|  |
| --- |
| **Payguide Acquirer**  **Протокол OpenAPI. Магазинная консоль** |

СОГЛАСОВАНО

| Наименование организации, предприятия | Должность исполнителя | Фамилия, имя, отчество | Подпись | Дата |
| --- | --- | --- | --- | --- |
|  |  |  |  |  |

ЛИСТ РЕГИСТРАЦИИ ИЗМЕНЕНИЙ

| № версии док-та | Дата изменения | Автор изменений | Изменения |
| --- | --- | --- | --- |
| 2.17 | 01.11.2023 | Муравицкая А.В. | Добавлен параметр additiveCardId в таблицу полей ответа на запрос RegisterPaymentRequest в п. 12.1. |
| 2.16 | 27.06.2023 | Муравицкая А.В.  Ягур В.С. | При получении списка транзакций в запрос добавлен параметр: amount (пункт 7.2). |
| 2.15 | 13.06.2023 | Муравицкая А.В. | Обновлен подраздел 6.2 «Получение данных для графиков». Добавлен пример ответа с данными для трёх графиков, если отсутствует параметра status. |
| 2.14 | 06.06.2023 | Муравицкая А.В. | Обновлен подраздел 6.2 «Получение данных для графиков». Добавлены определения введенных пятого и шестого чисел, входящих в массив "координат" точек для трёх графиков.  Обновлен пример ответа с данными для трех графиков. |
| 2.13 | 27.03.2023 | Муравицкая А.В. | В п. 7.1 добавлен тип платежа Mir Pay в таблицу «Параметры запроса»;  В п. 7.2 добавлен тип платежа Mir Pay в таблицу «Поля ответа» |
| 2.12 | 22.08.2022 | Сопот И.В. | В пункте 12.1 поле brand заменено на paymentSystem, и добавлены поля для виртуальной карты |
| 2.11 | 31.05.2022 | Ерчиковская Е.В. | Дополнено описание кода ошибки UNEXPECTED\_REQUEST. Приложение 1. Коды ошибок выполнения запросов. |
| 2.10 | 24.05.2022 | Сопот И.В. | Добавлено новое поле responseCodeDetails в пункт 7.2. |
| 2.9 | 16.03.2022 | Ерчиковская Е.В. | Изменено значение по умолчанию limit в пункте 13.1 «Запрос создания отчёта». |
| 2.8 | 28.02.2022 | Сопот И.В. | В пункт 4 в описание счёта добавлены новые поля. Добавлен пункт 14 «Конфигурация магазина». |
| 2.7 | 01.02.2022 | Анискин Р.А. | Обновлено описание в пунктах 4, 7.1 и 7.2.  Добавлен пункт 13 с описанием отчётов |
| 2.6 | 21.12.2021 | Дробышевский С.В. | Обновлено описание запроса списка транзакций и запроса количества транзакций. |
| 2.5 | 15.07.2021 | Кушнеров Н.Р. | Обновлено описание запроса списка транзакций. |
| 2.4 | 08.04.2021 | Мамченкова М.А. | Добавлено Приложение 1. Коды ошибок выполнения запросов. |
| 2.3 | 26.03.2021 | Сопот И.В. | Добавлен запрос на регистрацию платежа оператором магазина. |
| 2.2 | 16.02.2021 | Сопот И.В. | В список и в детали транзакции добавлено время завершения авторизации. В ответе на запрос проверки прав добавлены идентификатор оператора магазина, данные о счетах (имя и идентификатор). |
| 2.1 | 11.01.2021 | Анискин Р.А. | Обновлено описание запроса завершения авторизации. |
| 2.0 | 11.02.2020 | Мамченкова М.А. | Добавлены примеры запросов. |
| 1.0 | 15.08.2019 | Тарасенко Я.Н. | Начальная версия. |

Содержание

1 Общие сведения 4

1.1 Назначение документа 4

1.2 Сокращения и обозначения 4

1.3 Перечень исходных документов 4

2 Формат запросов и ответов 5

3 Авторизация и работа с профилем 7

3.1 Открытие сессии 7

3.2 Завершение сессии 7

3.3 Открытие сессии со сменой пароля 7

3.4 Продление сессии 8

3.5 Изменение пароля 8

4 Проверка прав 9

5 Получение доступных магазинов 11

6 Рабочий стол 12

6.1 Получение информации по оборотам за период 12

6.2 Получение данных для графиков 13

7 Получение данных о транзакциях 15

7.1 Получение количества транзакций 15

7.2 Получение списка транзакций 17

7.3 Получение информации об одной транзакции 20

7.4 Регистрация платежа оператором магазина 22

8 Завершение авторизации 23

9 Возврат денежных средств 24

10 История верификаций карт 26

10.1 Получение размера списка верификаций карт 26

10.2 Получение списка верификаций карт 26

10.3 Получение информации об одной операции верификации карты 28

11 История регистраций карт 29

11.1 Получение размера списка регистраций карт 29

11.2 Получение списка регистраций карт 29

11.3 Получение информации об одной операции регистрации карты 31

12 Платежные карты 32

12.1 Получение информации об одной карте 32

12.2 Получение размера списка карт 33

12.3 Получение списка карт 33

12.4 Деактивация карты 34

13 Отчёты 35

13.1 Запрос создания отчёта 35

13.2 Получение информации об одном отчёте 36

13.3 Скачивание отчёта 36

13.4 Удаление отчёта 37

13.5 Получение списка отчётов 37

14 Конфигурация магазина 38

Приложение 1. Коды ошибок выполнения запросов 40

# Общие сведения

## Назначение документа

Настоящий документ описывает OpenAPI Payguide™ Acquirer – интерфейс взаимодействия клиентского приложения с Сервисом. OpenAPI Payguide™ Acquirer является легковесным интерфейсом на базе протокола HTTPS.

Документ содержит общее описание интерфейса, а также описание запросов и ответов, касающихся работы с консолью магазина.

Документ описывает версию OpenApi v4 для PGA версии 4.2.87.

## Сокращения и обозначения

|  |  |  |
| --- | --- | --- |
| № | Термин | Содержание |
|  | Payguide™ Acquirer /Система | Payguide™ Acquirer – программный комплекс, обеспечивающий проведение безопасных электронных платежей с применением международных банковских карт в адрес интернет-магазинов. |
|  | PAN | Primary Account Number. Номер электронной пластиковой карты. |
|  | 3-D Secure | XML-протокол, который используется в качестве дополнительного защитного слоя при выполнении мобильных/электронных финансовых транзакций. Протокол разработан Visa для обеспечения безопасности интернет-платежей. Предлагается в виде услуг Verified by Visa и MasterCard SecureCode. |
|  | Клиент | Держатель международной банковской карты, осуществляющий покупки в электронном (интернет) магазине. |
|  | Эмитент | Банк-эмитент, финансовый институт, выпускающий в обращение кредитную или дебетовую карточку. |
|  | Сервис | Программно-аппаратный комплекс Payguide Acquirer |
|  | КИ | Клиентский интерфейс. Это может быть консоль магазина или внешняя система, действующая от имени оператора магазина |

## Перечень исходных документов

|  |  |  |  |
| --- | --- | --- | --- |
| ID | Название документа | Версия, Дата | Автор |
|  |  |  |  |

# Формат запросов и ответов

Взаимодействие осуществляется по протоколу HTTPS c использованием методов GET и POST в кодировке UTF-8.

Для всех запросов используется базовый адрес:

[https://<host>:<port>/api/<protocol\_version>**/**](https://mobile.mastercard.ru/api/v3/)**<portal\_id>/**

где  
<host> и <port> - адрес сервера,

<protocol\_version> - версия протокола. Максимальная из поддерживаемых сейчас версий - v4, минимальная - v3.

<portal\_id> - идентификатор портала, соответствующий веб-приложению, мобильному приложению или серверу сторонней системы.

Пример запроса:

POST /api/v4/01234567890ABCDEF/payment HTTP/1.1

Host: mobile.mastercard.ru

Content-Type: application/x-www-form-urlencoded

Content-Length: 41

param1=value1&param2=value2&param3=value3

В ответ от сервиса приложение получает JSON-структуру с необходимыми данными, например:

HTTP/1.1 200 OK

Content-Type: application/json

Content-Length: 51

Expires: Thu, 01 Dec 1994 16:00:00 GMT

Cache-Control: no-cache

{

    "param1":"value1",

    "param2":"value2"

}

Некоторые запросы являются командами Сервису на выполнение различных действий и не подразумевают передачи в ответе какой-либо информации, кроме статуса выполнения команды.

Примеры запросов-команд: запрос завершения сессии, запрос удаления карты.

При успешном выполнении таких запросов сервис в ответе выставляет HTTP-код 200, а тело сообщения при этом отсутствует. Например:

HTTP/1.1 200 OK

Date: Fri, 07 Feb 2020 13:01:05 GMT

Content-Length: 0

Access-Control-Allow-Origin: \*

Connection: close

В случае ошибки при обработке запроса Сервис выставляет один из неуспешных HTTP-кодов и возвращает информацию об ошибке. Например:

HTTP/1.1 400 Bad Request

Content-Type: application/json

Content-Length: 58

Expires: Thu, 01 Dec 1994 16:00:00 GMT

Cache-Control: no-cache

{

    "error":"INVALID\_PARAMETER",

    "param":"amount"

}

В случае, если приложению не удалось получить ответ от Сервиса, оно должно повторять запрос до получения ответа.

Для указания контекста безопасности используются HTTP-заголовки.

В частности, для указания параметров авторизации используется заголовок X-IV-Authorization.

Если операция выполняется в рамках сессии, то в заголовке указывается её идентификатор, например:

X-IV-Authorization: Session 769B189AB989D1A77153522F5D869049

Если идентификатор сессии указан неверно или отсутствует, Сервис возвращает сообщение об ошибке с HTTP-кодом 400 Bad Request. В теле ответа будет указан код ошибки ACCESS\_DENIED.

# Авторизация и работа с профилем

Все запросы, касающиеся магазинной консоли, должны выполняться только авторизованными пользователями. Для прохождения авторизации оператор должен предъявить логин и пароль. Сервис проверяет правильность пароля и открывает сессию. И далее все запросы отправляются в рамках этой сессии.

Профиль оператора магазина создаётся в административной консоли Сервиса.

Отправлять запросы Сервису может оператор магазина с помощью магазинной консоли. Или это может делать какая-то внешняя система от имени оператора магазина. Формат запросов и правила их обработки при этом одинаковые.

## Открытие сессии

Формат запроса открытия сессии:

POST /api/v4/<portal\_id>/session/start

Пример запроса открытия сессии:

POST /api/v4/F99B664D6D9B2DB0806656F28E3A4DF9/session/start HTTP/1.0

Content-Length: 37

Content-Type: application/x-www-form-urlencoded;charset=UTF-8

identifier=Petrov1&password=654321

**Параметры запроса**

| * **Параметр** |  | * **Описание** |
| --- | --- | --- |
| * identifier | * R | * Идентификатор профиля оператора магазина |
| * password | * R | * Пароль оператора магазина |

В случае успешной проверки пароля Сервис открывает новую сессию и возвращает ее идентификатор, например:

{

    "sessionId":"769B189AB989D1A77153522F5D869049"

}

Для всех запросов, отправляемых в рамках открытой сессии, КИ указывает в HTTP-заголовке «X-IV-Authorization»  полученный идентификатор сессии с префиксом «Session »

X-IV-Authorization: Session 769B189AB989D1A77153522F5D869049

## Завершение сессии

Завершение текущей сессии осуществляется следующим запросом без параметров:

POST /api/v4/<portal\_id>/session/finish

Сервис завершает текущую сессию и возвращает пустой ответ со статусом HTTP 200 OK.  
Сервис возвращает успешный результат выполнения запроса даже тогда, когда сессии с указанным ИД не существует или она просрочена.

