

      <xs:annotation>
        <xs:documentation> </xs:documentation>
      </xs:annotation>
      <xs:element name="subject_person">
        <xs:annotation>
          <xs:documentation> </xs:documentation>
        </xs:annotation>

      <xs:complexType>
        <xs:sequence>
          <xs:element name="participant_id">
            <xs:annotation>
              <xs:documentation>ID </xs:documentation>
            </xs:annotation>
            <xs:simpleType>
              <xs:restriction base="xs:long">
                <xs:minInclusive value="1"/>
              </xs:restriction>
            </xs:simpleType>
          </xs:element>
          <xs:element name="participant_auth">
            <xs:annotation>
              <xs:documentation> , / </xs:documentation>
            </xs:annotation>
            <xs:simpleType>

```

```
        <xs:restriction base="xs:string">
          <xs:maxLength value="500"/>
        </xs:restriction>
      </xs:simpleType>
    </xs:element>
  </xs:sequence>
</xs:complexType>
</xs:element>

</xs:sequence>
</xs:sequence>
<xs:attribute name="action_id" type="xs:decimal" use="required"
fixed="02.3"/>
```



```
</xs:complexType>
</xs:schema>
```

Пример xml:

```
<?xml version="1.0" encoding="utf-8"?>
<adding_user action_id="02.3">
  <participant_inn>5031076070</participant_inn>
  <subject_person>
    <participant_id>33</participant_id>
    <participant_auth> </participant_auth>
  </subject_person>
  <subject_person>
    <participant_id>33</participant_id>
    <participant_auth> </participant_auth>
  </subject_person>
</adding_user>
```

Приложение. Документ вывода товара из оборота при реализации конечному потребителю

Формат JSON

Пример:

```
{
  "order_number": "1234",
  "order_date": "2018-09-17T21:00:00.000Z",
  "inn": "7703799090",
  "action": 1,
  "action_date": "2018-09-17T21:00:00.000Z",
  "document_type": 1,
  "document_number": "document_view_1",
  "document_date": "2018-09-17T21:00:00.000Z",
  "products": [
    {
      "cis": "12345612345612345612345612345612345678",
      "product_cost": 1000,
      "product_tax": "10.05"
    }
  ]
}
```

Атрибутивный состав:

order_number- Номер заявки на вывод из оборота товара. Тип строковый. Обязательное.

order_date- Дата заявки на вывод из оборота товара. Тип строковый.

inn- ИНН участника оборота. Тип строковый.

action- Вид вывода из оборота. Тип числовой. Обязательное. Вид вывода из оборота: 1 - Розничная продажа; 2 - Экспорт в страны ЕАЭС; 3 - Экспорт за пределы стран ЕАЭС; 4 - Кредитный договор; 5 - Порча и утеря товара; 6 - Безвозмездная передача;

7 - Возврат физическому лицу; 8 - Банкротство или ликвидация ЮЛ/ИП; 9 - Реализации конфискованных товаров; 10 - Использование на предприятии; 11 - Реализация по договору рассрочки; 12 - В процессе реализации по договору рассрочки;

action_date- Дата вывода из оборота. Тип строковый. Обязательное.

document_type- Вид документа. Тип числовой. Необязательное.

document_number- Номер документа. Тип строковый. Обязательное.

document_date- Дата документа. Тип строковый. Обязательное.

cis- Киз. Тип строковый. Обязательное.

product_cost- Стоимость товара. Тип BigDecimal. Обязательное.

product_tax- Сумма налога. Тип BigDecimal. Обязательное.

Формат XML

xsd схема:

```
<?xml version="1.0" encoding="UTF-8"?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema"
elementFormDefault="qualified" attributeFormDefault="unqualified">
  <xs:include schemaLocation="../LP_base_types.xsd"/>
  <xs:element name="retail_sale" type="retail_sale" />
  <xs:complexType name="retail_sale">
    <xs:annotation>
      <xs:documentation>          </xs:documentation>
    </xs:annotation>
    <xs:sequence>
      <xs:element name="TRADE_PARTICIPANT_INN"
type="TRADE_PARTICIPANT_INN_type">
        <xs:annotation>
          <xs:documentation> ,      </xs:documentation>
        </xs:annotation>
      </xs:element>
      <xs:element name="action" type="action_type">
        <xs:annotation>
          <xs:documentation>
            :
            1 -
            2 -
            3 -
            4 -
            5 -
            6 -
            7 -      (          )
            8 -      /
            9 -
            10 -
            11 -
```

```

    12 -
    </xs:documentation>
  </xs:annotation>
</xs:element>
<xs:element name="action_date" type="date_type">
  <xs:annotation>
    <xs:documentation>    </xs:documentation>
  </xs:annotation>
</xs:element>
<xs:element name="doc_type" type="doc_type_enum">
  <xs:annotation>
    <xs:documentation>

    1 -
    2 -    ( )
    3 -
    4 -
    5 -
    6 - -
    7 -
    </xs:documentation>
  </xs:annotation>
</xs:element>
<xs:element name="doc_name" type="string200_type" minOccurs="0">
  <xs:annotation>
    <xs:documentation> </xs:documentation>
  </xs:annotation>
</xs:element>
<xs:element name="doc_number" type="string200_type">
  <xs:annotation>
    <xs:documentation> </xs:documentation>
  </xs:annotation>
</xs:element>
<xs:element name="doc_date" type="date_type">
  <xs:annotation>
    <xs:documentation> </xs:documentation>
  </xs:annotation>
</xs:element>
<xs:element name="conclusion">
  <xs:complexType>
    <xs:sequence>
      <xs:element name="detail" maxOccurs="unbounded">
        <xs:annotation>
          <xs:documentation>    </xs:documentation>
        </xs:annotation>
        <xs:complexType>
          <xs:sequence>
            <xs:element name="sign_num" type="gs1_uit_type">
              <xs:annotation>
                <xs:documentation> /</xs:documentation>

```

