



**Протокол взаимодействия между системой
мерчанта и коннектором Pay-logic.**

Программное обеспечение «PSP Pay-logic».

Руководство разработчика

АННОТАЦИЯ

| Содержит описание принципов работы, настройки и интеграции в системы партнеров

Версия руководства: 2.6

Руководство актуально для кабинета мерчанта версий 1.11

2008–2026 ООО «Софт-Лоджик», г. Барнаул, Россия

Данный документ входит в комплект поставки программных продуктов.

Права использования данного документа предусмотрены соответствующим лицензионным договором.

ООО «Софт-Лоджик»

656006, г. Барнаул, Малахова ул., дом 146в

Тел: (3852) 72-27-27

© Soft-logic

Web: <https://soft-logic.ru/>

Mail: info@soft-logic.ru

ОГЛАВЛЕНИЕ

ИСТОРИЯ ИЗМЕНЕНИЙ.....	5
ИЗМЕНЕНИЯ В ВЕРСИИ ДОКУМЕНТА 1.0.....	5
ИЗМЕНЕНИЯ В ВЕРСИИ ДОКУМЕНТА 2.0.....	5
ИЗМЕНЕНИЯ В ВЕРСИИ ДОКУМЕНТА 2.1.....	5
ИЗМЕНЕНИЯ В ВЕРСИИ ДОКУМЕНТА 2.2.....	6
ИЗМЕНЕНИЯ В ВЕРСИИ ДОКУМЕНТА 2.3.....	6
ИЗМЕНЕНИЯ В ВЕРСИИ ДОКУМЕНТА 2.4.....	6
ИЗМЕНЕНИЯ В ВЕРСИИ ДОКУМЕНТА 2.5.....	7
ИЗМЕНЕНИЯ В ВЕРСИИ ДОКУМЕНТА 2.6.....	7
1 ТЕРМИНЫ И СОКРАЩЕНИЯ.....	8
2 ПРЕДПОЛАГАЕМАЯ АУДИТОРИЯ.....	9
3 ВВЕДЕНИЕ.....	10
4 ОБЩИЕ ПОЛОЖЕНИЯ.....	11
5 ТЕХНИЧЕСКИЕ ТРЕБОВАНИЯ.....	12
5.1 СЕТЕВОЕ ВЗАИМОДЕЙСТВИЕ.....	12
5.2 АВТОРИЗАЦИЯ.....	12
5.2.1 ЭЛЕКТРОННАЯ ПОДПИСЬ.....	12
5.2.2 СЕРТИФИКАТ.....	13
6 СОЗДАНИЕ API.....	14
7 ИНТЕГРАЦИЯ В СИСТЕМУ ПАРТНЕРА.....	17
7.1 ПЛАТЕЖИ.....	17
7.1.1 ОБЩАЯ ИНФОРМАЦИЯ.....	17
7.1.2 СОЗДАНИЕ ОПЕРАЦИИ ПЛАТЕЖА.....	20
7.1.3 ПАРАМЕТРЫ ПЛАТЕЖЕЙ.....	25

7.1.3.1 ПАРАМЕТРЫ ДЛЯ ОПЛАТЫ ЧЕРЕЗ СБП.....	25
7.1.3.2 ПАРАМЕТРЫ ДЛЯ ОПЛАТЫ ЧЕРЕЗ СИСТЕМУ SBERPAY.....	26
7.1.3.3 ПАРАМЕТРЫ ДЛЯ ОПЛАТЫ ЧЕРЕЗ СИСТЕМУ НАЛЫК QR.....	27
7.1.3.4 ПАРАМЕТРЫ ДЛЯ ОПЛАТЫ ЧЕРЕЗ СИСТЕМУ ELQR.....	27
7.1.3.5 ПАРАМЕТРЫ ДЛЯ ОПЛАТЫ ЧЕРЕЗ СИСТЕМУ MOBILE MONEY.....	28
7.1.3.6 ПАРАМЕТРЫ ДЛЯ ОПЛАТЫ БАНКОВСКОЙ КАРТОЙ С ВВОДОМ ДАННЫХ НА СТОРОНЕ БАНКА.....	29
7.1.3.7 ПАРАМЕТРЫ ДЛЯ ОПЛАТЫ БАНКОВСКОЙ КАРТОЙ С ВВОДОМ ДАННЫХ НА СТОРОНЕ МЕРЧАНТА.....	30
7.2 ВЫПЛАТЫ.....	41
7.2.1 ОБЩАЯ ИНФОРМАЦИЯ.....	41
7.2.2 СОЗДАНИЕ ОПЕРАЦИИ ВЫПЛАТЫ.....	42
7.2.3 ПАРАМЕТРЫ ОПЕРАЦИИ ВЫПЛАТЫ.....	46
7.2.3.1 ПАРАМЕТРЫ ДЛЯ ВЫПЛАТЫ НА КАРТУ.....	46
7.2.3.2 ПАРАМЕТРЫ ДЛЯ ВЫПЛАТЫ ЧЕРЕЗ СИСТЕМУ MOBILE MONEY.....	46
7.3 СТАТУС ОПЕРАЦИИ.....	47
7.4 ВОЗВРАТ.....	49
7.4.1 ОБЩАЯ ИНФОРМАЦИЯ.....	49
7.4.2 СОЗДАНИЕ ОПЕРАЦИИ ВОЗВРАТА.....	50
7.4.3 ЗАПРОС СТАТУСА ОПЕРАЦИИ ВОЗВРАТА.....	52
7.5 РАСЧЕТ КОМИССИИ.....	53
7.6 БАЛАНС МЕРЧАНТА.....	56
7.7 УВЕДОМЛЕНИЕ ПАРТНЕРА О СТАТУСЕ ЗАКАЗА.....	57
8 ПРИЛОЖЕНИЕ.....	63
A. ПРИМЕРЫ ГЕНЕРАЦИИ И ПРОВЕРКИ ЭП НА JAVA.....	63
B. ПРИМЕРЫ ЗАПРОСА И ОТВЕТОВ ДЛЯ ОПЕРАЦИИ ПЛАТЕЖА.....	65
C. ПРИМЕРЫ ЗАПРОСА И ОТВЕТОВ ДЛЯ ОПЕРАЦИИ ВЫПЛАТЫ.....	80
D. ПРИМЕРЫ ЗАПРОСА И ОТВЕТОВ ДЛЯ ОПЕРАЦИИ ВОЗВРАТА.....	81
E. ПРИМЕРЫ ЗАПРОСА И ОТВЕТОВ ДЛЯ МЕТОДА ЗАПРОСА СТАТУСА ОПЕРАЦИИ...82	82
F. ПРИМЕР ЗАПРОСА И ОТВЕТА ДЛЯ МЕТОДА РАСЧЕТА КОМИССИИ.....	83
G. ПРИМЕР ОТВЕТА НА ЗАПРОС БАЛАНСА МЕРЧАНТА.....	83
H. ПРИМЕР ЗАПРОСА ОТПРАВКИ WEBHOOK-ОПОВЕЩЕНИЯ.....	84
I. СТАТУСЫ ОПЕРАЦИЙ.....	85
J. ПРИМЕРЫ ФОРМ ДЛЯ РЕДИРЕКТА ДЛЯ АВТОРИЗАЦИИ ПО ПРОТОКОЛУ 3DS.....	86

ИСТОРИЯ ИЗМЕНЕНИЙ

ИЗМЕНЕНИЯ В ВЕРСИИ ДОКУМЕНТА 1.0

Дата публикации: 23.05.2025 г.