Пример запроса завершения текущей сессии:

POST /api/v4/F99B664D6D9B2DB0806656F28E3A4DF9/session/finish HTTP/1.0

Content-Length: 0

X-IV-Authorization: Session 771E111111111122222222223333A273

## Открытие сессии со сменой пароля

Пароль к профилю оператора обычно имеет ограниченный срок жизни. Когда этот срок истекает, открыть сессию со старым паролем уже нельзя. В этом случае единственный вариант для оператора продолжить работу с Сервисом - выполнить запрос открытия сессии с одновременной сменой пароля:

POST /api/v4/<portal\_id>/session/start-with-change-password

Пример запроса открытия сессии со сменой пароля:

POST /api/v4/F99B664D6D9B2DB0806656F28E3A4DF9/session/start-with-change-password HTTP/1.0

Content-Length: 37

Content-Type: application/x-www-form-urlencoded;charset=UTF-8

identifier=Petrov1&password=654321&newPassword=654321

**Параметры запроса**

| * **Параметр** |  | * **Описание** |
| --- | --- | --- |
| * identifier | * R | * Идентификатор профиля оператора магазина |
| * password | * R | * Текущий пароль оператора магазина |
| * newPassword | * R | * Новый пароль оператора магазина |

В случае успешной проверки текущего пароля Сервис проверяет пригодность нового пароля. Если новый пароль удовлетворяет требованиям безопасности, то Сервис открывает новую сессию и возвращает ее идентификатор, например:

{

    "sessionId":"769B189AB989D1A77153522F5D869049"

}

## Продление сессии

Сессия имеет небольшое время жизни. Но при каждом поступлении запроса не сервер сессия, в рамках которой был отправлен этот запрос, продлевается.

Иногда бывают ситуации, когда пользователь выполняет длительные операции в приложении и не отправляет на сервер никаких запросов. Чтобы при этом сессия не завершилась по таймауту, приложение может периодически отправлять на сервер запрос продления сессии:

POST /api/v4/<portal\_id>/session/keep-alive

Пример запроса продления сессии:

POST /api/v4/F99B664D6D9B2DB0806656F28E3A4DF9/session/keep-alive HTTP/1.0

X-IV-Authorization: Session 771E111111111122222222223333A273

При успешном выполнении запроса Сервис возвращает пустой ответ со статусом HTTP 200 OK.

## Изменение пароля

Чтобы изменить пароль оператора, необходимо выполнить следующий запрос:

POST /api/v4/<portal\_id>/profile/password/change

Пример запроса изменения пароля:

POST /api/v4/F99B664D6D9B2DB0806656F28E3A4DF9/profile/password/change HTTP/1.0

Content-Length: 40

X-IV-Authorization: Session 771E111111111122222222223333A273

Content-Type: application/x-www-form-urlencoded

password=123456&newPassword=654321

**Параметры запроса**

| * **Параметр** |  | * **Описание** |
| --- | --- | --- |
| * password | * R | * Текущий пароль оператора магазина |
| * newPassword | * R | * Новый пароль оператора магазина |

Этот запрос выполняется в рамках открытой ранее сессии.  
При успешном выполнении запроса Сервис возвращает пустой ответ со статусом HTTP 200 OK.

# Проверка прав

После того, как была открыта сессия, КИ может отправить запрос проверки прав:

GET /api/v4/<portal\_id>/merchant/statistics/permission

Пример запроса проверки прав:

GET /api/v4/F99B664D6D9B2DB0806656F28E3A4DF9/merchant/statistics/permission HTTP/1.0

X-IV-Authorization: Session 771E111111111122222222223333A273

Если у оператора нет доступа ни к одному магазину выбранного партнёра, то в ответ будет передано сообщение об ошибке с кодом ACCESS\_DENIED. Если доступ хотя бы к одному магазину разрешён, то в ответе будут такие поля:

| * **Поле** |  | * **Описание** |
| --- | --- | --- |
| * identifier | * R | * Идентификатор оператора магазина |
| * refund | * R | * Флаг доступности оператору действий "возврат денежных средств" и «завершение авторизации» (true/false) |
| * paymentTypes | * R | * Массив типов транзакций платёжных схем, настроенных в Сервисе. Учитываются неблокированные и не скрытые схемы * (CARD\_TO\_CARD, CARD\_TO\_ACCOUNT, ACCOUNT\_TO\_CARD, GAMBLING, ACQUIRING, TEST\_PAYMENT) |
| * merchants | * R | * Массив с описанием доступных магазинов |

Описание одного магазина:

| * **Параметр** |  | * **Описание** |
| --- | --- | --- |
| * id | * R | * Идентификатор магазина |
| * name | * R | * Название магазина |
| * logoUrl | * R | * Адрес логотипа |
| * accounts | * O | * Массив с описанием счетов магазина |

Описание одного счёта:

| * **Параметр** |  | * **Описание** |
| --- | --- | --- |
| * id | * R | * Идентификатор счёта |
| * name | * R | * Название счёта |
| * terminalId | * R | * Идентификатор платежного терминала |
| * merchantId | * O | * Идентификатор магазина |
| * mcc | * O | * Код категории магазина |
| * full3dsOnly | * R | * Признак использования полного 3-D Secure |

**Примеры ответов**

**Пример успешного ответа с описанием трёх магазинов**

{

identifier: "79160000006",

    refund: "false",

    paymentTypes: [

        "ACQUIRING"

    ],

    merchants: [

        {id:"TESTMERCHANT", name:"Test Merchant", logoUrl: "[https://testmerchant.ru/logo.png"](https://testmerchant.ru/logo.png), accounts: [] },

        {id:"TESTMERCHANT-2", name:"Test Merchant 2", logoUrl: "[https://testmerchant2.ru/logo.png"](https://testmerchant2.ru/logo.png), accounts: [{id: 222233334444, name: "Name", "terminalId":"55667789", "merchantId":"Merchant123", "mcc":"6012", "full3dsOnly":true}]},

        {id:"TESTMERCHANT-3", name:"Test Merchant 3", logoUrl: "[https://testmerchant3.ru/logo.png"](https://testmerchant3.ru/logo.png), accounts: [{id: 333344445555, name: "Name2", "terminalId":"1122334455", "full3dsOnly":false}, {id: 444455556666, name: "Name3", "terminalId":"6677889900", "full3dsOnly":true}]}

    ]

}

**Пример неуспешного ответа**

//Доступ к данным партнера запрещен --- дальнейшие запросы невозможны, сессию можно закрывать

HTTP/1.1 400 Bad Request

{"error":"ACCESS\_DENIED","subject":"USER","reason":"OPERATION"}

# Получение доступных магазинов

Для получения расширенных данных о доступных оператору магазинах следует выполнить запрос:

GET /api/v4/<portal\_id>/merchant

В ответном сообщении будет содержаться список с информацией по доступным магазинам. Описание одного магазина содержит такие поля:

| * **Поле** |  | * **Описание** |
| --- | --- | --- |
| * id | * R | * Идентификатор магазина |
| * name | * R | * Название магазина |
| * logoUrl | * R | * URL логотипа магазина |
| * blocked | * R | * Статус блокировки магазина (true/false) |
| * refundEnabled | * R | * Доступность операции возврата средств и завершения авторизации (true/false) |
| * emailReceiptEnabled | * R | * Доступность отправки чека по e-mail (true/false) |
| * smsReceiptEnabled | * R | * Доступность отправки чека по SMS (true/false) |
| * acquiringTypes | * R | * Массив, описывающий доступные виды эквайринга: * INTERNET - Интернет-эквайринг, * MPOS - mPOS-эквайринг * MOBILE - мобильный эквайринг * CASH - платежи наличными |

**Пример ответа с описанием одного магазина**

HTTP/1.1 200 OK

Content-Type: application/json

[

   {

       "id": "TESTMERCHANT",

       "name": "Test Merchant",

       "logoUrl": "[https://testmerchant.ru/logo.png"](https://testmerchant.ru/logo.png),

       "blocked": false,

       "refundEnabled": true,

       "emailReceiptEnabled": true,

       "smsReceiptEnabled": true,

       "acquiringTypes": ["INTERNET", "MPOS", "MOBILE", "CASH"]

   }

# Рабочий стол

## Получение информации по оборотам за период

Запросить сводную информацию по оборотам за период можно с помощью запроса:

GET /api/v4/<portal\_id>/merchant/statistics/summary

Пример запроса сводной информации по оборотам за период:

GET /api/v4/F99B664D6D9B2DB0806656F28E3A4DF9/merchant/statistics/summary?from=1581282000000&to=1581886799999&type=ACQUIRING HTTP/1.0

X-IV-Authorization: Session 771E111111111122222222223333A273

**Параметры запроса**

| **Параметр** |  | **Описание** |
| --- | --- | --- |
| type | R | Тип транзакции (CARD\_TO\_CARD, CARD\_TO\_CASH, PAYMENT, ACQUIRING) |
| from | O | Начало диапазона (Java Timestamp — long) |
| to | O | Конец диапазона (Java Timestamp — long) |
| merchantId | O | Идентификатор магазина |

**Поля ответа**

| **Поле** |  | **Описание** |
| --- | --- | --- |
| sum | R | Общая сумма операции по успешным транзакциям |
| avg | R | Средняя сумма операции по успешным транзакциям |
| commission | R | Общая сумма комиссии по успешным транзакциям |
| totalCount | R | Общее количество завершённых транзакций |
| successfulCount | R | Количество успешных транзакций |

Транзакции при составлении сводной информации группируются по валюте. Т.е. транзакции рублёвых платежей считаются отдельно от транзакций долларовых платежей.

В качестве имен групп используется буквенный код валюты.  
Возможна ситуация, при которой транзакция завершилась с ошибкой до того, как была определена сумма и валюта операции. В базе данных такая транзакция будет храниться с пустым полем "Валюта операции". Данные о таких транзакциях тоже включаются в отчёт. При этом используется группа с названием "UNKNOWN", как указание на то, что это транзакции с неизвестным кодом валюты.

Если в фильтре по ИД магазина был указан несуществующий ИД, то Сервис вернёт код ошибки MERCHANT\_NOT\_FOUND. А если там был указан ИД существующего магазина, но у оператора нет прав на работу с этим магазином, то Сервис вернёт код ошибки ACCESS\_DENIED.

**Примеры ответов**

**Пример успешного ответа на запрос информации по оборотам**

{

 "RUB":

 {

  "sum": 2565000,

  "avg": 2565.0,

  "commission": 0,

  "totalCount": 60,

  "successfulCount": 54

 },

 "USD":

 {

  "sum": 2565,

  "avg": 256.0,

  "commission": 0,

  "totalCount": 6,

  "successfulCount": 5

 },

 "UNKNOWN":

 {

  "sum": 0,

  "avg": 0,

  "commission": 0,

  "totalCount": 1,

  "successfulCount": 0

 }

}

**Пример ответа, когда ни одна транзакция не попала в выборку**

{}

## Получение данных для графиков

Запрос данных для графика суммы операций:

GET /api/v4/<portal\_id>/merchant/statistics/amount

Запрос данных для графика комиссии:

GET /api/v4/<portal\_id>/merchant/statistics/commission

Запрос данных для графика числа транзакций:

GET /api/v4/<portal\_id>/merchant/statistics/trx

Запрос данных для трёх графиков сразу:

GET /api/v4/<portal\_id>/merchant/statistics/full

**Параметры запросов**

| **Параметр** |  | **Описание** |
| --- | --- | --- |
| type | O | Тип транзакции (CARD\_TO\_CARD, CARD\_TO\_CASH, PAYMENT, ACQUIRING) |
| scale | R | Величина шага по временной шкале графика (MINUTE, HOUR, DAY, WEEK, MONTH) |
| status | O | Статус транзакций (DECLINED, ERROR, PROCESSING, SUCCESS, INTERIM\_SUCCESS, FINISHED) Параметр необязательный. Если не указан, то учитываются транзакции с любым статусом |
| from | O | Начало диапазона (Java Timestamp — long) |
| to | O | Конец диапазона (Java Timestamp — long) |
| merchantId | O | Идентификатор магазина |

**Формат ответа**

Данные о транзакциях для графиков, также как и сводная информация по оборотам, группируются по валюте. Транзакции с неизвестным кодом валюты попадают в группу с названием "UNKNOWN".