```
    </xs:annotation>
  </xs:element>
  <xs:element name="cost" type="price_type" minOccurs="0">
    <xs:annotation>
      <xs:documentation> ( . )</xs:documentation>
    </xs:annotation>
  </xs:element>
  <xs:element name="vat_value" type="price_type" minOccurs="0">
    <xs:annotation>
      <xs:documentation> ( )</xs:documentation>
    </xs:annotation>
  </xs:element>
</xs:sequence>
</xs:complexType>
</xs:element>
</xs:sequence>
</xs:complexType>
</xs:element>
</xs:sequence>
<xs:attribute name="action_id" type="xs:int" use="required"
```

```
fixed="15" />
  </xs:complexType>
</xs:schema>
```

Пример:

```
<retail_sale_inst action_id="15" version="2">
  <trade_participant_inn>7724211288</trade_participant_inn>
  <action_inst>11</action_inst>
  <action_date>09.12.2018</action_date>
  <contract_inst_number>12345</contract_inst_number>
  <contract_inst_date>10.12.2018</contract_inst_date>
  <contract_sum_value>50000</contract_sum_value>
  <contract_products_amount>2</contract_products_amount>
  <contract_start_date>01.02.2019</contract_start_date>
  <contract_end_date>20.02.2019</contract_end_date>
  <prime_doc_inst_type>1</prime_doc_inst_type>
  <prime_doc_inst_number>654321</prime_doc_inst_number>
  <prime_doc_inst_date>08.12.2018</prime_doc_inst_date>
  <sum_value>10000</sum_value>
  <vat_sum_value>1800</vat_sum_value>
  <last_payment>>true</last_payment>
  <products_list>
    <product>
      <uit_38>0104059155704529215KPiBCnWJQZqW2406405</uit_38>
    </product>
    <product>
      <uit_38>0104059155704529225KPiBCnWJQZqW2406405</uit_38>
    </product>
  </products_list>
</retail_sale_inst>
```

Формат CSV

Пример:

```
PARTICIPANT_INN,empty_index,ACTION_TYPE
,ACTION_DATE,DOC_TYPE,DOC_NUM,DOC_DATE
7703799090,
,1,09.12.2018,1,document_view_1,09.12.2018
PRODUCT_CIS,PRODUCT_COST_INDEX = 1,PRODUCT_TAX_INDEX
12345612345612345612345612345612345678,"1000","10.05"
```

Приложение. Документ вывода товара из оборота при реализации договора рассрочки

Формат JSON

Пример:

```
{
  "trade_participant_inn": "<STRING>", : "7724211288"
  "action_inst": <INTEGER>, : 11
  "action_date": "2018-12-08T21:00:00.000Z",
  "contract_inst_number": "<STRING>",
  "contract_inst_date": "2018-12-09T21:00:00.000Z",
  "contract_sum_value": <BigDecimal>, : 50000
  "contract_products_amount": <INTEGER>, : 2
  "contract_start_date": "2018-01-31T21:00:00.000Z",
  "contract_end_date": "2018-02-19T21:00:00.000Z",
  "prime_doc_inst_type": <INTEGER>,
  "prime_doc_inst_number": "<STRING>",
  "prime_doc_inst_date": "2018-12-07T21:00:00.000Z",
  "sum_value": <BigDecimal>,
  "vat_sum_value": 1800,
  "last_payment": true,
  "products": [
    {
      "uit_38": <STRING> : "0104059155704529215KPiBCnWJQZqW2406405"
    },
    {
      "uit_38": "0104059155704529225KPiBCnWJQZqW2406405"
    }
  ]
}
```

Атрибутивный состав:

trade_participant_inn- ИНН участника оборота, осуществляющего вывод из оборота. Тип строковый. Обязательное.

action_inst- Вид вывода из оборота. Тип числовой. Должен быть равен 11 [причина вывода : 11 - Договор рассрочки]. Обязательное.

action_date- Дата вывода из оборота. Тип строковый. Обязательное.

contract_inst_number- Номер договора рассрочки. Тип строковый. Необязательное.

contract_inst_date- Дата договора рассрочки. Тип строковый. Необязательное.

contract_sum_value- Сумма рассрочки по договору. BigDecimal. Необязательное.

contract_products_amount- Количество товаров по договору. Тип числовой. Необязательное.

contract_start_date- Дата начала рассрочки. Тип числовой. Обязательное.

contract_end_date- Дата окончания рассрочки. Тип числовой. Обязательное.

prime_doc_inst_type- Вид документа. Тип числовой. Вид первичного документа: 1 - Фискальный документ; 2 - Платёжное поручение; Обязательное.

prime_doc_inst_number- Номер фискального документа. Тип строковый. Обязательное.

prime_doc_inst_date- Дата фискального документа. Тип строковый. Обязательное.

sum_value- Сумма платежа. Тип BigDecimal. Необязательное.

vat_sum_value- Сумма НДС. Тип BigDecimal. Необязательное.

last_payment- Признак последнего платежа. Тип boolean.

uit_38- УИТ товара. Тип строковый.

Формат XML

xsd схема:

```
<?xml version="1.0" encoding="UTF-8"?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema"
elementFormDefault="qualified" attributeFormDefault="unqualified">
  <xs:include schemaLocation="../LP_base_types.xsd"/>
  <xs:element name="retail_sale_inst" type="retail_sale_inst" />
  <xs:complexType name="retail_sale_inst">
    <xs:annotation>
      <xs:documentation>          </xs:documentation>
    </xs:annotation>
    <xs:sequence>
      <xs:element name="trade_participant_inn"
type="TRADE_PARTICIPANT_INN_type">
        <xs:annotation>
          <xs:documentation> ,      </xs:documentation>
        </xs:annotation>
      </xs:element>
      <xs:element name="action_inst" type="action_inst_type">
        <xs:annotation>
          <xs:documentation>
            :
            11 -
          </xs:documentation>
        </xs:annotation>
      </xs:element>
      <xs:element name="action_date" type="date_type">
        <xs:annotation>
          <xs:documentation> </xs:documentation>
        </xs:annotation>
      </xs:element>
      <xs:element name="contract_inst_number"
type="document_number_200_type" minOccurs="0">
        <xs:annotation>
          <xs:documentation> </xs:documentation>
        </xs:annotation>
      </xs:element>
      <xs:element name="contract_inst_date" type="date_type" minOccurs="0">
        <xs:annotation>
          <xs:documentation> </xs:documentation>
        </xs:annotation>
      </xs:element>
      <xs:element name="contract_sum_value" type="price_type"
minOccurs="0">
        <xs:annotation>
```

```

    <xs:documentation>    </xs:documentation>
  </xs:annotation>
</xs:element>
<xs:element name="contract_products_amount" minOccurs="0">
  <xs:annotation>
    <xs:documentation>    </xs:documentation>
  </xs:annotation>
  <xs:simpleType>
    <xs:restriction base="xs:decimal">
      <xs:totalDigits value="19"/>
      <xs:fractionDigits value="0"/>
      <xs:minInclusive value="1"/>
    </xs:restriction>
  </xs:simpleType>
</xs:element>
<xs:element name="contract_start_date" type="date_type">
  <xs:annotation>
    <xs:documentation>    </xs:documentation>
  </xs:annotation>
</xs:element>
<xs:element name="contract_end_date" type="date_type">
  <xs:annotation>
    <xs:documentation>    </xs:documentation>
  </xs:annotation>
</xs:element>
<xs:element name="prime_doc_inst_type" type="prime_doc_inst_type">
  <xs:annotation>
    <xs:documentation>

    1 -
    2 -
  </xs:documentation>
  </xs:annotation>
</xs:element>
<xs:element name="prime_doc_inst_number"
type="document_number_200_type">
  <xs:annotation>
    <xs:documentation>    </xs:documentation>
  </xs:annotation>
</xs:element>
<xs:element name="prime_doc_inst_date" type="date_type">
  <xs:annotation>
    <xs:documentation>    </xs:documentation>
  </xs:annotation>
</xs:element>
<xs:element name="sum_value" type="price_type" minOccurs="0">
  <xs:annotation>
    <xs:documentation>    </xs:documentation>
  </xs:annotation>
</xs:element>

```



```
<xs:element name="vat_sum_value" type="price_type" minOccurs="0">
  <xs:annotation>
    <xs:documentation> </xs:documentation>
  </xs:annotation>
</xs:element>
<xs:element name="last_payment" type="xs:boolean" minOccurs="0">
  <xs:annotation>
    <xs:documentation> </xs:documentation>
  </xs:annotation>
</xs:element>
<xs:element name="products_list">
  <xs:complexType>
    <xs:sequence>
      <xs:element name="product" maxOccurs="unbounded" minOccurs="0">
        <xs:annotation>
          <xs:documentation> ( products)</xs:documentation>
        </xs:annotation>
        <xs:complexType>
          <xs:sequence>
            <xs:element name="uit_38" type="uit_38_type">
              <xs:annotation>
                <xs:documentation></xs:documentation>
              </xs:annotation>
            </xs:element>
          </xs:sequence>
        </xs:complexType>
      </xs:element>
    </xs:sequence>
  </xs:complexType>
  <xs:attribute name="action_id" type="xs:int" use="required"
fixed="15"/>

```


Приложение. Документ ввода в оборот, произведенного на территории РФ (производство сторонней организацией)

Формат JSON

Пример:

```
{
  "producer_inn": "1111111111",
  "owner_inn": "2222222222",
  "production_date": "2019-04-10T00:00:00.000Z",
  "marking_type": "PRODUCT",
  "production_order": "CONTRACT_PRODUCTION",
  "products_list": [
    {
      "uit": "12345678912345678912345678912345678912",
      "tnved_code": "6401123456",
      "production_date": "2019-04-10T00:00:00.000Z",
      "certificate_doc_type": "CONFORMITY_CERT",
      "certificate_doc_number": "12345",
      "certificate_doc_date": "2019-04-10T00:00:00.000Z"
    }
  ]
}
```

Атрибутивный состав:

Имя	Описание	Тип	Обязательный
producer_inn	ИНН производителя или импортера товара	строковый	да
owner_inn	ИНН собственника товара	строковый	да
production_date	Дата производства	строковый	да
marking_type	Вид маркировки	строковый	да
production_order	Тип производственного заказа	строковый	да
uit	Уникальный идентификатор товара (УИТ)	строковый	Обязательное, если не указано uitu
uitu	Уникальный идентификатор транспортной упаковки (УИТУ)	строковый	Обязательное, если не указано uit
tnved_code	10-ти значный код ТН ВЭД ЕАЭС товара	строковый	да
production_date	Дата производства товаров (если отличается от указанной в общих сведениях о вводе товаров в оборот)	строковый	нет
certificate_doc_type	Виды документов обязательной	строковый	да
certificate_doc_number	Номер сертификата соответствия	строковый	да
certificate_doc_date	Дата сертификата соответствия	строковый	да

Формат XML

xsd схема:

```

<?xml version="1.0" encoding="UTF-8"?>
<xs:schema elementFormDefault="qualified"
xmlns:xs="http://www.w3.org/2001/XMLSchema">
  <xs:include schemaLocation="../../../LP_base_types.xsd"/>
  <xs:element name="vvod_contract_production"
type="vvod_contract_production"/>
  <xs:complexType name="vvod_contract_production">
    <xs:annotation>
      <xs:documentation> , </xs:documentation>
    </xs:annotation>
    <xs:sequence>
      <xs:element name="DocType">
        <xs:simpleType>
          <xs:restriction base="xs:string">
            <xs:enumeration value="vvod_5_2"/>
          </xs:restriction>
        </xs:simpleType>
      </xs:element>
      <xs:element name="producer_inn" type="TRADE_PARTICIPANT_INN_type">
        <xs:annotation>
          <xs:documentation> </xs:documentation>
        </xs:annotation>
      </xs:element>
      <xs:element name="owner_inn" type="TRADE_PARTICIPANT_INN_type">
        <xs:annotation>
          <xs:documentation> </xs:documentation>
        </xs:annotation>
      </xs:element>
      <xs:element name="production_date" type="date_type">
        <xs:annotation>
          <xs:documentation> </xs:documentation>
        </xs:annotation>
      </xs:element>
      <xs:element name="marking_type">
        <xs:annotation>
          <xs:documentation>
            - PRODUCT_PACKAGE
            - PRODUCT
            - PRODUCT_LABEL
          </xs:documentation>
        </xs:annotation>
        <xs:simpleType>
          <xs:restriction base="xs:string">
            <xs:enumeration value="PRODUCT_PACKAGE"/>
            <xs:enumeration value="PRODUCT"/>
            <xs:enumeration value="PRODUCT_LABEL"/>
          </xs:restriction>
        </xs:simpleType>
      </xs:element>
      <xs:element name="production_order">

```

```

<xs:annotation>
  <xs:documentation>
    - OWN_PRODUCTION
    - CONTRACT_PRODUCTION
  </xs:documentation>
</xs:annotation>
<xs:simpleType>
  <xs:restriction base="xs:string">
    <xs:enumeration value="OWN_PRODUCTION"/>
    <xs:enumeration value="CONTRACT_PRODUCTION"/>
  </xs:restriction>
</xs:simpleType>
</xs:element>
<xs:element name="products_list">
  <xs:complexType>
    <xs:sequence>
      <xs:element name="product" maxOccurs="unbounded">
        <xs:complexType>
          <xs:sequence>
            <xs:element name="uit" type="gsl_uit_type" minOccurs="0">
              <xs:annotation>
                <xs:documentation></xs:documentation>
              </xs:annotation>
            </xs:element>
            <xs:element name="uitu" type="gsl_uitu_type" minOccurs="0">
              <xs:annotation>
                <xs:documentation></xs:documentation>
              </xs:annotation>
            </xs:element>
            <xs:element name="TNVED_code" type="TNVED_code">
              <xs:annotation>
                <xs:documentation> </xs:documentation>
              </xs:annotation>
            </xs:element>
            <xs:element name="production_date" type="date_type">
              <xs:annotation>
                <xs:documentation> </xs:documentation>
              </xs:annotation>
            </xs:element>
            <xs:element name="certificate_doc_type">
              <xs:annotation>
                <xs:documentation>
                  - CONFORMITY_CERT
                  - CONFORMITY_DECLAR</xs:documentation>
                </xs:annotation>
              <xs:simpleType>
                <xs:restriction
base="xs:string">
                <xs:enumeration
value="CONFORMITY_CERT"/>

```

```
value="CONFORMITY_DECLAR"/>
                                <xs:enumeration
                                </xs:restriction>
                                </xs:simpleType>
</xs:element>
<xs:element name="certificate_doc_number">
  <xs:annotation>
    <xs:documentation>    </xs:documentation>
  </xs:annotation>
  <xs:simpleType>
    <xs:restriction base="xs:string">
      <xs:minLength value="1"/>
      <xs:maxLength value="255"/>
    </xs:restriction>
  </xs:simpleType>
</xs:element>
<xs:element name="certificate_doc_date" type="date_type">
  <xs:annotation>
    <xs:documentation>    </xs:documentation>
  </xs:annotation>
</xs:element>
</xs:sequence>
</xs:complexType>
</xs:element>
</xs:sequence>
</xs:complexType>
</xs:element>
</xs:sequence>
  <xs:attribute name="action_id" type="xs:string" use="required"
fixed="5.2"/>
```

```
<xs:attribute name="version" type="xs:int" use="required" fixed="1"/>
</xs:complexType>
</xs:schema>
```

Пример:

```
<?xml version="1.0" encoding="UTF-8"?>
<vvod_contract_production>
  <doc_type>vvod_5_2</doc_type>
  <producer_inn>1111111111</producer_inn>
  <owner_inn>2222222222</owner_inn>
  <production_date>10.04.2019</production_date>
  <marking_type>PRODUCT</marking_type>
  <production_order>CONTRACT_PRODUCTION</production_order>
  <products_list>
    <product>
      <uit>12345678912345678912345678912345678912</uit>
      <tnved_code>6401</tnved_code>
      <production_date>10.04.2019</production_date>
      <certificate_doc_type>CONFORMITY_CERT</certificate_doc_type>
      <certificate_doc_number>12345</certificate_doc_number>
      <certificate_doc_date>10.04.2019</certificate_doc_date>
    </product>
  </products_list>
</vvod_contract_production>
```

Формат CSV

Пример:

```
' ' ' ' ' '
1111111111,2222222222,10.04.2019,PRODUCT,CONTRACT_PRODUCTION
' ' ' ' ' '
12345678912345678912345678912345678912,,6401,10.04.2019,CONFORMITY_CERT,
12345,10.04.2019
```

Приложение. Документ ввода в оборот товара, полученного от физических лиц, не являющихся индивидуальными предпринимателями, в том числе при комиссионной торговле

Формат JSON

Пример:

```

{
  "participant_inn": "1111111111",
  "marking_type": "PRODUCT",
  "products_list": [
    {
      "uit": "12345678912345678912345678912345678912"
    }
  ]
}

```

Атрибутивный состав:

Имя	Описание	Тип	Обязательный
participant_inn	ИНН участника оборота	строковый	да
marking_type	Вид маркировки: PRODUCT_PACKAGE (На потребительскую упаковку), PRODUCT (На товар), PRODUCT_LABEL (На товарный ярлык)	строковый	да
uit	Уникальный идентификатор товара (УИТ)	строковый	да

Формат XML

xsd схема:

```

<?xml version="1.0" encoding="UTF-8"?>
<xs:schema elementFormDefault="qualified"
xmlns:xs="http://www.w3.org/2001/XMLSchema">
  <xs:include schemaLocation="../../../LP_base_types.xsd"/>
  <xs:element name="vvod_individuals" type="vvod_individuals"/>
  <xs:complexType name="vvod_individuals">
    <xs:annotation>
      <xs:documentation> , </xs:documentation>
    </xs:annotation>
    <xs:sequence>
      <xs:element name="doc_type">
        <xs:simpleType>
          <xs:restriction base="xs:string">
            <xs:enumeration value="vvod_individuals"/>
          </xs:restriction>
        </xs:simpleType>
      </xs:element>
      <xs:element name="participant_inn"
type="TRADE_PARTICIPANT_INN_type">
        <xs:annotation>
          <xs:documentation> </xs:documentation>
        </xs:annotation>
      </xs:element>
      <xs:element name="marking_type">
        <xs:annotation>

```



```

    <xs:documentation>
      - PRODUCT_PACKAGE
      - PRODUCT
      - PRODUCT_LABEL
    </xs:documentation>
  </xs:annotation>
  <xs:simpleType>
    <xs:restriction base="xs:string">
      <xs:enumeration value="PRODUCT_PACKAGE"/>
      <xs:enumeration value="PRODUCT"/>
      <xs:enumeration value="PRODUCT_LABEL"/>
    </xs:restriction>
  </xs:simpleType>
</xs:element>
<xs:element name="products_list">
  <xs:complexType>
    <xs:sequence>
      <xs:element name="product" maxOccurs="unbounded">
        <xs:complexType>
          <xs:sequence>
            <xs:element name="uit" type="gs1_uit_type" minOccurs="0">
              <xs:annotation>
                <xs:documentation></xs:documentation>
              </xs:annotation>
            </xs:element>
            <xs:element name="uitu" type="gs1_uitu_type" minOccurs="0">
              <xs:annotation>
                <xs:documentation></xs:documentation>
              </xs:annotation>
            </xs:element>
          </xs:sequence>
        </xs:complexType>
      </xs:element>
    </xs:sequence>
  </xs:complexType>
  <xs:attribute name="action_id" type="xs:string" use="required"
fixed="5.3"/>
  <xs:attribute name="version" type="xs:int" use="required" fixed="1"/>
</xs:complexType>

```

```
</xs:schema>
```

Пример:

```
<?xml version="1.0" encoding="UTF-8"?>
<vvod_individuals>
  <doc_type>vvod_5_2</doc_type>
  <participant_inn>1111111111</participant_inn>
  <marking_type>PRODUCT</marking_type>
  <products_list>
    <product>
      <uit>12345678912345678912345678912345678912</uit>
    </product>
  </products_list>
</vvod_individuals>
```

Приложение. Документ списания КМ с/без СИ

Формат JSON

Пример:

```
{
  "trade_participant_inn": "7724211288",
  "cancellation_reason": "KM_SPOILED",
  "cancellation_doc_date": "2019-02-09T21:00:00.000Z",
  "cancellation_doc_number": "document number",
  "km_list": [
    {
      "uit": "0104059155704529215KPiBCnWJQZqW2406405",
      "uitu": "123456789123456789"
      "cancellation_reason": "KM_SPOILED"
    }
  ]
}
```

Атрибутивный состав:

trade_participant_inn- ИНН участника оборота, осуществляющего списание кодов маркировки. Тип строковый. Обязательное.

cancelation_reason- Причина списания. Тип строковый. Не обязательное

cancellation_doc_date- дата документа, подтверждающего списание. Обязательное

cancellation_doc_number-Номер документа, подтверждающего списание. Тип строковый. Обязательное

km_list - Список Км на списание. Количество элементов больше нуля

uit - УИТ. Тип строковый. Обязательное, если не указан УИТУ.

uitu-УИТУ. Тип строковый. Обязательное, если не указан УИТ.

cancellation_reason - Причина списания. Возможные значения: KM_SPOILED/KM_LOST/KM_DESTROYED. Не обязательное

Формат XML

xsd схема:

```
<?xml version="1.0" encoding="UTF-8"?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema"
elementFormDefault="qualified" attributeFormDefault="unqualified">
  <xs:include schemaLocation="../LP_base_types.xsd"/>
  <xs:element name="km_cancellation" type="km_cancellation"/>
  <xs:complexType name="km_cancellation">
    <xs:annotation>
      <xs:documentation> </xs:documentation>
    </xs:annotation>
    <xs:sequence>
      <xs:element name="trade_participant_inn"
type="TRADE_PARTICIPANT_INN_type">
        <xs:annotation>
          <xs:documentation> , </xs:documentation>
        </xs:annotation>
      </xs:element>
      <xs:element name="cancellation_reason" minOccurs="0">
        <xs:annotation>
          <xs:documentation>
            KM_SPOILED -
            KM_LOST -
            KM_DESTROYED -
          </xs:documentation>
        </xs:annotation>
        <xs:simpleType>
          <xs:restriction base="xs:string">
            <xs:enumeration value="KM_SPOILED"/>
            <xs:enumeration value="KM_LOST"/>
            <xs:enumeration value="KM_DESTROYED"/>
          </xs:restriction>
        </xs:simpleType>
      </xs:element>
      <xs:element name="cancellation_doc_date" type="date_type">
        <xs:annotation>
          <xs:documentation> , </xs:documentation>
        </xs:annotation>
      </xs:element>
      <xs:element name="cancellation_doc_number" type="string255_type">
        <xs:annotation>
          <xs:documentation> , </xs:documentation>
        </xs:annotation>
      </xs:element>
    </xs:sequence>
  </xs:complexType>
</xs:schema>
```