Изменение	Раздел
Общие улучшения:	
Документ создан	-

ИЗМЕНЕНИЯ В ВЕРСИИ ДОКУМЕНТА 2.0

Дата публикации: 18.08.2025.

Изменение	Раздел
Общие улучшения:	
Документ обновлен и дополнен описанием новых операций и методов	-

ИЗМЕНЕНИЯ В ВЕРСИИ ДОКУМЕНТА 2.1

Дата публикации: 20.10.2025.

Изменение	Раздел
Общие улучшения:	
Добавлено описание основного атрибута платежа (pma)	7.1.2, 7.1.3.1-5

ИЗМЕНЕНИЯ В ВЕРСИИ ДОКУМЕНТА 2.2

Дата публикации: 07.11.2025 г.

Изменение	Раздел
Общие улучшения:	
Обновлено описание создания API	6
Добавлено описание параметров для работы с источником оплаты HALYK QR	7.1.3.3

ИЗМЕНЕНИЯ В ВЕРСИИ ДОКУМЕНТА 2.3

Дата публикации: 10.11.2025 г.

Изменение	Раздел
Общие улучшения:	
Обновлено название документа	-

ИЗМЕНЕНИЯ В ВЕРСИИ ДОКУМЕНТА 2.4

Дата публикации: 26.11.2025 г.

Изменение	Раздел
Общие улучшения:	
Добавлено описание параметров для работы с источником оплаты ELQR	7.1.3.4

ИЗМЕНЕНИЯ В ВЕРСИИ ДОКУМЕНТА 2.5

Дата публикации: 03.12.2025 г.

Изменение	Раздел
Общие улучшения:	
Общие улучшения документа	-

ИЗМЕНЕНИЯ В ВЕРСИИ ДОКУМЕНТА 2.6

Дата публикации: 16.02.2026 г.

Изменение	Раздел
Общие улучшения:	
Добавлено описание метода запроса баланса мерчанта	7.6
Добавлено описание метода расчета комиссии	7.5

1 ТЕРМИНЫ И СОКРАЩЕНИЯ

Мерчант — юридическое лицо, занимающееся продажей товаров или предоставлением услуг и принимающее оплату через электронные платежные системы.

ПЦ — процессинговый центр.

СБП — система быстрых платежей.

ЭП — электронная подпись.

URL (англ. Uniform Resource Locator) — стандартизированный способ записи адреса ресурса в сети Интернет.

2 ПРЕДПОЛАГАЕМАЯ АУДИТОРИЯ

Данное руководство предназначено для использования в качестве спецификации для настройки и интеграции коннектора Paylogic в систему мерчанта.

Документ адресован IT-специалистам (программистам и администраторам), обслуживающим информационную систему поставщика услуг и знакомым с технологиями разработки ПО, языками программирования, основами работы с XML и сетевым взаимодействием по HTTP/HTTPS, владеющим основами криптографии.

Необходимо знание языков программирования.

3 ВВЕДЕНИЕ

Документ предназначен для систем мерчантов и их разработчиков:

1. Для проведения доработки (разработки) программного обеспечения для систем мерчантов, разработки технических заданий на доработку (разработку) соответствующих систем, обеспечивающих возможность подключения к системам мерчантов.
2. Для оценки качества проведенных доработок и соответствия доработанной системы общим техническим требованиям во время комплексных испытаний. Реализация требований данного документа позволит произвести подключение к системе приема платежей и обеспечить проведение операций с денежными средствами.

4 ОБЩИЕ ПОЛОЖЕНИЯ

Система предназначена для обработки онлайн-транзакций платежей и выплат с использованием различных вариантов движения денежных средств.

Бизнес-цель создания системы — расширить систему платежей и выплат, а так же увеличить перечень платежных инструментов, с помощью которых клиент может совершать транзакции.

5 ТЕХНИЧЕСКИЕ ТРЕБОВАНИЯ

5.1 СЕТЕВОЕ ВЗАИМОДЕЙСТВИЕ

Сетевое взаимодействие системы мерчанта и ПЦ должно происходить в рамках протокола HTTPS. Для взаимодействия с системой мерчанта ПЦ предоставляет адрес, на который будут отправляться запросы. Запросы отправляются методом POST.

При ответе на полученный пакет ПЦ возвращает пакет с подписью, в котором содержатся результаты обработки данных пакета. Данные должны помещаться в теле страницы, генерируемой в ответ на обращение со стороны системы мерчанта.

Сервер ПЦ имеет возможность обрабатывать параллельно несколько запросов.

5.2 АВТОРИЗАЦИЯ

Авторизация на сервере ПЦ может происходить двумя способами:

1. С использованием электронной подписи;
2. С использованием сертификата.

Способ авторизации устанавливается при создании API, путем выбора типа API.

5.2.1 ЭЛЕКТРОННАЯ ПОДПИСЬ

Тип API «Host to Host sign»

Для авторизации запросов в системе используется проверка электронной подписи, для этого система ожидает от мерчанта передачу специальных HTTP-заголовков:

- **PSP-Point** — идентификатор API мерчанта, присвоенный в системе Paylogic;
- **PSP-Sign** — подпись запроса.

Ключи для формирования подписи генерируются на сервере ПЦ при создании API с типом «Host to Host sign» (создание API и ключей для формирования подписи описано в разделе [6](#)).

Электронная подпись формируется от строки <REQUEST METHOD>+<REQUEST FULL URI>+<Request Body> по алгоритму SHA256+RSA в формате Base64 (UTF8).

Пример подписываемой строки:

```
POST/psp/external/api/payment/  
process{"checkNumber":10023,"currency":"RUB","date":"2025-05-  
16T09:38:33.589Z","feeSum":0,"id":"123456789","purchaseSum":1,"service":5  
,"source":{"code":"SBP"}}
```

Получившаяся подпись будет иметь вид:

```
vhF2ZzOYR1tdnASpbd1tCXCrPh3IdLo2vhB133fkvEX7vNSkVvkT2G+sqf2jMAUTQ7c/T2s/  
ppuz/7aLXZWAQbAPYIatgRXIwXSJW8y8SO99h4jQrenVly1abPLSgC3SK0scmhBwkMGA/iI/  
i/5S+q+tIfDcnbCR1tI1We9Gi3A=
```

Пример генерации и проверки подписи приведен в приложении [А. Примеры генерации и проверки ЭП на java](#).

5.2.2 СЕРТИФИКАТ

Тип API «Host to Host cert»

Для авторизации мерчанта в системе используется сертификат. Каждый запрос на сервер ПЦ должен происходить с использованием сертификата, полученного при создании API с типом «Host to Host cert» (создание API и ключей для формирования подписи описано в разделе [6](#)).

6 СОЗДАНИЕ API

Для создания запросов на проведение операций необходимо создать в личном кабинете мерчанта API с типом «Host to Host sign» или «Host to Host cert» (раздел «API — Список API») (рисунок 6.1).