В ответах на запросы /amout, /commission и /trx содержатся данные для одного графика.  
Группа данных по одной валюте содержит массив "координат" точек для графика.   
Первая координата - горизонтальная, это "время", указывается в виде Java Timestamp.   
Вторая координата - вертикальная, это может быть "сумма операций" или "комиссия по операциям", указывается в минорных единицах. А может быть "количество транзакций", указывается в штуках.

В ответе на запрос /full содержатся данные сразу для трёх графиков.  
Группа данных по одной валюте содержит массив "координат" точек для трёх графиков.

Первое число - координата по горизонтали, "время", указывается в виде Java Timestamp.   
Второе число - координата по вертикали, "количество транзакций", указывается в штуках.

Третье число - координата по вертикали, "сумма операций" указывается в минорных единицах.

Четвёртое число - координата по вертикали, "комиссия по операциям", указывается в минорных единицах.

Пятое число - координата по вертикали, “общее количество транзакций”, без учета статуса, указывается в штуках.

Шестое число - координата по вертикали, “общая сумма операций”, без учета статуса указывается в минорных единицах.

Точки для трех графиков, если в параметрах запроса отсутствует параметр status:

1. первое число - координата по горизонтали, "время", указывается в виде Java Timestamp;
2. второе число - координата по вертикали, "общее количество транзакций", указывается в штуках;
3. третье число - координата по вертикали, "общая сумма операций" указывается в минорных единицах;
4. четвёртое число - координата по вертикали, "общая комиссия по операциям", указывается в минорных единицах.

Если в фильтре по ИД магазина был указан несуществующий ИД, то Сервис вернёт код ошибки MERCHANT\_NOT\_FOUND. А если там был указан ИД существующего магазина, но у оператора нет прав на работу с этим магазином, то Сервис вернёт код ошибки ACCESS\_DENIED.

**Примеры ответов**

**Пример ответа с данными для одного графика**

{

 "RUB":

 [

  [1407452400000, 13],

  [1407538800000, 21],

  [1407625200000, 65],

  [1407711600000, 48],

  [1407798000000, 25],

  [1407884400000, 15],

  [1407970800000, 88]

 ],

 "USD":

 [

  [1407538800000, 2],

  [1407625200000, 6]

 ]

}

**Пример ответа с данными для трёх графиков**

{

"RUB":

[

[1407452400000, 13, 4500, 45, 20, 9600],

[1407538800000, 21, 200, 2, 25, 5000],

[1407625200000, 65, 12500, 125, 78, 35600],

[1407711600000, 48, 344, 4, 55, 1500],

[1407798000000, 25, 10000, 100, 36, 23000],

[1407884400000, 15, 5000, 300, 29, 9700],

[1407970800000, 88, 12000, 120, 97, 25500]

],

"USD":

[

[1407538800000, 2, 200, 2, 5, 1500],

[1407625200000, 6, 200, 2, 11, 2300]

]

}

**Пример ответа с данными для трёх графиков, если отсутствует параметра status**

{

"RUB":

[

[1407452400000, 13, 4500, 45],

[1407538800000, 21, 200, 2],

],

"USD":

[

[1407538800000, 2, 200, 2],

[1407625200000, 6, 200, 2]

]

}

**Пример ответа, когда ни одна транзакция не попала в выборку**

{}

# Получение данных о транзакциях

## Получение количества транзакций

Запрос получения количества транзакций:

GET /api/v4/<portal\_id>/merchant/history/trx-count

Пример запроса получения количества транзакций:

GET /api/v4/EC301C372085C1251C4E4EEBBD6E8AAB/merchant/history/trx-count?from=1580677200000&status=FINISHED&to=1581281999999&type=acquiring HTTP/1.0

X-IV-Authorization: Session 771E111111111122222222223333A273

**Параметры запроса**

| **Название** |  | **Описание** |
| --- | --- | --- |
| type | O | Тип транзакции (CARD\_TO\_CARD, CARD\_TO\_ACCOUNT, ACCOUNT\_TO\_CARD, GAMBLING, ACQUIRING, TEST\_PAYMENT).  Можно указать несколько значений через запятую |
| status | O | Статус транзакций.  Возможные значения:   * DECLINED – операция отклонена клиентом * ERROR – неуспешная операция * PROCESSING – операция не завершена * SUCCESS – успешная операция * INTERIM\_SUCCESS – неоконченная операция с промежуточным успешным результатом * FINISHED – операция завершена (статус SUCCESS, INTERIM\_SUCCESS, ERROR или DECLINED)   Если параметр не указан, фильтрация по статусу не выполняется  Можно указать несколько значений через запятую |
| from | O | Начало диапазона (Java Timestamp — long) |
| to | O | Конец диапазона (Java Timestamp — long) |
| params.\* | O | Имя и значение параметра платежа. Можно указать несколько параметров.  Пример:  params.orderId=123456 params.checkout=true |
| rrn | O | RRN (строка) |
| srcType | O | Тип источника платежа Возможные значения: CARD, CARD\_ID, APPLE\_PAY. GOOGLE\_PAY, GAZPROM\_PAY, SBP, MIR\_PAY  Можно указать несколько значений через запятую |
| src | O | Источник денежных средств для платежа/перевода. При платеже с карты тут может быть указан маскированный PAN карты, (маска NNNNNNxxxxxxNNNN), либо последние 4 цифры номера карты (NNNN) |
| dst | O | Приёмник денежных средств для платежа/перевода. При переводе с карты на карту тут нужно указать маскированный PAN карты, маска NNNNNNxxxxxxNNNN, либо последние 4 цифры номера карты (NNNN) |
| currency | O | Трехсимвольный код валюты транзакции |
| token | O | Полный токен транзакции или часть токена, начинающаяся с его первого символа |
| merchantId | O | Идентификатор магазина |
| merchantTrx | O | Номер транзакции магазина |
| accountId | O | Идентификатор магазина |
| srcId | O | ИД карты-источника, для которой нужно искать платежи |
| recurrent | O | Фильтрация платежей по признаку рекуррентности.  При указании true Сервис вернёт только рекуррентные платежи, при указании false - только нерекуррентные |
| srcAddedToProfile | O | Фильтрация платежей по флагу "Добавление карты в профиль". При указании true сервер вернёт только платежи, по итогам которых карта была сохранена в профиле магазина, при указании false - только платежи, в которых сохранения карты не было |

**Поля ответа**

В ответе будет только поле count, содержащее количество транзакций, подошедших под условия фильтрации.

Если в фильтре по ИД магазина был указан несуществующий ИД, то Сервис вернёт код ошибки MERCHANT\_NOT\_FOUND.

А если там был указан ИД существующего магазина, но у оператора нет прав на работу с этим магазином, то Сервис вернёт код ошибки ACCESS\_DENIED.

**Ограничения временного интервала**

Для уменьшения нагрузки на систему предусмотрена возможность ограничения «тяжелых» запросов по поиску данных о транзакциях. Для этого в конфигурации протокола OpenApi предусмотрен флаг "Ограничивать интервал запросов истории". При включении этой настройки начинают действовать следующие ограничения на интервал времени, за который можно выполнять поиск данных:

1. Если в запросе есть хотя бы один из индексных фильтров (token, rrn, merchantTrx, то ограничения на временной интервал времени не накладываются, фильтр по времени может вообще быть не заполнен. Значения остальных фильтров при этом не анализируются.
2. Если в запросе нет ни одного из индексных фильтров, то фильтр по времени должен быть заполнен.
3. Если в запросе нет ни одного из индексных фильтров, и есть только фильтры по merchantId или type, то временной интервал долен быть не более 93 дней (~3 месяца).
4. Если в запросе не заполнен ни один индексный фильтр, но заполнены какие-то дополнительные фильтры, то период поиска не должен превышать 31 день (~1 месяц).

Ограничения действуют на запросы «Получение количества транзакций» и «Получение списка транзакций»

## Получение списка транзакций

Запрос получения списка транзакций:

GET /api/v4/<portal\_id>/merchant/history/trx

Пример запроса получения списка транзакций:

GET /api/v4/F99B664D6D9B2DB0806656F28E3A4DF9/merchant/history/trx?from=1581282000000&limit=10&offset=0&orderDir=DESC&orderName=merchantTrx&status=FINISHED&to=1581886799999&type=acquiring HTTP/1.0

X-IV-Authorization: Session 771E111111111122222222223333A273

**Параметры запроса**

Такие же, как и у запроса количества транзакций. Добавляются параметры для сортировки и постраничного просмотра.

| **Параметр** |  | **Описание** |
| --- | --- | --- |
| offset | O | Смещение (для постраничного отображения). По умолчанию 0 |
| limit | O | Количество элементов (для постраничного отображения). По умолчанию 10 |
| orderDir | O | Направление сортировки (DESC/ASC). По умолчанию DESC Сортировка выполняется по дате старта транзакции |
| amount | O | Фильтр по сумме транзакции |

**Поля ответа**

Список транзакций содержит сокращённое описание каждой из операций:

|  |  |  |
| --- | --- | --- |
| **Название** |  | **Описание** |
| token | R | Идентификатор транзакции |
| startedAt | R | Дата старта операции (Java Timestamp — long) |
| finishedAt | C | Дата завершения операции (Java Timestamp — long) |
| completedAt | O | Дата завершения авторизации (Java Timestamp — long) |
| correctedAt | O | Дата корректировки операции (Java Timestamp — long) |
| type | R | Тип транзакции (CARD\_TO\_CARD, CARD\_TO\_CASH, PAYMENT, ACQUIRING) |
| rrn | C | RRN (строка) |
| approvalCode | C | Код авторизации |
| srcType | O | Тип источника платежа Возможные значения: CARD, CARD\_ID, APPLE\_PAY, MIR\_PAY |
| src | C | Источник денежных средств. Для платежа с карты тут будет маскированный PAN отправителя, маска NNNNNNxxxxxxNNNN |
| srcAddedToProfile | O | Флаг добавления карты в профиль магазина. Если в ходе платежа карта была добавленато у этого поля будет значение true. Если карта не была добавлена, то поле отсутствует |
| dst | O | Получатель денежных средств. Для перевода с карты на карту тут будет маскированный PAN карты-получателя, маска NNNNNNxxxxxxNNNN |
| amount | O | Сумма транзакции |
| commission | O | Комиссия транзакции |
| fullAmount | O | Полная сумма транзакции |
| refundAmount | O | Общая сумма выполненных возвратов |
| currency | O | Трехсимвольный код валюты транзакции (RUB) |
| merchantId | R | Идентификатор магазина |
| merchantTrx | R | Идентификатор транзакции, полученный на первой фазе со стороны ТСП |
| status | R | Статус транзакций (DECLINED, ERROR, PROCESSING, SUCCESS, INTERIM\_SUCCESS) |
| extendedCode | O | Расширенный код завершения транзакции (буквенный) |
| responseCode | O | Код ответа процессинга |
| responseCodeDetails | O | Детализация кода ответа процессинга |
| isFullyAuthenticated | C | Была ли успешная 3-D Secure аутентификация Возможные значения: true, false |
| orderStatus | O | Статус заказа.  Возможные значения: UNKNOWN - статус заказа неизвестен (или нет дополнительно статуса у данного заказа) REGISTERED\_AT\_MERCHANT - заказ зарегистрирован у продавца (магазине) RECEIVED - заказ выдан получателю REGISTRATION\_ATTEMPTS\_EXHAUSTED - исчерпаны попытки регистрации заказа у продавца (магазина) |
| orderStatusChangedAt | O | Дата и время последнего изменения статуса заказа в формате Unix time |
| refundable | R | Возможно ли выполнение отмены авторизации или возврата средств по данной транзакции (true, false) |
| recurrent | O | Признак рекуррентности платежа.  Если платёж был рекуррентным, то у этого поля будет значение true.  Если платёж не был рекуррентным, то поле отсутствует |
| hasSignature | O | Признак наличия графического изображения подписи (true, false). По умолчанию false |

Операции в списке отсортированы по дате старта.