```

<xs:complexType>
  <xs:sequence>
    <xs:element name="km" maxOccurs="unbounded">
      <xs:annotation>
        <xs:documentation> </xs:documentation>
      </xs:annotation>
      <xs:complexType>
        <xs:sequence>
          <xs:element name="uit" type="gsl_uit_type" minOccurs="0">
            <xs:annotation>
              <xs:documentation></xs:documentation>
            </xs:annotation>
          </xs:element>
          <xs:element name="uitu" type="gsl_uitu_type" minOccurs="0">
            <xs:annotation>
              <xs:documentation></xs:documentation>
            </xs:annotation>
          </xs:element>
          <xs:element name="cancellation_reason">
            <xs:annotation>
              <xs:documentation>
                KM_SPOILED -
                KM_LOST -
                KM_DESTROYED -
              </xs:documentation>
            </xs:annotation>
            <xs:simpleType>
              <xs:restriction base="xs:string">
                <xs:enumeration value="KM_SPOILED"/>
                <xs:enumeration value="KM_LOST"/>
                <xs:enumeration value="KM_DESTROYED"/>
              </xs:restriction>
            </xs:simpleType>
          </xs:element>
        </xs:sequence>
      </xs:complexType>
    </xs:element>
  </xs:sequence>
  <xs:complexType>
    <xs:sequence>
      <xs:element name="action_id" type="xs:int" use="required"
        fixed="14"/>
    </xs:sequence>
  </xs:complexType>
</xs:sequence>
</xs:complexType>

```

```
<xs:attribute name="version" type="xs:int" use="required" fixed="1"/>
</xs:complexType>
</xs:schema>
```

Пример:

```
<km_cancellation action_id="14" version="1">
  <trade_participant_inn>7724211288</trade_participant_inn>
  <cancellation_reason>KM_SPOILED</cancellation_reason>
  <cancellation_doc_date>10.02.2019</cancellation_doc_date>
  <cancellation_doc_number>document number</cancellation_doc_number>
  <km_list>
    <km>
      <uit>0104059155704529215KPiBCnWJQZqW2406405</uit>
      <cancellation_reason>KM_SPOILED</cancellation_reason>
    </km>
  </km_list>
</km_cancellation>
```

Формат CSV

Пример:

```
' ' ' '
7724211288 ,KM_SPOILED,10.02.2019,document number
' ' '
0104059155704529215KPiBCnWJQZqW2406405 , ,KM_SPOILED
```

Приложение. Документ перемаркировки товара

Формат JSON

Пример:

```

{
  "participant_inn": "inn",
  "marking_type": "PRODUCT_PACKAGE",
  "remarking_date": "2018-09-17T21:00:00.000Z",
  "remarking_cause": "DAMAGED_OR_LOST",
  "products": [
    {
      "certificate_document": "CONFORMITY_CERT",
      "certificate_document_number": "certificateDocumentNumber",
      "certificate_document_date": "2018-09-17T21:00:00.000Z",
      "last_uin": "lastUin",
      "new_uin": "newUin",
      "marking_type": "PRODUCT_PACKAGE",
      "remarking_date": "2018-09-17T21:00:00.000Z",
      "remarking_cause": "DAMAGED_OR_LOST"
    }
  ]
}

```

Атрибутивный состав:

Имя	Описание	Тип	Обязательный
participant_inn	ИНН участника оборота	строковый	да
marking_type	Вид маркировки: PRODUCT_PACKAGE (На потребительскую упаковку), PRODUCT (На товар), PRODUCT_LABEL (На товарный ярлык)	строковый	да
remarking_date	Дата повторной маркировки	строковый	да
remarking_cause	Причина повторной маркировки: DAMAGED_OR_LOST (Испорчено либо утеряно СИ с КМ), ERROR_DETECTED (Выявлены ошибки описания товара)		
certificate_document	Вид документа обязательной сертификации. CONFORMITY_CERT - сертификат соответствия, CONFORMITY_DECLAR - декларация соответствия	строковый	да
certificate_document_number	Номер документа обязательной сертификации	строковый	да
certificate_document_date	Дата документа обязательной сертификации	строковый	да
last_uin	Предыдущий уникальный идентификатор товара	строковый	да
new_uin	Новый уникальный идентификатор товара	строковый	да
marking_type	Вид маркировки: PRODUCT_PACKAGE (На потребительскую упаковку), PRODUCT (На товар), PRODUCT_LABEL (На товарный ярлык)	строковый	нет
remarking_date	Дата повторной маркировки	строковый	нет
remarking_cause	Причина повторной маркировки: DAMAGED_OR_LOST (Испорчено либо утеряно СИ с КМ), ERROR_DETECTED (Выявлены ошибки описания товара)	строковый	нет

Формат XML

xsd схема:

```

<?xml version="1.0" encoding="UTF-8"?>
<xs:schema elementFormDefault="qualified"
xmlns:xs="http://www.w3.org/2001/XMLSchema">

```