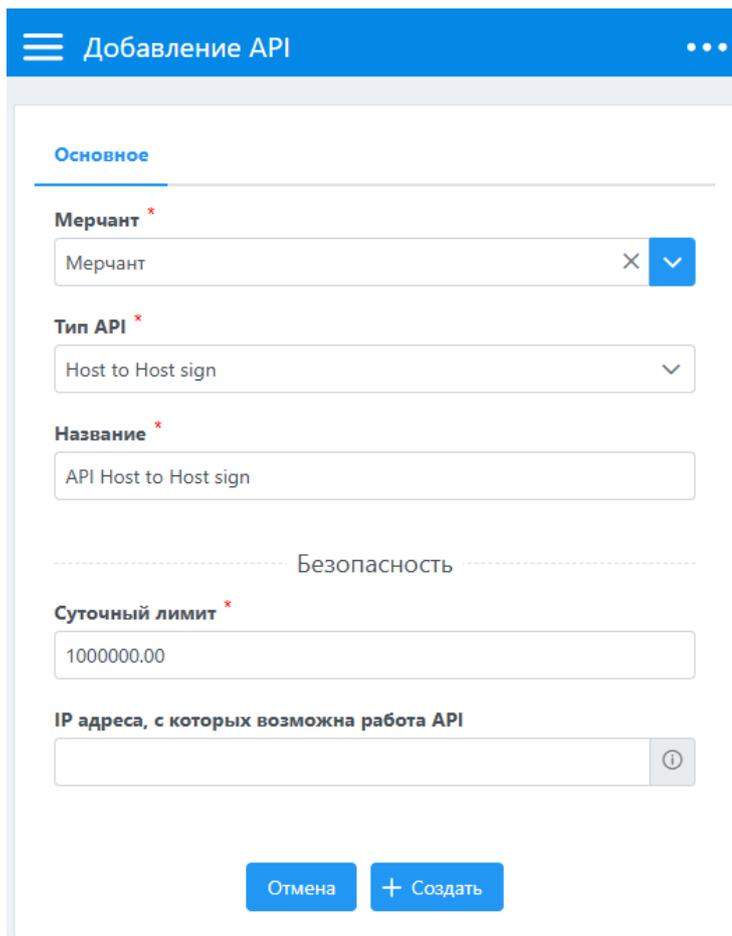


Рисунок 6.1 — Создание API с типом «Host to Host sign»

1. Для создания API заполните следующие поля:

- **Мерчант** — мерчант, которому принадлежит API;
- **Тип API** — «Host to Host sign» или «Host to Host cert»;

- **Название** — будет использоваться в отчетах и выпадающих списках в личном кабинете;
 - **Суточный лимит** — максимальная сумма оплат в сутки через API;
 - **IP-адреса, с которых возможна работа API** — IP-адреса-мерчанта, с которых возможно будет передавать запросы. При указании, работа с других IP-адресов будет невозможна, IP указываются через запятую.
2. При успешном создании API будет сформировано уведомление о необходимости генерации ключей или создании сертификата, в зависимости от вида создаваемого API. Ключи или сертификат можно сгенерировать сразу, либо позже на вкладке «Безопасность». Набор ключей, открытая и закрытая часть, будет доступен для скачивания на этой же вкладке (рисунок 6.3). Сертификат для скачивания и пароль к нему так же будут доступны на вкладке «Безопасность» (рисунок 6.2).

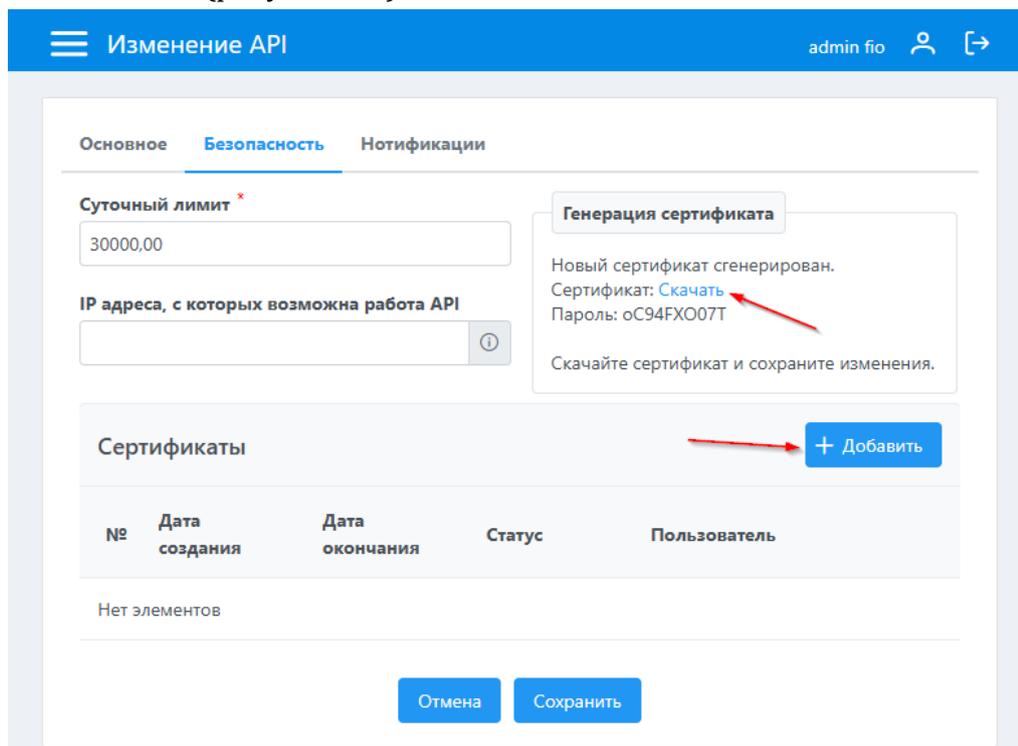


Рисунок 6.2 — Вкладка «Безопасность» для API типа «Host to Host cert»