**Пример ответа**

**Пример ответа с описанием двух транзакций**

[

 {

  "token" : "234234LK234",

  "startedAt": "1415266887057",

  "finishedAt": "1415266887057",

  "completedAt": "1415266897057",

  "type": "CARD\_TO\_CARD",

  "rrn": 123456789001,

  "approvalCode": "123456",

  "srcType": "card",

  "src": "5454xxxxxxxx5454",

  "dst": "4111xxxxxxxx1111",

  "amount": 49500,

  "commission": 743,

  "fullAmount": 50243,

  "refundAmount": 52500,

  "currency": "RUB",

  "merchantId": "TestMerchant",

  "merchantTrx": "TX-582346237845",

  "status": "SUCCESS",

  "isFullyAuthenticated": true,

"extendedCode": "OK",

"responseCode": "000",

"responseCodeDetails": "Successful",

  "refundable": true

 },

 {

  "token" : "233NB2323",

  "startedAt": 1415266887057,

  "finishedAt": "1415266887057",

  "type": "CARD\_TO\_CARD",

  "rrn": "123456789002",

  "approvalCode": "123457",

  "srcType": "card",

  "src": "5454xxxxxxxx5454",

  "dst": "4111xxxxxxxx1111",

  "amount": 132500,

  "commission": 1888,

  "fullAmount": 134488,

  "refundAmount": 0,

  "currency": "RUB",

  "merchantId": "TestMerchant",

  "merchantTrx": "TX-582346237843",

  "status": "SUCCESS",  
  "isFullyAuthenticated": true,

  "extendedCode": "OK",

"responseCode": "000",

"responseCodeDetails": "Successful",

  "refundable": true

 }

]

## Получение информации об одной транзакции

Запрос получения информации об одной транзакции:

GET /api/v4/<portal\_id>/merchant/history/trx/<token>

Параметров у запроса нет.

Пример запроса получения информации об одной транзакции:

GET /api/v4/F99B664D6D9B2DB0806656F28E3A4DF9/merchant/history/trx/9XBVR1YVU3G358C4 HTTP/1.0

X-IV-Authorization: Session 771E111111111122222222223333A273

**Поля ответа**

В ответе будет полное описание одной операции. По сравнению с сокращённым описанием операции дополнительно будут такие данные:

| **Поле** |  | **Описание** |
| --- | --- | --- |
| srcId | O | ИД карты-источника. Поле присутствует, если платёж выполнялся с зарегистрированной карты. Платёж, в ходе которого была зарегистрирована карта, считается проведённым с зарегистрированной карты. |
| cardholder | O | Имя держателя карты |
| params.\* | O | Набор параметров заказа (o.\*) |
| description | R | Описание заказа |
| refunds.\* | O | Список возвратов |
| location.\* | O | GPS-координаты в формате десятичной дроби с точностью до 6 знаков |
| location.latitude | R | GPS-координаты, широта |
| location.longitude | R | GPS-координаты, долгота |

Описание одного возврата:

| **Параметр** |  | **Описание** |
| --- | --- | --- |
| createdAt | R | Дата регистрации возврата (Java Timestamp — long) |
| action | O | Выполненное действие.  Возможные значения:   * PAYMENT - был выполнен "обратный" платеж * REVERSAL - была выполнена отмена платежа * REFUND - был выполнен возврат средств * UNKNOWN - тип выполненного действия неизвестен (используется только при возникновении ошибки в ходе выполнения запроса) |
| initiator | O | Инициатор возврата средств. Возможные значения:   * CUSTOMER - клиент * PARTNER - партнёр (через партнёрскую часть OpenApi или в партнёрской консоли), * MERCHANT - магазин (через магазинную часть OpenApi или в консоли магазина), * OPERATOR - оператор админ. консоли |
| initiatorId | R | Идентификатор оператора партнёрской консоли, выполнившего возврат. Или идентификатор клиента, если возврат был инициирован клиентом. Или идентификатор оператора админ. консоли. |
| comment | O | Комментарий к возврату |
| status | R | Статус транзакций возврата (ERROR, PROCESSING, SUCCESS) |

**Пример ответа**

**Пример ответа с описанием одной транзакции**

{

 "token" : "34234NNM234234",

 "startedAt": 1415266887057,

 "finishedAt": "1415266887057",

 "type": "CARD\_TO\_CARD",

 "rrn": "123456789001",

 "approvalCode": "123456",

 "srcType": "card",

 "src": "5454xxxxxxxx5454",

 "dst": "4111xxxxxxxx1111",

 "amount": 49500,

 "commission": 743,

 "fullAmount": 50243,

 "refundAmount": 0,

 "currency": "RUB",

 "merchantId": "TestMerchant",

 "merchantTrx": "TX-582346237843",

 "status": "SUCCESS",

 "isFullyAuthenticated": true,

 "refundable": true,

 "cardholder": "IVAN IVANOV",

 "params": {

    "order": "12370"

 },

 "description": "Order 12370 - The Martian",

 "refunds":[

  {

   "createdAt: "1415266887057",

   "amount: "2500",

   "msisdn: "79160000001",

   "comment: "Partial refund",

   "status: "PROCESSING"

  }

 ],

 "hasSignature": true,

 "location": { latitude: 12.345678, longitude: -123.456789 }

}

## Регистрация платежа оператором магазина

Запрос регистрации платежа:

POST /api/v4/\*/merchant/history/trx/<token>/register

Параметров у запроса нет.

Пример запроса регистрации платежа:

POST /api/v4/\*/merchant/history/trx/9XBVR1YVU3G358C4/register

HTTP/1.0

X-IV-Authorization: Session 771E111111111122222222223333A273

**Поля ответа**

Ответ в виде JSON объекта с одним полем orderStatus:

| **Поле** |  | **Описание** |
| --- | --- | --- |
| orderStatus | R | * Статус заказа. Возможные значения: * UNKNOWN - статус заказа неизвестен (или нет дополнительно статуса у данного заказа) * REGISTERED\_AT\_MERCHANT - операция зарегистрирована у продавца (магазина) |

**Пример ответа**

Пример успешного ответа

{

orderStatus: "REGISTERED\_AT\_MERCHANT"

}

****Пример неуспешного ответа****

{

    orderStatus: "UNKNOWN"

}

****Пример ошибки "Транзакция не найдена"****

{

    error: "TRANSACTION\_NOT\_FOUND"

}

****Пример ошибки "Транзакция  уже зарегистрирована"****

{

    error: "OPERATION\_NOT\_AVAILABLE",

reason: "Transaction is already registered at merchant"

}

# Завершение авторизации

Завершение авторизации можно выполнить только для транзакции, по которой до этого была успешно выполнена предавторизация. Признаком таких транзакций является статус INTERIM\_SUCCESS и расширенный код PREAUTHORIZATION\_OK.

Запрос на завершение авторизации:

POST /api/v4/<portal\_id>/merchant/history/trx/<token>/complete

Пример запроса на завершение авторизации:

POST /api/v4/EC301C372085C1251C4E4EEBBD6E8AAB/merchant/history/trx/IP49367DOKOUM2FU/complete HTTP/1.0

X-IV-Authorization: Session 771E111111111122222222223333A273

Content-Length: 25

Content-Type: application/x-www-form-urlencoded;charset=UTF-8

amount=12300&currency=GEL

**Параметры запроса**

| **Название** |  | **Описание** |
| --- | --- | --- |
| amount | R | Сумма. Не должна превышать сумму предавторизации |
| currency | R | Код валюты (трехбуквенный), должен соответствовать оригинальному значению |
| description | O | Новое описание заказа |

Если запрос не был допущен к исполнению, то Сервис вернёт сообщение об ошибке. Ошибки могут быть такие:

* ACCESS\_DENIED - если оператору недоступна операция завершения авторизации или двухфазная авторизация запрещена для магазина,
* OPERATION\_NOT\_AVAILABLE – если выполнить завершение авторизации невозможно из-за неподходящего состояния транзакции,
* INVALID\_PARAMETER - если один или несколько параметров некорректны,
* AMOUNT\_TOO\_BIG - сумма завершения авторизации превышает сумму предавторизации,
* CURRENCY\_MISMATCH - валюта завершения авторизации не совпадает с валютой предавторизации

Если при выполнении запроса возникла неожиданная ошибка, то Сервис вернёт сообщение об ошибке с соответствующим кодом:

* INTERNAL\_ERROR - если процессинг вернул неопределённый результат завершения авторизации или произошла ошибка на сервере

Если запрос был допущен к исполнению и обработан без ошибок, то Сервис вернёт новое описание состояния транзакции. В этом описании, по сравнению с предыдущим описанием, могут измениться поля:

1. amount - если сумма завершения авторизации не совпадает с суммой предавторизации;
2. description – если в запросе было указано новое описание заказа;
3. status
4. extendedCode
5. responseCode
6. rrn - если протокол процессинга предполагает разные rrn для первой и второй фазы;
7. approvalCode

**Примеры ответов**

**Пример успешного ответа на запрос завершения авторизации**

HTTP/1.1 200 OK

Content-Type: application/json

{

    token: "1234567890ABCDEF",

    startedAt: "1415266887057",

    finishedAt: "1415266887057",

completedAt: "1415266887057",

    type: "ACQUIRING",

    rrn: "123123123",

    approvalCode: "123434",

    srcType: "card",

    src: ""5454xxxxxxxx5454"",

    amount: "10000",

    fullAmount: "10000",

    currency: "RUB",

    merchantId: "MERCH1",

    merchantTrx: "trx-232323",

    status: "SUCCESS",

    extendedCode: "OK",

    responseCode: "00",

    isFullyAuthenticated: true,

    orderStatus: "REGISTERED\_AT\_MERCHANT",

    orderStatusChangedAt: "1415266887057",

    refundable: true,

    params: {

        order\_id: "121212"

    },

    description: "Какой-то платёж"

}

**Пример неуспешного ответа на запрос завершения авторизаци**

HTTP/1.1 400 Bad Request

Content-Type: application/json

{"error":"AMOUNT\_TOO\_BIG"}

# Возврат денежных средств

Запрос на выполнение возврата может быть выполнен только для операций с признаком refundable, равным true.