```

<xs:element name="RemReq" type="RemReq" />

<xs:simpleType name="date_type">
  <xs:annotation>
    <xs:documentation>    ..</xs:documentation>
  </xs:annotation>
  <xs:restriction base="xs:string">
    <xs:length value="10" />
    <xs:pattern
value="(0[1-9]{1}|[1-2]{1}[0-9]{1}|3[0-1]{1})\.(0[1-9]{1}|1[0-2]{1})\.(1
[0-9]{3}|20[0-9]{2})" />
    </xs:restriction>
  </xs:simpleType>

<xs:simpleType name="gs1_uit_type">
  <xs:annotation>
    <xs:documentation> </xs:documentation>
  </xs:annotation>
  <xs:restriction base="xs:string">
    <xs:length value="31" />
    <xs:pattern value="[A-Za-z0-9]{31}" />
  </xs:restriction>
</xs:simpleType>

<xs:complexType name="RemReq">
  <xs:annotation>
    <xs:documentation>    </xs:documentation>
  </xs:annotation>
  <xs:sequence>
    <xs:element name="PARTICIPANT_INN">
      <xs:annotation>
        <xs:documentation></xs:documentation>
      </xs:annotation>
      <xs:simpleType>
        <xs:restriction base="xs:string">
          <xs:minLength value="10" />
          <xs:maxLength value="12" />
          <xs:pattern
value="([0-9]{1}[1-9]{1}|[1-9]{1}[0-9]{1})[0-9]{8})|((([0-9]{1}[1-9]{1}|
[1-9]{1}[0-9]{1})[0-9]{10})" />
          </xs:restriction>
        </xs:simpleType>
      </xs:element>
    <xs:element name="PARTICIPANT_NAME">
      <xs:annotation>
        <xs:documentation> </xs:documentation>
      </xs:annotation>
      <xs:simpleType>
        <xs:restriction base="xs:string">

```

```

    <xs:maxLength value="1000"/>
  </xs:restriction>
</xs:simpleType>
</xs:element>
<xs:element name="MARKING_TYPE">
  <xs:annotation>
    <xs:documentation>
      1 -
      2 -
      3 -    </xs:documentation>
    </xs:annotation>
    <xs:simpleType>
      <xs:restriction base="xs:int">
        <xs:enumeration value="1"/>
        <xs:enumeration value="2"/>
        <xs:enumeration value="3"/>
      </xs:restriction>
    </xs:simpleType>
  </xs:element>
<xs:element name="REMARKING_DATE" type="date_type">
  <xs:annotation>
    <xs:documentation>    </xs:documentation>
  </xs:annotation>
</xs:element>
<xs:element name="REMARKING_CAUSE">
  <xs:annotation>
    <xs:documentation>
      1 -
      2 -    </xs:documentation>
    </xs:annotation>
    <xs:simpleType>
      <xs:restriction base="xs:int">
        <xs:enumeration value="1"/>
        <xs:enumeration value="2"/>
      </xs:restriction>
    </xs:simpleType>
  </xs:element>
<xs:element name="PRODUCT_DETAILS">
  <xs:annotation>
    <xs:documentation>    </xs:documentation>
  </xs:annotation>
  <xs:complexType>
    <xs:sequence>
      <xs:element name="PRODUCT" maxOccurs="unbounded">
        <xs:complexType>
          <xs:sequence>
            <xs:element name="MARKING_TYPE" minOccurs="0">
              <xs:annotation>
                <xs:documentation>
                  1 -

```



```

    2 -
    3 - </xs:documentation>
</xs:annotation>
<xs:simpleType>
  <xs:restriction base="xs:int">
    <xs:enumeration value="1"/>
    <xs:enumeration value="2"/>
    <xs:enumeration value="3"/>
  </xs:restriction>
</xs:simpleType>
</xs:element>
<xs:element name="REMARKING_DATE" minOccurs="0"
type="date_type">
  <xs:annotation>
    <xs:documentation> </xs:documentation>
  </xs:annotation>
</xs:element>
<xs:element name="REMARKING_CAUSE" minOccurs="0">
  <xs:annotation>
    <xs:documentation>
    1 -
    2 - </xs:documentation>
  </xs:annotation>
  <xs:simpleType>
    <xs:restriction base="xs:int">
      <xs:enumeration value="1"/>
      <xs:enumeration value="2"/>
    </xs:restriction>
  </xs:simpleType>
</xs:element>
<xs:element name="LAST_UIT" type="gs1_uit_type">
  <xs:annotation>
    <xs:documentation> </xs:documentation>
  </xs:annotation>
</xs:element>
<xs:element name="NEW_UIT" type="gs1_uit_type">
  <xs:annotation>
    <xs:documentation> </xs:documentation>
  </xs:annotation>
</xs:element>
<xs:element name="PROD_CERT_DOC_TYPE" minOccurs="0">
  <xs:annotation>
    <xs:documentation>
    1 -
    2 - </xs:documentation>
  </xs:annotation>
  <xs:simpleType>
    <xs:restriction base="xs:int">
      <xs:enumeration value="1"/>
      <xs:enumeration value="2"/>
    </xs:restriction>
  </xs:simpleType>
</xs:element>

```

```
        </xs:restriction>
    </xs:simpleType>
</xs:element>
<xs:element name="PROD_CERT_DOC_NUM" minOccurs="0">
    <xs:annotation>
        <xs:documentation>    </xs:documentation>
    </xs:annotation>
    <xs:simpleType>
        <xs:restriction base="xs:string">
            <xs:maxLength value="1000"/>
        </xs:restriction>
    </xs:simpleType>
</xs:element>
<xs:element name="PROD_CERT_DOC_DATE" type="date_type"
minOccurs="0">
    <xs:annotation>
        <xs:documentation>    </xs:documentation>
    </xs:annotation>
</xs:element>
</xs:sequence>
</xs:complexType>
</xs:element>
</xs:sequence>
</xs:complexType>
</xs:element>
</xs:sequence>
<xs:attribute name="action_id" type="xs:int" use="required"
```