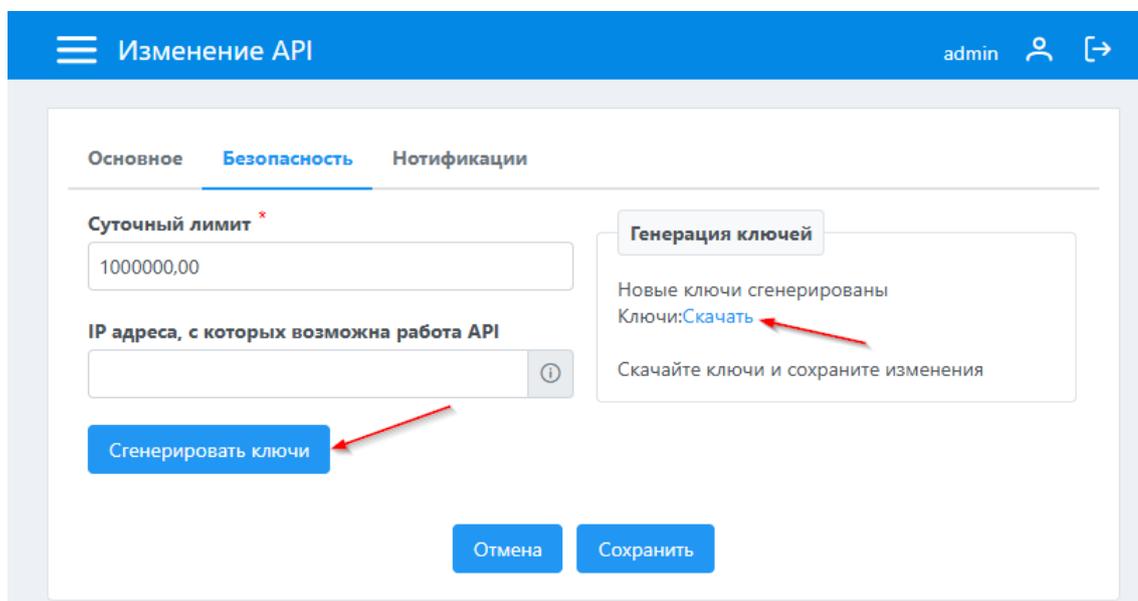


Рисунок 6.3 — Вкладка «Безопасность» для API типа «Host to Host sign»

7 ИНТЕГРАЦИЯ В СИСТЕМУ ПАРТНЕРА

7.1 ПЛАТЕЖИ

7.1.1 ОБЩАЯ ИНФОРМАЦИЯ

Схема взаимодействия участников приведена на рисунке 7.1.1.

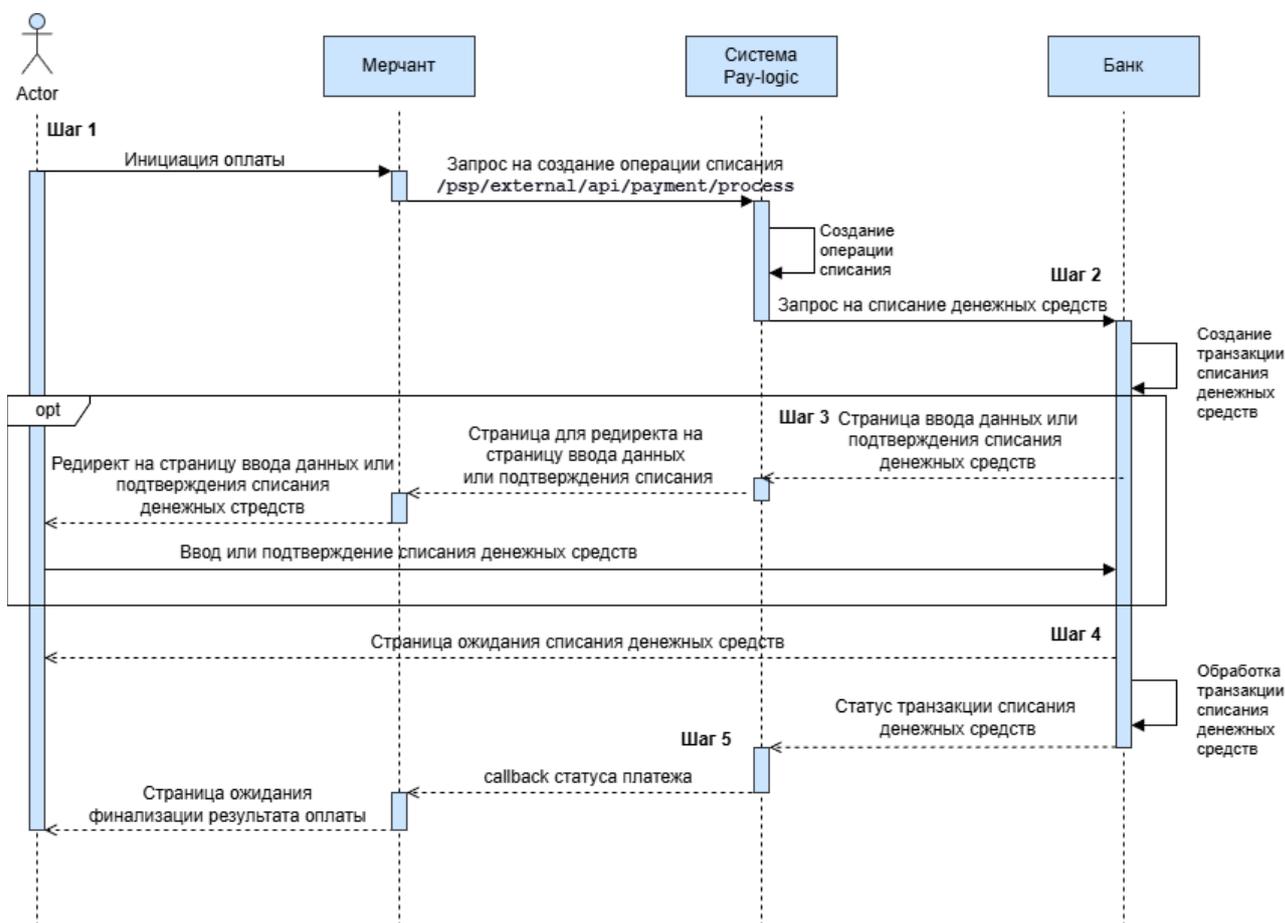


Рисунок 7.1.1 — Диаграмма взаимодействия участников процесса списания денежных средств

-
- Шаг 1:** Для создания операции списания денежных средств мерчант вызывает POST-запрос `/psp/external/api/payment/process`. При вызове данного метода операция списания будет зарегистрирована в системе.
- Шаг 2:** Система PSP отправляет запрос на списание денежных средств банку. Банк создает транзакцию списания денежных средств.
- Шаг 3:** Опционально банк может запросить подтверждение списания денежных средств и отправит системе страницу для подтверждения. Система PSP отправляет мерчанту страницу для редиректа клиента. Мерчант производит редирект клиента на страницу для подтверждения списания денежных средств. Клиент производит подтверждение списания денежных средств.
- Шаг 4:** Банк производит обработку транзакции списания денежных средств и возвращает ее результат PSP.
- Шаг 5:** PSP отправляет callback-нотификацию статуса платежа в систему мерчанта. В соответствии с полученным результатом статуса операции списания денежных средств мерчант производит редирект клиента на страницу результата оплаты (Успех, Ошибка, Обрабатывается).

Дополнительные возможности:

- При необходимости мерчантом может быть дополнительно запрошен статус заказа вызовом метода `/psp/external/api/payment/status`. Метод вернет в ответ текущий статус заказа (описание метода и его параметров приведено в разделе [7.3](#)). Рекомендуется запросить статус если автоматически не был получен ответ в течении 5 минут.
- В системе предусмотрена защита от дубликатов операций списания, проверка на дубликаты производится по уникальному идентификатору операции мерчанта. При повторном вызове метода создания операции списания по уже зарегистрированной в системе операции списания с одинаковым значением параметра `id`, метод вернет в ответ результате обработки заказа и его текущий статус.