POST /api/v4/<portal\_id>/merchant/history/trx/<token>/refund

Пример запроса на выполнение возврата денежных средств:

POST /api/v4/F99B664D6D9B2DB0806656F28E3A4DF9/merchant/history/trx/XFYCVKFXFL0N153J/refund HTTP/1.0

X-IV-Authorization: 366320873CE433D3C19788E8FA5ECD04

Content-Length: 40

Content-Type: application/x-www-form-urlencoded

amount=7500&currency=USD&comment=comment

**Параметры запроса refund**

| **Название** |  | **Описание** |
| --- | --- | --- |
| amount | R | Сумма возврата. Не должна превышать оставшуюся сумму (сумма оригинальной транзакции за вычетом выполненных ранее частичных возвратов) |
| currency | R | Код валюты (трехбуквенный), должен соответствовать оригинальному значению |
| comment | O | Текстовый комментарий |

**Поля ответа**

| **Поле** |  | **Описание** |
| --- | --- | --- |
| status | R | Статус обработки запроса:   * SUCCESS - запрос выполнен успешно * FAILED - запрос выполнен неуспешно * UNKNOWN - статус выполнения запроса неопределённый * PROCESSING - запрос в обработке. Используется в некоторых платёжных схемах, где возврат средств по платежу занимает длительное время |
| action | R | Действие, выполненное в ходе обработки запроса   * REVERSAL - по транзакции была выполнена отмена, т.е. Reversal * REFUND - по транзакции был выполнен возврат средств, т.е. Refund |
| rrn | O | Код RRN выполненной операции Reversal или Refund, если они были выполнены и если протокол банка подразумевает наличие этого поля в результате выполнения операции |

**Примеры ответов на запрос возврата средств**

// Возврат выполнен успешно

HTTP/1.1 200 OK

Content-Type: application/json

{"status":"SUCCESS","action":"REFUND","rrn":"1231234123"}

// Возврат не выполнен

HTTP/1.1 200 OK

Content-Type: application/json

{"status":"FAILED","action":"REFUND","rrn":"1231234123"}

# История верификаций карт

## Получение размера списка верификаций карт

Запрос получения размера списка верификаций карт:

GET /api/v4/<portal\_id>/merchant/history/card-verification-count

Пример запроса получения размера списка верификаций карт:

GET /api/v4/F99B664D6D9B2DB0806656F28E3A4DF9/merchant/history/card-verification-count?from=1581282000000&limit=10&offset=0&orderDir=DESC&orderName=merchantTrx&status=FINISHED&to=1581886799999 HTTP/1.0

X-IV-Authorization: Session 771E111111111122222222223333A273

**Параметры запроса**

| * **Название** |  | * **Описание** |
| --- | --- | --- |
| * status | * O | * Статус операции. Возможные значения:   DECLINED - операция отклонена клиентом  ERROR - неуспешная операция  PROCESSING - операция не завершена  SUCCESS - успешная операция  FINISHED - операция завершена (статус SUCCESS, ERROR или DECLINED)   * Если параметр не указан, фильтрация по статусу не выполняется |
| * from | * O | * Начало диапазона (Java Timestamp — long, включительно) |
| * to | * O | * Конец диапазона (Java Timestamp — long, не включительно) |
| * pan | * O | * PAN карты. Может быть указан маскированный PAN карты, (маска NNNNNNxxxxxxNNNN), либо последние 4 цифры номера карты (NNNN) |
| * token | * O | * Полный токен операции или часть токена, начинающаяся с его первого символа |
| * merchantId | * O | * Идентификатор магазина |
| * merchantTrx | * O | * Номер транзакции магазина |
| * cardId | * O | * Идентификатор карты в системе |

**Поля ответа**

В ответе будет только поле count, содержащее количество операций, подошедших под условия фильтрации.

Если в фильтре по идентификатору магазина был указан несуществующий ИД, то Сервис вернёт код ошибки MERCHANT\_NOT\_FOUND.

А если там был указан идентификатору существующего магазина, но у оператора нет прав на работу с этим магазином, то Сервис вернёт код ошибки ACCESS\_DENIED.

## Получение списка верификаций карт

Запрос получения списка верификаций карт:

GET /api/v4/<portal\_id>/merchant/history/card-verification

Пример запроса получения списка верификаций карт:

GET /api/v4/F99B664D6D9B2DB0806656F28E3A4DF9/merchant/history/card-verification?from=1581282000000&limit=10&offset=0&orderDir=DESC&orderName=merchantTrx&status=FINISHED&to=1581886799999 HTTP/1.0

X-IV-Authorization: Session 771E111111111122222222223333A273

**Параметры запроса**

Такие же, как и у запроса размера списка верификаций карт. Добавляются параметры для сортировки и постраничного просмотра

| * Параметр |  | * Описание |
| --- | --- | --- |
| * offset | * O | * Смещение (для постраничного отображения). По умолчанию 0 |
| * limit | * O | * Количество элементов (для постраничного отображения). По умолчанию 10 |
| * orderDir | * O | * Направление сортировки (DESC/ASC). По умолчанию DESC Сортировка выполняется по дате начала операции |

**Поля ответа**

|  |  |  |
| --- | --- | --- |
| * **Название** |  | * **Описание** |
| * token | * R | * Идентификатор операции верификации карты |
| * startedAt | * R | * Дата старта операции (Java Timestamp — long) |
| * finishedAt | C | * Дата завершения операции (Java Timestamp — long) |
| * pan | C | * Маскированный PAN карты, маска NNNNNNxxxxxxNNNN |
| * merchantId | * R | * Идентификатор магазина |
| * merchantTrx | * R | * Идентификатор транзакции в магазине |
| * status | * R | * Статус транзакций (DECLINED, ERROR, PROCESSING, SUCCESS) |
| * extendedCode | * O | * Расширенный код завершения транзакции (буквенный) |
| * responseCode | * O | * Код ответа процессинга |
| * isFullyAuthenticated | C | * Была ли успешная 3-D Secure аутентификация. Возможные значения: true, false. Заполняется только если была попытка. |
| * orderStatus | * O | * Статус заказа. Возможные значения: * UNKNOWN - статус заказа неизвестен (или нет дополнительно статуса у данного заказа) * REGISTERED\_AT\_MERCHANT - операция зарегистрирована у продавца (магазина) * REGISTRATION\_ATTEMPTS\_EXHAUSTED - исчерпаны попытки регистрации операции у продавца (магазина) |
| * amountReverted | C | * Присутствует со значением "true", только если все списанные в процессе верификации средства были успешно возвращены на карту. |

Операции в списке отсортированы по дате старта.

**Пример ответа**

[

 {

  "token" : "234234LK234",

  "startedAt": "1415266887057",

  "finishedAt": "1415266887057",

  "pan": "5454xxxxxxxx5454",

  "merchantId": "TestMerchant",

  "merchantTrx": "TX-582346237845",

  "status": "SUCCESS",

  "extendedCode": "OK",

  "responseCode": "000",

  "isFullyAuthenticated": true,

  "orderStatus": "REGISTERED\_AT\_MERCHANT",

  "amountReverted": true

 }

]

## Получение информации об одной операции верификации карты

Запрос получения информации об одной операции верификации карты:

GET /api/v4/<portal\_id>/merchant/history/card-verification/<token>

Параметров у запроса нет.

Пример запроса получения информации об одной операции верификации карты:

GET /api/v4/EC301C372085C1251C4E4EEBBD6E8AAB/merchant/history/card-verification/1999FNFTKJC03 HTTP/1.0

X-IV-Authorization: Session 771E111111111122222222223333A273

**Поля ответа**

В ответе будет полное описание одной операции. По сравнению с сокращённым описанием операции, как в списке выше, дополнительно будут такие данные:

| * **Название** |  | * **Описание** |
| --- | --- | --- |
| * cardId | * C | * Идентификатор карты, для которой выполнялась верификация |

**Пример ответа**

{

 "token" : "234234LK234",

 "startedAt": "1415266887057",

 "finishedAt": "1415266887057",

 "pan": "5454xxxxxxxx5454",

 "merchantId": "TestMerchant",

 "merchantTrx": "TX-582346237845",

 "status": "SUCCESS",

 "extendedCode": "OK",

 "responseCode": "000",

 "isFullyAuthenticated": true,

 "orderStatus": "REGISTERED\_AT\_MERCHANT",

 "amountReverted": true,

 "cardId": "34234NNM234234"

}

# История регистраций карт

## Получение размера списка регистраций карт

Запрос получения размера списка регистраций карт:

GET /api/v4/<portal\_id>/merchant/history/card-registration-count

Пример запроса получения размера списка регистраций карт:

GET /api/v4/F99B664D6D9B2DB0806656F28E3A4DF9/merchant/history/card-registration-count?from=1581282000000&limit=10&offset=0&orderDir=DESC&orderName=merchantTrx&status=FINISHED&to=1581886799999 HTTP/1.0

X-IV-Authorization: Session 771E111111111122222222223333A273

**Параметры запроса**

| * **Название** |  | * **Описание** |
| --- | --- | --- |
| * status | O | * Статус операции. Возможные значения:   DECLINED - операция отклонена клиентом  ERROR - неуспешная операция  PROCESSING - операция не завершена  SUCCESS - успешная операция  FINISHED - операция завершена (статус SUCCESS, ERROR или DECLINED)   * Если параметр не указан, фильтрация по статусу не выполняется |
| * from | O | * Начало диапазона (Java Timestamp — long, включительно) |
| * to | O | * Конец диапазона (Java Timestamp — long, не включительно) |
| * pan | O | * PAN карты. Может быть указан маскированный PAN карты, (маска NNNNNNxxxxxxNNNN), либо последние 4 цифры номера карты (NNNN) |
| * token | O | * Полный токен операции или часть токена, начинающаяся с его первого символа |
| * merchantId | O | * Идентификатор магазина |
| * merchantTrx | O | * Номер транзакции магазина |
| * cardId | O | * Идентификатор карты в системе |

**Поля ответа**

В ответе будет только поле count, содержащее количество операций, подошедших под условия фильтрации.

Если в фильтре по идентификатору магазина был указан несуществующий ИД, то Сервис вернёт код ошибки MERCHANT\_NOT\_FOUND.

А если там был указан идентификатору существующего магазина, но у оператора нет прав на работу с этим магазином, то Сервис вернёт код ошибки ACCESS\_DENIED.

## Получение списка регистраций карт

Запрос получения списка регистраций карт:

GET /api/v4/<portal\_id>/merchant/history/card-registration

Пример запроса получения списка регистраций карт:

GET /api/v4/F99B664D6D9B2DB0806656F28E3A4DF9/merchant/history/card-registration?from=1581282000000&limit=10&offset=0&orderDir=DESC&orderName=merchantTrx&status=FINISHED&to=1581886799999 HTTP/1.0

X-IV-Authorization: Session 771E111111111122222222223333A273

**Параметры запроса**

Такие же, как и у запроса размера списка регистраций карт. Добавляются параметры для сортировки и постраничного просмотра

| * Параметр |  | * Описание |
| --- | --- | --- |
| * offset | * O | * Смещение (для постраничного отображения). По умолчанию 0 |
| * limit | * O | * Количество элементов (для постраничного отображения). По умолчанию 10 |
| * orderDir | * O | * Направление сортировки (DESC/ASC). По умолчанию DESC Сортировка выполняется по дате начала операции |

**Поля ответа**

|  |  |  |
| --- | --- | --- |
| * **Название** |  | * **Описание** |
| * token | R | * Идентификатор операции регистрации карты |
| * startedAt | R | * Дата старта операции (Java Timestamp — long) |
| * finishedAt | C | * Дата завершения операции (Java Timestamp — long) |
| * pan | C | * Маскированный PAN карты, маска NNNNNNxxxxxxNNNN |
| * merchantId | R | * Идентификатор магазина |
| * merchantTrx | R | * Идентификатор транзакции в магазине |
| * status | R | * Статус транзакций (DECLINED, ERROR, PROCESSING, SUCCESS) |
| * extendedCode | O | * Расширенный код завершения транзакции (буквенный) |
| * responseCode | O | * Код ответа процессинга |
| * isFullyAuthenticated | C | * Была ли успешная 3-D Secure аутентификация. Возможные значения: true, false. Заполняется только если была попытка. |
| * orderStatus | O | * Статус заказа. Возможные значения: * UNKNOWN - статус заказа неизвестен (или нет дополнительно статуса у данного заказа) * REGISTERED\_AT\_MERCHANT - операция зарегистрирована у продавца (магазина) * REGISTRATION\_ATTEMPTS\_EXHAUSTED - исчерпаны попытки регистрации операции у продавца (магазина) |
| * amountReverted | C | * Присутствует со значением "true", только если все списанные в процессе регистрации средства были успешно возвращены на карту |

Операции в списке отсортированы по дате старта.