```
fixed="09" />
</xs:complexType>
</xs:schema>
```

Пример:

```
<?xml version="1.0" encoding="UTF-8"?>
<RemReq action_id="09">
  <PARTICIPANT_INN>7703799090</PARTICIPANT_INN>
  <PARTICIPANT_NAME>string</PARTICIPANT_NAME>
  <MARKING_TYPE>1</MARKING_TYPE>
  <REMARKING_DATE>01.01.2018</REMARKING_DATE>
  <REMARKING_CAUSE>1</REMARKING_CAUSE>
  <PRODUCT_DETAILS>
    <!--1 or more repetitions:-->
    <PRODUCT>
      <LAST_UIT>LAST_UIT</LAST_UIT>
      <NEW_UIT>NEW_UIT</NEW_UIT>
      <PROD_CERT_DOC_TYPE>1</PROD_CERT_DOC_TYPE>
      <PROD_CERT_DOC_NUM>string</PROD_CERT_DOC_NUM>
      <PROD_CERT_DOC_DATE>01.01.2018</PROD_CERT_DOC_DATE>
      <MARKING_TYPE>1</MARKING_TYPE>
      <REMARKING_DATE>01.01.2018</REMARKING_DATE>
      <REMARKING_CAUSE>1</REMARKING_CAUSE>
    </PRODUCT>
  </PRODUCT_DETAILS>
</RemReq>
```

Формат CSV

Пример:

```
' ' ' ' ' '
participantInn,participant, ,01.01.2019,
' ' ' ' ' ' ' '
,,,LAST_UIT,NEW_UIT, ,112233,02.02.2018
,03.03.2018,,,LAST_UIT,NEW_UIT, ,112233,02.02.2018
```

Примеры

Аутентификация по УКЭП

Пример получения токена при авторизации 1С

```

// sThumbprint - , ; ,
// 195934d72dcdf69149901d6632aca4562d8806d8
// Base64
// bDetached - / - /
// 60
(, sThumbprint, bDetached) CADESCOM_BASE64_TO_BINARY = 1; // Base64
CADESCOM_CADES_TYPE = 1; //
oSigner = COM("CADESCOM.CPSigner");
// , .
oSigner.Certificate = (sThumbprint);
= ();
oSignedData = COM("CADESCOM.CadesSignedData");
// CadesSignedData
.
oSignedData.ContentEncoding = CADESCOM_BASE64_TO_BINARY;
SignedData.Content = ();
EncodingType = 0;
sSignedMessage = oSignedData.SignCades(oSigner, CADESCOM_CADES_TYPE,
bDetached,
EncodingType);
// . sSignedMessage; // Base64

// - HEX
() = ; // (Com-)
CAPICOM_CURRENT_USER_STORE = 2;
//2 - " " .
CAPICOM_MY_STORE = "My";
// , " " CAPICOM_STORE_OPEN_READ_ONLY = 0; // oStore =
COM("CADESCOM.Store"); //

oStore.Open(CAPICOM_CURRENT_USER_STORE, CAPICOM_MY_STORE,
CAPICOM_STORE_OPEN_READ_ONLY); //
// 1 :
CAPICOM_CERTIFICATE_FIND_SHA1_HASH = 0;
Certificates =
oStore.Certificates.Find(CAPICOM_CERTIFICATE_FIND_SHA1_HASH, );
= Certificates.Item(1);
//2 :
// oStore.Certificates
// = .Thumbprint; //

// () = () // = ;
//;
//;
//;
oStore.Close(); // 61

;

```

Пример получения токена при авторизации REACT JS.

```
import { CadesPluginer as anyName } from "@crpt/cades-pluginer"; //or
import CadesPluginer from "@crpt/cades-pluginer";
//usually you will need 2 methods
CadesPluginer.getFinalCertsArray().then(certs => {
  console.log("certs", certs);
  CadesPluginer.signMessage("lalala", certs[0].certificate).then(signed =>
    console.log("signed", signed);
  );
});
```

Пример получения токена при авторизации Java.

```
String data = ...//
//
KeyStore keyStore = loadKeyStore();//
List<X509CertificateHolder> chain = new ArrayList<>();
List<Certificate> certs =
Arrays.asList(keyStore.getCertificateChain(alias));
certs.forEach(cert -> {
  try {
    chain.add(new X509CertificateHolder(cert.getEncoded()));
  } catch (IOException | CertificateEncodingException e) {
    log.error("Error while building certificate chain", e);
  }
});
PrivateKey privateKey = (PrivateKey) (keyStore.getKey(alias, password));
ByteArrayOutputStream out = new ByteArrayOutputStream();
CAdESSignature signature = new CAdESSignature(false);
signature.setCertificateStore(new CollectionStore(chain));
// .
signature.addSigner(JCP.PROVIDER_NAME, JCP.GOST_DIGEST_OID,
JCP.GOST_EL_DH_OID, privateKey, certs, CAdESType.CAdES_BES, null,
false);
signature.open(out);
signature.update(data);
signature.close();
byte[] signedCode = out.toByteArray();
//
```

Описание методов плагина:

Описание методов плагина:

Метод	Описание
getFinalCertsArray(): Promise<Cert[]>	Init plugin, create store, . Cert: {certificate: {}, info: [] }, certificate is an original cadesplugin Certificate object, info is an array [name, date_from, date_to, serial_number].
signMessage(message, cert, detached): Promise	Sign message with selected certificate.