Жизненный цикл операции платежа представлен на рисунке 7.1.2:

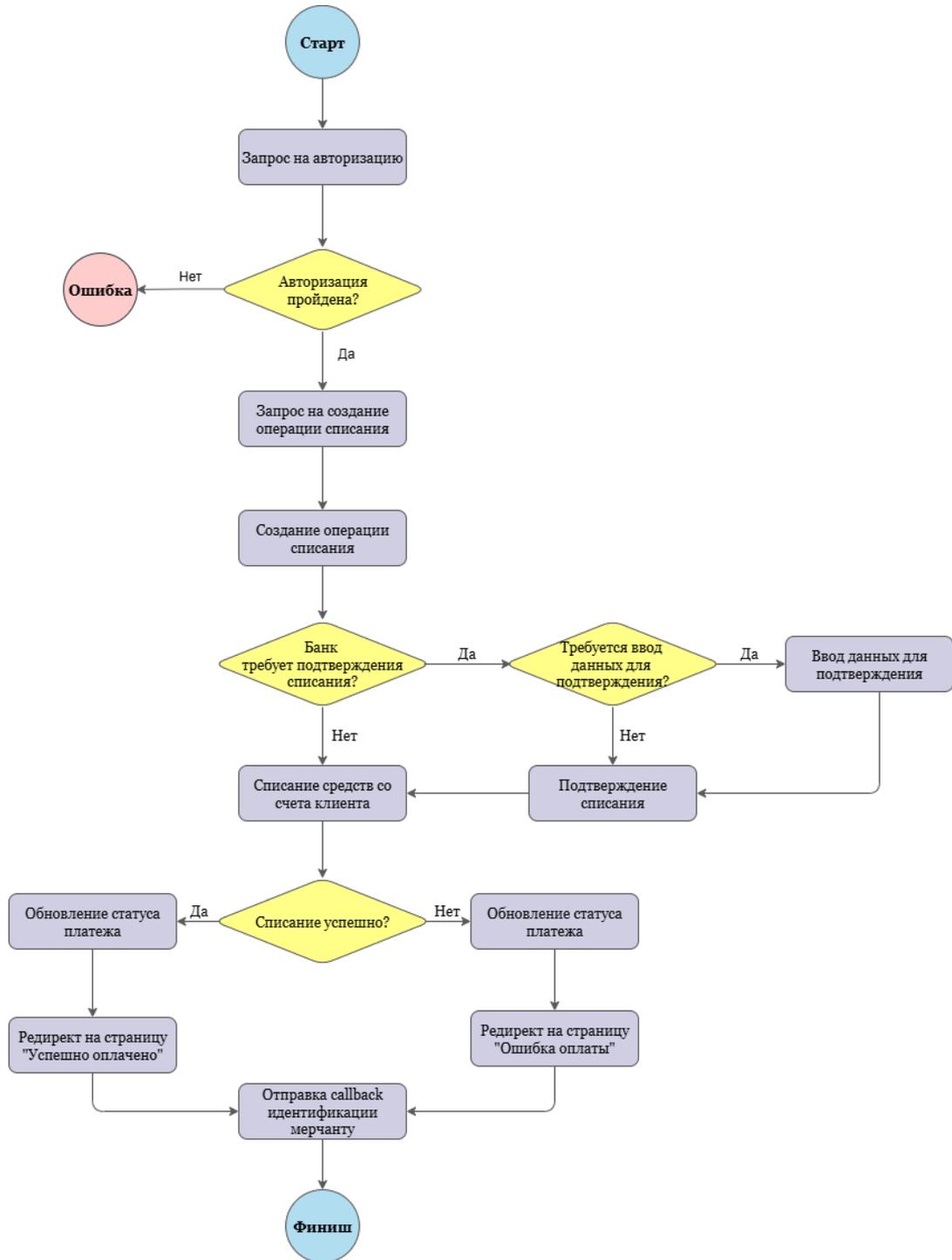


Рисунок 7.1.2 — Схема жизненного цикла операции платежа

7.1.2 СОЗДАНИЕ ОПЕРАЦИИ ПЛАТЕЖА

POST метод `/psp/external/api/payment/process` предназначен для создания операции платежа.

Структура запроса:

```
{
  "checkNumber": 0,
  "currency": "string",
  "date": "2025-05-22T02:38:00.752Z",
  "feeSum": 0,
  "id": "string",
  "idInvoice": 0,
  "params": [
    {
      "code": "string",
      "value": "string"
    }
  ],
  "purchaseSum": 0,
  "service": 0,
  "source": {
    "code": "string",
    "params": [
      {
        "code": "string",
        "value": "string"
      }
    ]
  }
}
```

Таблица 7.1.2.1 — Атрибуты запроса на создание операции платежа

Название	Описание	Обяз.
checkNumber	Номер операции, сообщается клиенту. Если параметр не используется на стороне мерчанта необходимо	Да

	передавать «0»	
currency	Код валюты транзакции в формате ISO 4217	Да
date	Дата и время создания операции, в формате уууу-ММ-dd'T'HH:mm:ss	Да
feeSum	Сумма комиссии, взимаемая с клиента на стороне мерчанта, в минорной единице валюты. Не используется при расчетах или списаниях на стороне PSP	Да
id	Уникальный идентификатор транзакции на стороне мерчанта	Да
purchaseSum	Сумма списания со счета клиента, в минорной единице валюты	Да
params []	Объект со списком дополнительных параметров транзакции. Обязательно должен содержать параметр id1 — идентификатор клиента (id клиента в системе партнера или номер телефона клиента или e-mail клиента)	Да
params [].code	Код параметра транзакции	Нет
params [].value	Значение параметра транзакции	Нет
service	Идентификатор сервиса на стороне ПЦ. Предоставляется мерчанту при интеграции системы	Нет
source []	Объект со списком дополнительных параметров	Нет

	транзакции необходимых для обработки операции списания.	
source[] .code	Код источника оплаты. Предоставляется мерчанту при интеграции системы	Нет
source[] .params	Список параметров, необходимых для создания банковской операции. Список параметров и их формат зависит от способа оплаты, см. раздел <u>7.1.3</u>	Нет

Структура ответа:

```
{
  "attributes": [
    {
      "code": "string",
      "flags": 0,
      "title": "string",
      "value": "string",
      "valueTitle": "string"
    }
  ],
  "bankPayment": {
    "attributes": [
      {
        "code": "string",
        "flags": 0,
        "title": "string",
        "value": "string",
        "valueTitle": "string"
      }
    ],
    "error": 0,
    "errorMessage": "string",
    "status": "ERROR",
    "transaction": "string"
  },
  "date": "string",
```

```
"error": 0,  
"errorDetail": {  
  "errorFields": [  
    "string"  
  ],  
  "message": "string",  
  "type": "COMMISSION_ERROR"  
},  
"id": "string",  
"menu": {  
  "category": "string",  
  "service": 0,  
  "serviceName": "string"  
},  
"message": "string",  
"status": "ALREADY_EXISTS",  
"transaction": 0  
}
```

Таблица 7.1.2.2 — Атрибуты ответа на запрос о создании платежа

Параметр	Описание	Обяз.
id	Уникальный идентификатор транзакции мерчанта	Да
status	Статус операции на стороне PSP. Возможные статусы приведены в таблице I.1	Да
bankPayment. status	Статус банковской операции. Возможные статусы приведены в таблице I.1	Да
bankPayment. transaction	Уникальный идентификатор транзакции на стороне банка	Да