**Пример ответа**

[

 {

  "token" : "234234LK234",

  "startedAt": "1415266887057",

  "finishedAt": "1415266887057",

  "pan": "5454xxxxxxxx5454",

  "merchantId": "TestMerchant",

  "merchantTrx": "TX-582346237845",

  "status": "SUCCESS",

  "extendedCode": "OK",

  "responseCode": "000",

  "isFullyAuthenticated": true,

  "orderStatus": "REGISTERED\_AT\_MERCHANT",

  "amountReverted": true

 }

]

## Получение информации об одной операции регистрации карты

Запрос получения информации об одной операции верификации карты:

GET /api/v4/<portal\_id>/merchant/history/card-registration/<token>

Параметров у запроса нет.

Пример запроса получения информации об одной операции регистрации карты:

GET /api/v4/EC301C372085C1251C4E4EEBBD6E8AAB/merchant/history/card-registration/1999FNFTKJC03 HTTP/1.0

X-IV-Authorization: Session 771E111111111122222222223333A273

**Поля ответа**

В ответе будет полное описание одной операции. По сравнению с сокращённым описанием операции, как в списке выше, дополнительно будут такие данные:

| * **Название** |  | * **Описание** |
| --- | --- | --- |
| * cardId | * C | * Идентификатор зарегистрированной карты |

**Пример ответа**

{

 "token" : "234234LK234",

 "startedAt": "1415266887057",

 "finishedAt": "1415266887057",

 "pan": "5454xxxxxxxx5454",

 "merchantId": "TestMerchant",

 "merchantTrx": "TX-582346237845",

 "status": "SUCCESS",

 "extendedCode": "OK",

 "responseCode": "000",

 "isFullyAuthenticated": true,

 "orderStatus": "REGISTERED\_AT\_MERCHANT",

 "amountReverted": true,

 "cardId": "34234NNM234234"

}

# Платежные карты

## Получение информации об одной карте

Для получения информации по отдельной зарегистрированной карте, доступной оператору магазина, необходимо указать в запросе ее идентификатор:

GET /api/v4/<portal\_id>/merchant/card/<id>

Пример запроса получения информации по отдельной зарегистрированной карте:

GET /api/v4/4EEA096E50948B54C32C32455AD6BCAC/merchant/card/C7MYTVV3XIP4 HTTP/1.0

X-IV-Authorization: Session 771E111111111122222222223333A273

**Поля ответа**

| * **Поле** |  | * **Описание** |
| --- | --- | --- |
| * id | * R | * Идентификатор карты |
| * status | * R | * Статус карты (HIDDEN, UNVERIFIED, VERIFIED, VERIFICATION\_FAILED, VERIFYING) |
| * blocked | * O | * true, если карта заблокирована |
| * paymentSystem | * R | * Платежная система карты (VISA/MASTERCARD/MIR/AMEX/UNIONPAY/ELCART/GCLOCAL) |
| * pan | * R | * Маскированный номер карты |
| * expiry | * O | * Срок действия карты в формате MMYY |
| * expired | * O | * Признак просроченности карты (передаётся true, если просрочена) |
| * cardholder | * O | * Имя владельца карты |
| * registeredAt | * R | * Время регистрации карты в системе (Unix time в ms) |
| * merchantId | * R | * Идентификатор магазина |
| * recurrentSupported | * O | * Флаг поддержки картой рекуррентных платежей. Возможные значения: true/false Присутствует, если проверка поддержки рекуррентов выполнялась при регистрации/добавлении карты и у этой проверки был не-UNKNOWN-статус |
| * virtual | * O | * Признак виртуальной карты, заполняется только когда карта виртуальная, т.е. возможное значение только true |
| * externalToken | * С | * Обязательный параметр из запроса старта, если карта виртуальная |
| * additiveCardId | * O | * Уникальный токен платёжной карты. Формат AN29 |

**Пример запроса и ответа:**

// Запрос

GET /api/v4/1FB196F908BDCFDF3126B1AB10E4529A/merchant/card/34234NNM234234

// Тело ответа

{

    id: "34234NNM234234",

    status: "VERIFIED",

    //blocked: true,

    paymentSystem: "VISA",

    pan: "4756xxxxxxxx6346",

    expiry: "1225",

    //expired: true,

    cardholder: "James Bond",

    registeredAt: "1415266887057",

    merchantId: "TESTMERCHANT"

}

## Получение размера списка карт

Для получения размера списка зарегистрированных карт, относящихся к доступным оператору магазинам, необходимо выполнить следующий запрос:

GET /api/v4/<portal\_id>/merchant/card/count

Пример запроса получения размера списка зарегистрированных карт:

GET /api/v4/F99B664D6D9B2DB0806656F28E3A4DF9/merchant/card/count?from=1581282000000&limit=10&offset=0&orderDir=DESC&orderName=merchantTrx&status=FINISHED&to=1581886799999 HTTP/1.0

X-IV-Authorization: Session 771E111111111122222222223333A273

**Параметры запроса**

| * **Название** |  | * **Значение** |
| --- | --- | --- |
| * registeredFrom | * O | * Начало диапазона даты регистрации карты (Java Timestamp — long) (включительно) |
| * registeredTo | * O | * Конец диапазона даты регистрации карты (Java Timestamp — long) (не включительно) |
| * pan | * O | * PAN карты, маскированный (маска NNNNNNxxxxxxNNNN) или последние 4 цифры номера карты (NNNN). |
| * merchantId | * O | * Идентификатор магазина |
| * status | * O | * Статус карты (HIDDEN, UNVERIFIED, VERIFIED, VERIFICATION\_FAILED, VERIFYING) |

**Поля ответа**

В ответе будет только поле count, содержащее количество транзакций, подошедших под условия фильтрации.

Если в фильтре по ИД магазина был указан несуществующий ИД, то Сервис вернёт код ошибки MERCHANT\_NOT\_FOUND.

А если там был указан ИД существующего магазина, но у оператора нет прав на работу с этим магазином, то Сервис вернёт код ошибки ACCESS\_DENIED.

## Получение списка карт

Для получения списка зарегистрированных карт, относящихся к доступным оператору магазинам, необходимо выполнить следующий запрос:

GET /api/v4/<portal\_id>/merchant/card/

Пример запроса получения списка карт:

GET /api/v4/F99B664D6D9B2DB0806656F28E3A4DF9/merchant/card?limit=10&offset=0&orderDir=DESC&orderName=merchantName&registeredFrom=1581282000000&registeredTo=1581886799999 HTTP/1.0

X-IV-Authorization: Session 771E111111111122222222223333A273

**Параметры запроса**

 Такие же, как и у запроса размера списка. Добавляются параметры для сортировки и постраничного просмотра:

| * **Параметр** |  | * **Описание** |
| --- | --- | --- |
| * offset | * O | * Смещение (для постраничного отображения). По умолчанию 0 |
| * limit | * O | * Количество элементов (для постраничного отображения). По умолчанию 10. Максимум 1000 |
| * orderDir | * O | * Направление сортировки (DESC/ASC). По умолчанию DESC. Сортировка выполняется по дате регистрации карты |

Ответ Сервиса содержит массив объектов, каждый из которых описывает отдельную карту. Поля ответа такие же, как в ответе на запрос одной карты, только отсутствует поле cardholder.

## Деактивация карты

Для деактивации карты по идентификатору необходимо выполнить следующий запрос без параметров:

POST /api/v4/<portal\_id>/merchant/card/<id>/delete

Пример запроса деактивации карты:

POST /api/v4/4EEA096E50948B54C32C32455AD6BCAC/merchant/card/C7MYTVV3XIP4/delete HTTP/1.0

Content-Length: 0

X-IV-Authorization: Session 771E111111111122222222223333A273

После успешной обработки запроса Сервис возвращает пустой ответ со статусом HTTP 200 OK.

Карта будет переведена в статус HIDDEN и перестанет быть доступной для платежей.

# Отчёты

## Запрос создания отчёта

Для того, чтобы построить отчёт, нужно сначала отправить запрос создания отчёта:

POST /api/v4/<portal\_id>/merchant/report/create

**Параметры запроса**

| * **Название** |  | * **Значение** |
| --- | --- | --- |
| * reportType | * O | Тип отчёта. Возможные значения:  PAYMENT\_TRX – транзакционный отчёт  PAYMENT\_TRX\_SUMMARY – сводный транзакционный отчёт.  По умолчанию PAYMENT\_TRX |
| * format | * O | Формат отчёта. Возможные значения:  CSV, XLS, XLSX, PDF.  По умолчанию CSV |
| * from | * O | Начало временного диапазона (Java Timestamp — long) |
| * to | * O | Конец временного диапазона (Java Timestamp — long) |
| * currency | * O | Трехсимвольный код валюты транзакции |
| * merchantId | * O | Идентификатор магазина |
| * type | * O | Тип транзакции (CARD\_TO\_CARD, CARD\_TO\_ACCOUNT, ACCOUNT\_TO\_CARD, GAMBLING, ACQUIRING, TEST\_PAYMENT).   * Можно указать несколько значений через запятую |
| * status | * O | Статус транзакций.  Возможные значения:   * DECLINED – операция отклонена клиентом * ERROR – неуспешная операция * PROCESSING – операция не завершена * SUCCESS – успешная операция * INTERIM\_SUCCESS – неоконченная операция с промежуточным успешным результатом * FINISHED – операция завершена (статус SUCCESS, INTERIM\_SUCCESS, ERROR или DECLINED)   Если параметр не указан, фильтрация по статусу не выполняется   * Можно указать несколько значений через запятую |
| * onlyAuthorized | * O | * Фильтрация только транзакций с успешной авторизацией, т. е. с кодом ответа процессинга «000», «00» или «0». * Возможные значения: true / false |
| * limit | * O | * Максимальное количество записей в отчёте. * Максимальное значение параметра: 500000. Значение по умолчанию: 150000. |

**Поля ответа**

| * **Поле** |  | * **Описание** |
| --- | --- | --- |
| * token | * R | * Токен отчёта, сгенерированный Сервисом |

При получении запроса на создание отчёта Сервис ставит этот запрос в общую очередь. Т.е. отчёты не строятся сразу, а могут некоторое время лежать в очереди и ждать, пока будут построены отчёты, заказанные ранее. Это позволяет не перегружать Сервис параллельным созданием нескольких отчётов.

PAYMENT\_TRX – это обычный отчёт от платёжных транзакциях. Каждая строка отчёта - одна транзакция. Записи отчёта отсортированы по дате старта транзакции.

PAYMENT\_TRX\_SUMMARY – сводный отчёт о платёжных транзакциях. Платёжные транзакции, по которым строится отчёт, сгруппированы по значению расширенного кода результата (Extended Code). Каждая строка отчёта содержит значение кода и количество транзакций с таким кодом.