<code>bankPayment.attributes[]</code>	Объект параметров банковской операции	Да
<code>bankPayment.attributes[].order-id</code>	Уникальный идентификатор заказа на стороне банка. Может дублировать значение, передаваемое в параметре <code>bankPayment.transaction</code> для некоторых банков	Да
<code>bankPayment.attributes[].payload</code>	URL-ссылка для подтверждения списания денежных средств. Мерчант должен сделать редирект клиента на полученную ссылку	Нет
<code>transaction</code>	Уникальный идентификатор транзакции на стороне ПЦ	Нет
<code>error</code>	Код ошибки	Нет
<code>errorDetail</code>	Детализирующий код ошибки	Нет
<code>errorMessage</code>	Описание ошибки	Нет

Таблица 7.1.2.3 — Значения параметра `error`

Код	Описание
0	Ошибок нет, запрос обработан успешно. Необходимо анализировать значение поля <code>status</code>
!=0	Ошибка выполнения запроса, требуется обратиться к сотруднику системы для локализации причины

Примеры запроса и ответов приведены в приложении [В. Примеры запроса и ответов для операции платежа.](#)

7.1.3 ПАРАМЕТРЫ ПЛАТЕЖЕЙ

Параметры платежей передаются в объекте `source[] .params`, они могут отличаться в зависимости от источника оплаты.

7.1.3.1 ПАРАМЕТРЫ ДЛЯ ОПЛАТЫ ЧЕРЕЗ СБП

При оплате через СБП параметр `source[] .code` имеет значение **SBP**

Таблица 7.1.3.1.1 — Параметры, передаваемые в запросе при оплате через СБП

Параметр	Описание	Обяз.
<code>pma</code>	Идентификатор клиента	Да
<code>client- ipAddress</code>	IP адрес клиента (ipv4 / ipv6)	Да
<code>email</code>	E-mail адрес клиента	Да
<code>redirectUrlSu ccess</code>	URL-адрес для возврата из приложения банка-эмитента после успешной оплаты	Нет
<code>redirectUrlEr ror</code>	URL-адрес для возврата из приложения эмитента после неуспешной оплаты	Нет
<code>merch-url</code>	URL сайта мерчанта	Нет

Дополнительные параметры необходимо уточнить у менеджера проекта.

7.1.3.2 ПАРАМЕТРЫ ДЛЯ ОПЛАТЫ ЧЕРЕЗ СИСТЕМУ SBERPAY

При оплате через систему SberPay параметр **source [] . code** имеет значение **SBERPAY**.

Таблица 7.1.3.2.1 — Параметры, передаваемые в запросе при оплате через систему SberPay.

Параметр	Описание	Обяз.
num	Номер телефона клиента, в формате 79600000000	Да
serviceCode	Код сервиса. Необходимо уточнить у менеджера проекта	Да
frontType	Атрибут, указывающий на тип завершения способа оплаты. Может принимать значения: <ul style="list-style-type: none">• back2app — для завершения оплаты будет отправлено push или sms-уведомление на указанный в запросе номер телефона;• app2app — для завершения оплаты в ответе вернется параметр sbol-deep-link, содержащий URL-ссылку на приложение «Сбербанк Онлайн»	Нет
osType	Тип операционной системы клиента. Параметр обязателен только для frontType=app2app . Может принимать значения: <ul style="list-style-type: none">• ios;• android	Нет
deepLink	URL-ссылка для возврата в приложение мерчанта. Параметр обязателен только для	Нет

	frontType=app2app	
redirectUrlError	URL-ссылка для редиректа клиента в случае ошибочного проведения операции	Нет
redirectUrlSuccess	URL-ссылка для редиректа клиента в случае успешного проведения операции	Нет

7.1.3.3 ПАРАМЕТРЫ ДЛЯ ОПЛАТЫ ЧЕРЕЗ СИСТЕМУ HALYK QR

При оплате через систему HALYK QR параметр **source[].code** имеет значение **HALYK_QR**.

Таблица 7.1.3.3.1 — Параметры, передаваемые в запросе при оплате через HALYK_QR

Параметр	Описание	Обяз.
pnma	Номер телефона клиента	Да
redirectUrlSuccess	URL-адрес для возврата из приложения после успешной оплаты	Нет
redirectUrlError	URL-адрес для возврата из приложения после неуспешной оплаты	Нет

7.1.3.4 ПАРАМЕТРЫ ДЛЯ ОПЛАТЫ ЧЕРЕЗ СИСТЕМУ ELQR

При оплате через систему ELQR параметр **source [].code** имеет значение **ELQR**.

Таблица 7.1.3.4.1 — Параметры, передаваемые в запросе при оплате через ELQR

Параметр	Описание	Обяз.
partner	Идентификатор клиента (id клиента в системе партнера или номер телефона клиента или e-mail клиента)	Да

7.1.3.5 ПАРАМЕТРЫ ДЛЯ ОПЛАТЫ ЧЕРЕЗ СИСТЕМУ MOBILE MONEY

При оплате через систему Mobile Money параметр **source[].code** имеет значение **MOBILE_MONEY**.

Таблица 7.1.3.5.1 — Параметры, передаваемые в запросе при оплате через систему Mobile Money.

Параметр	Описание	Обяз.
partner	Номер телефона клиента, в формате: <ul style="list-style-type: none">• КОТ-Д'ИВУАР — 225000000000;• СЕНЕГАЛ — 221700000000;• УГАНДА — 256709000000;• БЕНИН — 229000000000;• МАЛИ — 223000000000	Да
customerName	Имя клиента	Да

customerEmail	Электронная почта клиента	Да
redirectUrlProcess	URL-ссылка для редиректа клиента в случае получения нефинального статуса операции	Да
country	Код страны в формате ISO 3166-1 alpha-2	Да
providerId	Идентификатор оператора	Да
redirectUrlSuccess	URL-ссылка для редиректа клиента в случае успешного проведения операции	Нет
redirectUrlError	URL-ссылка для редиректа клиента в случае ошибочного проведения операции	Нет

7.1.3.6 ПАРАМЕТРЫ ДЛЯ ОПЛАТЫ БАНКОВСКОЙ КАРТОЙ С ВВОДОМ ДАННЫХ НА СТОРОНЕ БАНКА

При оплате банковской картой параметр **source[] . code** имеет значение **BANKCARD-PS**

Таблица 7.1.3.6.1 — Параметры, передаваемые в запросе при оплате банковской картой, с вводом данных на стороне банка

Параметр	Описание	Обяз.
client-ipAddress	IP адрес клиента (ipv4 / ipv6)	Да

redirectUrlSuccess	URL для возврата из приложения банка-эмитента после успешной оплаты	Нет
redirectUrlError	URL для возврата из приложения эмитента после не успешной оплаты	Нет
email	E-mail адрес клиента	Нет
merch-url	URL сайта мерчанта	Нет

Дополнительные параметры необходимо уточнить у менеджера проекта.

7.1.3.7 ПАРАМЕТРЫ ДЛЯ ОПЛАТЫ БАНКОВСКОЙ КАРТОЙ С ВВОДОМ ДАННЫХ НА СТОРОНЕ МЕРЧАНТА

При оплате банковской картой параметр **source [] . code** имеет значение **BANKCARD-PS**

Таблица 7.1.3.7.1 — Параметры, передаваемые в запросе при оплате банковской картой, с вводом данных на стороне мерчанта

Параметр	Описание	Обяз.
client-ipAddress	IP адрес клиента (ipv4 / ipv6)	Да
email	E-mail адрес клиента	Да
client-browserInfo	Строка в формате base64, содержащая json информацию о браузере клиента согласно: EMVCo_3DS_Spec_v220_122018/threeDSRequestorAut	Да

	henticationInfo	
client-agentName	Имя приложения клиента	Да
pan	Номер банковской карты	Да
expiry	Срок действия банковской карты	Да
cvv	cvv код банковской карты	Да
holder	Держатель банковской карты	Да
redirectUrlSuccess	URL для возврата из приложения банка-эмитента после успешной оплаты	Нет
redirectUrlError	URL для возврата из приложения эмитента после неуспешной оплаты	Нет
merch-url	URL-сайта мерчанта	Нет

Дополнительные параметры необходимо уточнить у менеджера проекта.

При оплате банком-эмитентом может быть затребовано подтверждение операции списания по протоколу 3DS.

Для определения версии протокола 3DS системой анализируются вернувшиеся в ответе параметр **paReq**:

- если в ответе вернулся параметр **paReq**, то будет использоваться протокол **3DSv1**;

-
- если в ответе параметр **paReq** отсутствует, то будет использоваться протокол **3DSv2**.

Схема определения версии протокола 3DS и подтверждения платежа представлена на рисунке 7.1.3.

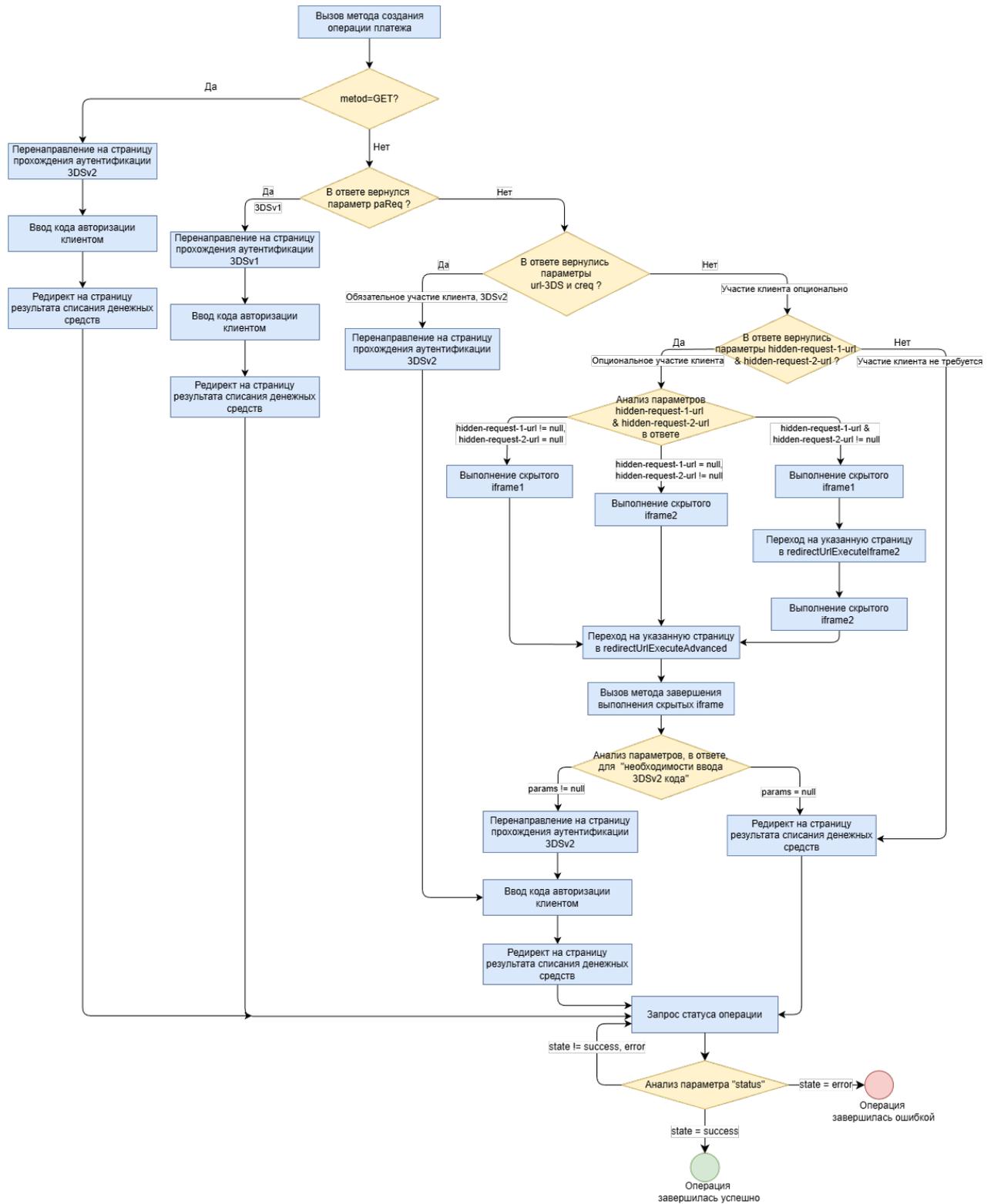


Рисунок 7.1.3 — Схема определения версии протокола и прохождения авторизации по 3DS

ПОДТВЕРЖДЕНИЕ ОПЕРАЦИИ ПО ПРОТОКОЛУ ВЕРСИИ 3DSV1

После получения ответа на запрос создания операции списания мерчанту необходимо перенаправить клиента на страницу прохождения аутентификации 3DSv1. Для этого мерчанту необходимо вызвать форму для редиректа, для формирования которой необходимо заполнить параметры, описанные в таблице 7.1.3.7.2 (пример формы в формате html приведен в приложении I. Примеры форм для редиректа для авторизации по протоколу 3DS).

Таблица 7.1.3.7.2 — Параметры для прохождения аутентификации по версии 3DSv1

Параметр	Описание	Обяз.
action	Значение параметра url-3DS из ответа на запрос	Да
PaReq	Значение параметра paReq из ответа на запрос	Да
TermUrl	Значение параметра redirect-url-3DS из ответа на запрос	Нет
MD	Значение параметра md из ответа на запрос	Нет

После вызова формы клиент будет перенаправлен на страницу прохождения 3DSv1-аутентификации, на которой вводит код из SMS-сообщения, полученного от банка. Затем производится редирект клиента на страницу результата списания денежных средств.

ПОДТВЕРЖДЕНИЕ ОПЕРАЦИИ ПО ПРОТОКОЛУ ВЕРСИИ 3DSV2

При прохождении 3DS-аутентификации по протоколу 3DSv2 дополнительно анализируются параметры **url-3DS** и **creq**, полученные в ответе:

- если оба параметра **url-3DS** и **creq** присутствуют в ответе, то требуется обязательное участие клиента для прохождения аутентификации по протоколу 3DSv2;
- если параметры **url-3DS** и **creq** отсутствуют в ответе, то участие клиента для прохождения аутентификации по протоколу 3DSv2 опционально и может не требоваться.