## Получение информации об одном отчёте

После того, как КИ отправил запрос на создание отчёта, он может периодически запрашивать информацию о нём, чтобы определить, когда отчёт будет готов и его можно будет скачать. Формат запроса информации об отчёте:

GET /api/v4/<portal\_id>/merchant/report/<token>

Где token – токен отчёта

Пример запроса получения статуса отчёта:

GET /api/v4/F99B664D6D9B2DB0806656F28E3A4DF9/merchant/report/123456AAC HTTP/1.0

X-IV-Authorization: Session 771E111111111122222222223333A273

Параметров у запроса нет.

**Поля ответа**

| * **Название** |  | * **Значение** |
| --- | --- | --- |
| * token | * R | * Токен отчёта |
| * format | * R | * Формат отчёта |
| * addedAt | * R | * Время добавления отчёта в очередь |
| * description | * R | * Описание параметров построения отчёта. Это Json-объект, каждое поле которого соответствует одному фильтру транзакций |
| * status | * R | * Статус отчёта. Возможные значения: * WAITING – отчёт ожидает в очереди * IN\_PROGRESS – отчёт строится * DONE – отчёт успешно построен * FAILED – при построении отчёта произошла ошибка |

## Скачивание отчёта

После того, как отчёт успешно построен, его можно скачать. Формат запроса скачивания отчёта:

GET /api/v4/<portal\_id>/merchant/report/<token>/content

где token – токен отчёта

Пример запроса скачивания отчёта:

GET /api/v4/F99B664D6D9B2DB0806656F28E3A4DF9/merchant/report/123456AAC/content HTTP/1.0

X-IV-Authorization: Session 771E111111111122222222223333A273

Параметров у запроса нет.

В ответ Сервис вернёт запакованный отчёт. Заголовок Content-Type при этом выставляется в значение «application/zip».

Один и тот же отчёт можно скачать несколько раз.

## Удаление отчёта

Сервис автоматически удаляет отчёты, у которых закончился срок жизни. Но КИ может и сам удалить отчёт, если он уже не нужен. Формат запроса удаления отчёта:

POST /api/v4/<portal\_id>/merchant/report/<token>/delete

где token – токен отчёта

Пример запроса удаления отчёта:

GET /api/v4/F99B664D6D9B2DB0806656F28E3A4DF9/merchant/report/123456AAC/delete HTTP/1.0

X-IV-Authorization: Session 771E111111111122222222223333A273

Параметров у запроса нет.

В случае успешного удаления отчёта Сервис вернёт пустой ответ. В случае неуспешного удаления отчёта Сервис вернёт сообщение об ошибке.

## Получение списка отчётов

КИ может запросить список отчётов. Формат запроса:

GET /api/v4/<portal\_id>/merchant/report

Пример запроса получения списка отчётов:

GET /api/v4/F99B664D6D9B2DB0806656F28E3A4DF9/merchant/report HTTP/1.0

X-IV-Authorization: Session 771E111111111122222222223333A273

**Параметры запроса**

| **Параметр** |  | **Описание** |
| --- | --- | --- |
| offset | O | Смещение (для постраничного отображения). По умолчанию 0 |
| limit | O | Количество элементов (для постраничного отображения).  Максимальное значение: 100  По умолчанию 10 |

 Ответ Сервиса содержит массив объектов, каждый из которых описывает отдельный отчёт. В этот массив попадают только отчёты, запрос создания которых был получен через тот же портал, что и запрос списка отчётов, и от имени того же оператора, что и запрос списка отчётов. Другими словами, каждый оператор магазинной консоли получит от Сервиса только список своих отчётов, и не получит отчёты, созданные другими операторами.

**Поля ответа**

| * **Название** |  | * **Значение** |
| --- | --- | --- |
| * totalCount | * R | * Общее количество отчётов |
| * items | * R | * Массив, каждый элемент которого описывает один отчёт. Поля описания отчёта такие же, как в ответе на запрос информации об одном отчёте |

# Конфигурация магазина

Запрос для просмотра конфигурации магазина:

GET /api/v4/<portal\_id>/merchant/<merchant\_id>/config

Где <merchant\_id> - идентификатор магазина.

**Поля ответа**

| * **Название** |  | * **Значение** |
| --- | --- | --- |
| * name | * R | * Название платежной схемы |
| * className | * R | * Имя класса платежной схемы |
| * schemaBlocked | * R | * Признак блокировки схемы |
| * identifier | * R | * Идентификатор правила выбора платежной схемы |
| * operationType | * R | * Тип операции правила |
| * ruleBlocked | * O | * Признак блокировки правила |
| * merchantId | * R | * Идентификатор магазина |
| * merchantName | * O | * Название магазина в конфигурации схемы |
| * threeDSMode | * O | * Версия 3-D Secure |
| * mcc | * R | * Код категории магазина |
| * cpaUrl | * O | * Адрес проверки платежа |
| * rpUrl | * O | * Адрес проверки доступности операции |
| * rpSigningDisabled | * O | * Отключение подписи RPReq |

**Пример запроса и ответа:**

// Запрос

GET /api/v4/F99B664D6D9B2DB0806656F28E3A4DF9/merchant/TESTMERCHANT/config

// Тело ответа

[

{

"name": "Test Internet Acquiring GCS",

"className": "com.intervale.ecp.payment.schemas.GCSAcquiringSchema",

"schemaBlocked": false,

"identifier": "PS1",

"operationType": "INTERNET\_ACQUIRING",

"ruleBlocked": false,

"merchantId": "99887766",

"merchantName": "Test Merchant",

"threeDSMode": "THREE\_DS\_1\_AND\_2",

"mcc": "1234",

"cpaUrl": "https://testmerchant.ru/cpa.php",

"rpUrl": "https://testmerchant.ru/rp.php",

"rpSigningDisabled": false

},

{

"name": "Test Internet Acquiring GCS RUB 2",

"className": "com.intervale.ecp.payment.schemas.GCSAcquiringSchema",

"schemaBlocked": false,

"identifier": "PS2",

"operationType": "INTERNET\_ACQUIRING",

"ruleBlocked": false,

"merchantId": "11223344",

"merchantName": "Test Merchant RUB",

"threeDSMode": "THREE\_DS\_1",

"mcc": "5678",

"cpaUrl": "https://testmerchant.ru/cpa.php",

"rpUrl": "https://testmerchant.ru/rp.php",

"rpSigningDisabled": false

},

{

"name": "Test CardReg GCS",

"className": "com.intervale.ecp.card.schemas.GCSCardRegistrationSchema",

"schemaBlocked": false,

"identifier": "CRS1",

"operationType": "CARD\_REGISTRATION",

"ruleBlocked": false,

"merchantId": "11223344",

"merchantName": "Test Merchant Reg",

"threeDSMode": "THREE\_DS\_1",

"cpaUrl": "https://testmerchant.ru/cpa.php",

"rpUrl": "https://testmerchant.ru/rp.php",

"rpSigningDisabled": false

}

]

# Приложение 1. Коды ошибок выполнения запросов

Таблица 1 Коды ошибок выполнения запросов

|  |  |  |
| --- | --- | --- |
| HTTP-код | Наименование ошибки | Описание |
| 400  Bad Request | ACCESS\_DENIED | Отказано в доступе. Необходимо проверить правильность введенных данных или обратиться в службу технической поддержки. Параметры:   * **subject** - сущность, к которой не удалось получить доступ (PARTNER, PORTAL, MERCHANT, USER). * **reason** - причина отказа в доступе (VERSION, OPERATION, INVALID, PAUSED, EXPIRED, BLOCKED). * **failedAttempts** - количество неудачных попыток предъявления пароля/PIN. * **delay** - количество секунд до снятия сущности с паузы.   Пример:  [subject=USER, reason=BLOCKED] - профиль клиента заблокирован;  [subject=USER, reason=INVALID] - неверный Id клиента, Id сессии или PIN устройства. Попытка продолжить операцию в анонимном режиме, но начатаю в не анонимном режиме. Для ситуации с неверным PIN устройства в поле failedAttempts помещается значение счетчика попыток предъявления PIN.  [subject=USER, reason=PAUSED] - профиль поставлен на паузу для запрошенной операции.  [subject=USER, reason=OPERATION] - профилю клиента запрещено выполнение запрошенной операции;  [subject=PORTAL, reason=INVALID] - неверный Id портала или попытка использовать сессию, открытую для другого партнера;  [subject=PORTAL, reason=BLOCKED] - портал заблокирован;  [subject=PORTAL, reason=OPERATION] - запрошенная операция не настроена или к ней запрещен доступ с указанного портала;  [subject=PORTAL, reason=VERSION] - портал недоступен для указанной версии приложения или портал требует обязательного указания версии приложения (если в запросе версия приложения не указана);  [subject=PARTNER, reason=BLOCKED] - партнер заблокирован.  Cочетание [subject=USER, reason=EXPIRED] преобразуется в отдельный код ошибки SESSION\_EXPIRED. |
| SESSION\_EXPIRED | Время сессии истекло.  Пример:  {"error":"SESSION\_EXPIRED"} |
| ALREADY\_REGISTERED | Номер телефона, e-mail или карта уже зарегистрированы в системе. Необходимо использовать уникальные данные для регистрации в системе. Параметр:   * **entity** – тип сущности (EMAIL, MSISDN, PROFILE, CARD)   Пример:  {"error":"ALREADY\_REGISTERED", “entity”:”EMAIL”} |
| CARD\_NOT\_SUPPORTED | Карта не поддерживается. Необходимо использовать установленный тип платежных инструментов.  Пример:  {"error":"CARD\_NOT\_SUPPORTED"} |
| CARD\_RANGE\_BLOCKED | Диапазон карт заблокирован. Параметр:   * **side** – указывает, является ли карта источником или приёмником денежных средств (DST, SRC)   Пример:  {"error":"CARD\_RANGE\_BLOCKED", “side”: “SRC”} |
| CARD\_BLOCKED | Карта заблокирована. Параметр:   * **side** – указывает, является ли карта источником или приёмником денежных средств (DST, SRC)   Пример:  {"error":"CARD\_BLOCKED", “side”: “SRC”} |
| CARD\_EXPIRED | Истек срок действия карты. Параметр:   * **side** – указывает, является ли карта источником или приёмником денежных средств (DST, SRC)   Пример:  {"error":"CARD\_EXPIRED", “side”: “SRC”} |
| CARD\_NOT\_VERIFIED | Карта не верифицирована.  Пример:  {"error":"CARD\_NOT\_VERIFIED"} |
| CARD\_TOKEN\_UNRESOLVED | Произошла ошибках при расшифровке токена Apple Pay, проверке его подписи и валидации значений полей. Пример:  {"error":"CARD\_TOKEN\_UNRESOLVED"} |
| DUPLICATE\_TITLE | Название карты или шаблона не уникально. Необходимо использовать уникальное название карты и шаблона.  Пример:  {"error":"DUPLICATE\_TITLE"} |
|
| EXTRA\_PARAMETER | В запросе указан лишний параметр. Необходимо проверить правильность запроса.  Пример:  {"error":"EXTRA\_PARAMETER","param":"otp"}  где param – имя параметра. |
| INVALID\_PARAMETER | Недопустимое значение параметра. Необходимо проверить правильность введенных параметров.  Пример:  {"error":"INVALID\_PARAMETER",  "param":"phonenumber", "detail":"Wrong  phone number format!"}  где param – имя параметра.  Также ошибка может означать проблемы с параметрами платежа при работе с шаблонами платежей. В этом случае параметра param в сообщении об ошибке отсутствует. |
| INVALID\_PASSWORD | Логин не существует или пароль неверен или нужна двухфакторная авторизация. Необходимо проверить правильность введенных данных или обратиться в службу технической поддержки. Параметры:   * **failedAttempts** - количество неудачных попыток предъявления пароля; * **attemptsLeft** - количество оставшихся попыток. Указывается для операций, выполняемых в рамках сессии (смена/проверка пароля, удаление профиля). * **delay** - количество секунд до снятия пароля с паузы. Указывается, если пароль был поставлен на паузу после неудачной попытки предъявления пароля.   Пример:  {"error":"INVALID\_PASSWORD","failedAttempts":2} |
| INVALID\_PAYMENT\_STATE | Платеж имеет неподходящее для выполнения запрошенной операции состояние.  Возможные варианты: код используется, если отправка уведомления запрошена для незавершенной транзакции.  Пример:  {"error":"INVALID\_PAYMENT\_STATE"} |
| INVALID\_OTP | Одноразовый пароль, отправленный на телефон/почту, не прошел проверку, или Сервис не отправлял клиенту одноразовый пароль. Необходимо запросить новый одноразовый пароль или обратиться в службу технической поддержки.  Пример:  {"error":"INVALID\_OTP",  "failedAttempts":2,  "attemptsLeft":1}  где failedAttempts – количество неудачных попыток предъявления OTP;  attemptsLeft – количество оставшихся попыток. |
| INVALID\_REQUEST | Запрос не прошел валидацию. Необходимо проверить правильность и наличие всех обязательных параметров в запросе.  Пример:  {"error": "INVALID\_REQUEST", "detail": "May not be empty: 'otp'"}  где detail – описание ошибки. |
| MISSING\_PARAMETER | В запросе отсутствует обязательный параметр. Необходимо проверить наличие всех обязательных параметров в запросе.  Пример:  {"error":"MISSING\_PARAMETER","param":"otp"}  где param – имя параметра. |
| OTP\_EXPIRED | Истек срок действия одноразового пароля. Необходимо запросить новый одноразовый пароль.  Пример:  {"error":"OTP\_EXPIRED"} |
| OTP\_PAUSED | Генерация нового одноразового пароля поставлена на паузу. После тайм-аута пауза будет снята автоматически.  Пример:  {"error":"OTP\_PAUSED", "delay":135}  где delay – количество секунд до снятия с паузы. |
| OTP\_BLOCKED | Одноразовый пароль заблокирован из-за превышения количества попыток ввода. Необходимо запросить новый одноразовый пароль.  Пример:  {"error":"OTP\_BLOCKED"} |
| PASSWORD\_EXPIRED | Срок действия пароля истек. Необходимо запросить новый одноразовый код.  Пример:  {"error":"EXPIRED\_EXPIRED"} |
| PASSWORD\_PAUSED | Действие пароля приостановлено после неудачных попыток ввести правильный пароль. Необходимо проверить правильность введенных данных или обратиться в службу технической поддержки.  Параметры:   * **delay** – количество секунд до снятия с паузы.   Пример:  {"error":"PASSWORD\_PAUSED", "delay":125} |
| PROFILE\_BLOCKED | Операция отклонена, т.к. профиль клиента заблокирован. Необходимо обратиться в службу технической поддержки.  Пример:  {"error":"PROFILE\_BLOCKED"} |
| PROFILE\_NOT\_ACTIVATED | Профиль не активирован. Необходимо подтвердить регистрацию профиля с помощью одноразового пароля.  Пример:  {"error":"PROFILE\_NOT\_ACTIVATED"} |
| TRANSACTION\_NOT\_SUCCESSFUL | Транзакция не успешна.  Пример:  {"error":"TRANSACTION\_NOT\_SUCCESSFUL"} |
| UNSUPPORTED\_VERSION | Указанная версия протокола не поддерживается. Параметры:   * **minVersion** - минимальная поддерживаемая версия протокола. Праматер указывается, если в запросе указана недопустимо ранняя версия протокола; * **maxVersion** - максимальная поддерживаемая версия протокола. Праматер указывается, если в запросе указана недопустимо поздняя версия протокола.   Пример:  {"error":"UNSUPPORTED\_VERSION", "minVersion":3} |
| UNEXPECTED\_REQUEST | Непредусмотренный запрос. После получения такой ошибки в ходе выполнения транзакции рекомендуется выполнить запрос статуса платежа (верификации/регистрации). Ответ сервера даст понимание того, что нужно делать для продолжения транзакции.  Запрос имеет корректный формат, но Сервис не может его обработать, так как это противоречит текущему состоянию транзакции. Код используется для сообщений, относящихся к выполнению платежа или регистрации карты.  Возможные варианты: пришел запрос на старт платежа с токеном, который уже используется для другого платежа. Пришел запрос подтверждения оферты, а платеж уже завершен.  Пример:  {"error":"UNEXPECTED\_REQUEST"} |
| WEAK\_PASSWORD | Новый пароль не отвечает требованиям стойкости. Необходимо задать пароль в соответствии с установленными правилами.  Пример:  {"error":"WEAK\_PASSWORD"} |
| MERCHANT\_BLOCKED | Магазин заблокирован.  Пример:  {"error":"MERCHANT\_BLOCKED"} |
| TERMINAL\_NOT\_FOUND | Терминал с указанным идентификатором не найден.  Пример:  {"error":"TERMINAL\_NOT\_FOUND"} |
| REPORT\_NOT\_FOUND | Отчёт с указанным токеном не найден.  Пример:  {"error":"REPORT\_NOT\_FOUND"} |
| ACQUIRING\_TYPE\_NOT\_SUPPORTED\_BY\_MERCHANT | Данный тип эквайринга не поддерживается (или не настроен) для данного магазина.  Пример:  {"error":"ACQUIRING\_TYPE\_NOT\_SUPPORTED\_BY\_MERCHANT"} |
| SRC\_CARD\_NOT\_SUPPORTED | Карта-источник не поддерживается или не подходит для данной операции. Код используется, если для карты-источника не удалось подобрать платежную схему, или бренд карты не поддерживается Сервисом.  Пример:  {"error":"SRC\_CARD\_NOT\_SUPPORTED"} |
| CURRENCY\_NOT\_SUPPORTED | Неподдерживаемая валюта.  Пример:  {"error":"CURRENCY\_NOT\_SUPPORTED"} |
| SRC\_TYPE\_NOT\_SUPPORTED | Тип источника средств не поддерживается.  Пример:  {"error":"SRC\_TYPE\_NOT\_SUPPORTED"} |
| INVALID\_TRANSACTION\_STATE | Транзакция имеет неподходящее для выполнения запрошенной операции состояние. Код используется для сообщений, относящихся к выполнению платежа или регистрации карты.  Пример:  {"error":"INVALID\_TRANSACTION\_STATE"} |
| TRANSACTION\_NOT\_FINISHED | Транзакция не завершена.  Пример:  {"error":"TRANSACTION\_NOT\_FINISHED"} |
| CURRENCY\_MISMATCH | Валюта запроса не совпадает с ожидаемой.  Пример:  {"error":"CURRENCY\_MISMATCH"} |
| AMOUNT\_TOO\_SMALL | Сумма, указанная в запросе, меньше допустимой.  Пример:  {"error":"AMOUNT\_TOO\_SMALL"} |
| AMOUNT\_TOO\_BIG | Сумма, указанная в запросе, больше допустимой.  Пример:  {"error":"AMOUNT\_TOO\_BIG"} |
| AMOUNT\_MISMATCH | Сумма, указанная в запросе, не совпадает с ожидаемой.  Пример:  {"error":"AMOUNT\_MISMATCH"} |
| COMMISSION\_MISMATCH | Комиссия, указанная в запросе, не совпадает с ожидаемой.  Пример:  {"error":" COMMISSION\_MISMATCH"} |
| TRANSACTION\_BUSY | Транзакция занята.  Данный код ошибки может быть получен в ответ на запрос возврата средств или завершения авторизации.  Он означает, что в данный момент для транзакции уже выполняется запрос и выполнение нового запроса невозможно. Клиентский интерфейс при получении данного кода должен подождать некоторое время и запросить актуальный статус транзакции.  Пример:  {"error":" TRANSACTION\_BUSY "} |
| OPERATION\_NOT\_AVAILABLE | Операция недоступна.  где reason - причина недоступности.  Пример:  {"error":"OPERATION\_NOT\_AVAILABLE", "reason":"otp"} |
| 404 Not Found | CARD\_NOT\_FOUND | Карта не найдена. Необходимо зарегистрировать карту в системе. Параметр:   * **side** – указывает, является ли карта источником или приёмником денежных средств (DST, SRC)   Пример:  {"error":"CARD\_NOT\_FOUND"} |
| EMAIL\_NOT\_FOUND | Е-mail не указан в профиле пользователя. Необходимо указать e-mail в профиле пользователя.  Пример:  {"error":"EMAIL\_NOT\_FOUND"} |
| LOGIN\_NOT\_FOUND | Логин не найден. Необходимо зарегистрироваться в системе.  Пример:  {"error":"LOGIN\_NOT\_FOUND"} |
| LOYALTY\_NOT\_FOUND | Сущность "лояльность" не найдена.  Пример:  {"error":"LOYALTY\_NOT\_FOUND"} |
| MERCHANT\_NOT\_FOUND | ТСП не найдено. Необходимо обратиться в службу технической поддержки.  Пример:  {"error":"Merchant not found"} |
| NOT\_FOUND | Сущность не найдена. Это может быть платежная транзакция, портал и т.п. сущности, для которых не заведены отдельные коды ошибок. Также ошибка используется, если в HTTP-запросе указан URL, который по формату не подходит для запросов Сервиса.  Пример:  {"error":"NOT\_FOUND"} |
| PAYMENT\_NOT\_FOUND | Платеж с указанным алиасом не найден. Под платежом понимается описание платежа и платежной схемы. Необходимо проверить параметры запроса или обратиться в службу технической поддержки.  Пример:  {"error":"PAYMENT\_NOT\_FOUND"} |
| TRANSACTION\_NOT\_FOUND | Транзакция не найдена. Подразумевается платежная транзакция или транзакция регистрации карты.  Пример:  {"error":"TRANSACTION\_NOT\_FOUND"} |
| PORTAL\_NOT\_FOUND | Портал не найден.  Пример:  {"error":"PORTAL\_NOT\_FOUND"} |
| 405 Method Not Allowed | METHOD\_NOT\_ALLOWED | HTTP-метод не поддерживается для указанного запроса. Необходимо использовать только допустимые HTTP-методы. В HTTP-ответе будет заголовок Allow, содержащий допустимые для запроса HTTP-методы.  Пример:  {"error":"METHOD\_NOT\_ALLOWED"} |
| 500 Internal Server Error | INTERNAL\_ERROR | Внутренняя ошибка Сервиса. Необходимо обратиться в службу технической поддержки.  Пример:  {"error":"INTERNAL\_ERROR"} |
|  |
|  |
|  |
| EXTERNAL\_SYSTEM\_ERROR | Ошибка при взаимодействии с внешней системой.  Пример:  {"error":"EXTERNAL\_SYSTEM\_ERROR"} |
| 503 Service Unavailable | MAINTENANCE | Сервис временно недоступен. Проводятся регламентные работы.  Параметр:   * **scope** – область действия регламентных работ.   Возможные значения параметра:   * Server - регламентные работы на сервере. В этом случае сервер на любые запросы отвечает сообщением об ошибке с кодом MAINTENANCE и параметром scope=Server. Кроме того, в ответе сервера может содержаться следующее поле: finishTS - примерное время окончания работ (Unix time в ms). Параметр необязательный. * Payment - регламентные работы только в рамках одной платежной схемы. В данном случае сервер отвечает штатно на любые запросы, кроме запросов, связанных с проведением платежа по данной платежной схеме или комиссиями по платежу. На эти запросы он отвечает сообщением об ошибке с кодом MAINTENANCE и параметром scope=Payment.   Пример:  {"error":"MAINTENANCE", "scope":"Payment" } |