ОБЯЗАТЕЛЬНОЕ УЧАСТИЕ КЛИЕНТА ДЛЯ ПРОХОЖДЕНИЯ 3DS-АУТЕНТИФИКАЦИИ

После получения ответа на запрос создания операции списания мерчанту необходимо перенаправить клиента на страницу прохождения аутентификации 3DSv2. Для этого мерчанту необходимо вызвать форму для редиректа, для формирования которой необходимо заполнить параметры, описанные в таблице 7.1.3.7.3 (пример формы в формате html приведен в приложении J. Примеры форм для редиректа для авторизации по протоколу 3DS).

Таблица 7.1.3.7.3 — Параметры для прохождения аутентификации по версии 3DSv2

Параметр	Описание	Обяз.
action	Значение параметра url-3DS из ответа на запрос	Да
creq	Значение параметра creq из ответа на запрос	Да
gatewaySessionId	Значение параметра gatewaySessionId из ответа на запрос	Нет
threeDSSessionData	Значение параметра threeDSSessionData из ответа на запрос	Нет

После вызова формы (форма должна быть вызвана в течении 10 секунд после получения ответа), клиент будет перенаправлен на страницу прохождения 3DSv2-

аутентификации, на которой вводит код из SMS-сообщения, полученного от банка. Затем производится редирект клиента на страницу результата списания денежных средств.

ОПЦИОНАЛЬНОЕ УЧАСТИЕ КЛИЕНТА ДЛЯ ПРОХОЖДЕНИЯ 3DS-АУТЕНТИФИКАЦИИ

В случае если в ответе не вернулись параметры **url-3DS** и **creq**, в ответе на метод **/psp/external/api/payment/process**, участие клиента может быть опционально. Для определения участия клиента, необходимо дополнительно проанализировать параметры **hidden-request-1-url** и **hidden-request-2-url** и, в зависимости от результатов ответа, дополнительно может потребоваться пройти скрытые **iframe**:

1. Если оба параметра **hidden-request-1-url** и **hidden-request-2-url** в ответе отсутствуют, то участие клиента в прохождении авторизации не требуется. Клиент будет перенаправлен на страницу результата списания денежных средств;
2. Если в ответе вернулся только параметр **hidden-request-1-url**, то клиент будет перенаправлен на выполнение скрытого **iframe1**;
3. Если в ответе вернулся только параметр **hidden-request-2-url**, то клиент будет перенаправлен на выполнение скрытого **iframe2**;
4. Если в ответе вернулись оба параметра **hidden-request-1-url** и **hidden-request-2-url**, то клиент будет перенаправлен мерчантом на выполнение скрытого **iframe1**, затем скрытого **iframe2**.

1. Выполнение скрытого **iframe1**

Для выполнения скрытого **iframe1** мерчанту необходимо вызвать форму, в которой должен быть заполнен параметр **action** — значение из параметра **hidden-request-1-url**, который вернулся в ответе (пример формы для редиректа приведен в приложении I. Примеры форм для редиректа для авторизации по протоколу 3DS).

После выполнения *iframe1* должен быть выполнен редирект на страницу, которая была передана в параметре `redirectUrlExecuteAdvanced`. Для завершения выполнения *iframe2* необходим вызов `advanced` функции `complete-3DS`, см. подраздел Запрос для завершения скрытых iframe.

2. Выполнение скрытого *iframe2*

Для выполнения скрытого *iframe2* мерчанту необходимо вызвать форму, в которой должны быть заполнены параметры (пример формы для редиректа приведен в приложении I. Примеры форм для редиректа для авторизации по протоколу 3DS):

- **action** — значение из параметра **hidden-request-2-url**, который вернулся в ответе на запрос;
- **threeDSMethodData** — значение из параметра **threeDSMethodData**, который вернулся в ответе на запрос.

После выполнения *iframe2* должен быть выполнен редирект на страницу, которая была передана в параметре `redirectUrlExecuteAdvanced`. Для завершения выполнения *iframe2* необходим вызов `advanced` функции `complete-3DS`, см. подраздел Запрос для завершения скрытых iframe.

3. Выполнение скрытого *iframe1* и *iframe2*

Для выполнения скрытого *iframe1* мерчанту необходимо вызвать форму, в которой должен быть заполнен параметр **action** — значение из параметра **hidden-request-1-url**, который вернулся в ответе.

Затем необходимо вызвать форму выполнения скрытого *iframe2*, в которой должны быть заполнены параметры:

- **action** — значение из параметра **hidden-request-2-url**, который вернулся в ответе на запрос;

- **threeDSMethodData** — значение из параметра **threeDSMethodData**, который вернулся в ответе на запрос.

После выполнения *iframe2* должен быть выполнен редирект на страницу, которая была передана в параметре `redirectUrlExecuteAdvanced`. Для завершения выполнения *iframe2* необходим вызов `advanced` функции `complete-3DS`, см. подраздел Запрос для завершения скрытых iframe.

ЗАПРОС ДЛЯ ЗАВЕРШЕНИЯ СКРЫТЫХ IFRAME

POST-метод `/psp/external/api/payment/advanced` предназначен для завершения скрытых iframe.

Структура запроса:

```
{
  "function": "string",
  "id": "string",
  "locale": "string",
  "params": [
    {
      "code": "string",
      "value": "string"
    }
  ],
  "provider": 0,
  "service": 0
}
```

Таблица 7.1.3.7.4— Атрибуты запроса для завершения скрытых iframe

Параметр	Описание	Обяз.
function	Константа, название вызова требуемой функции, в данном случае имеет значение complete-3DS	Да

gateway-session-id	Значение параметра gateway-session-id из ответа на на ВЫЗОВ метода /psp/external/api/payment/process	Да
threeDSMethodData	Значение параметра threeDSMethodData из ответа на ВЫЗОВ метода /psp/external/api/payment/process	Да
bank-operation-id	Значение параметра bank-operation-id из ответа на ВЫЗОВ метода /psp/external/api/payment/process	Да
service	Идентификатор сервиса на стороне ПЦ. Предоставляется мерчанту при интеграции системы	Да

Структура ответа:

```
{
  "data": {
    "attributes": [
      {
        "code": "string",
        "flags": 0,
        "title": "string",
        "value": "string",
        "valueTitle": "string"
      }
    ],
    "feeSum": 0,
    "nested": {
      "data": [
        null
      ],
      "id": "string"
    },
    "sumPurchase": 0
  },
}
```

```

"errorDetail": {
  "errorFields": [
    "string"
  ],
  "message": "string",
  "type": "COMMISSION_ERROR"
},
"status": "NOT_FOUND"
}

```

Таблица 7.1.3.7.5 — Атрибуты ответа на запрос для завершения скрытых iframe

Параметр	Описание	Обяз.
status	Статус обработки запроса. Возможные статусы приведены в таблице I.1	Нет
gateway-session-id	Уникальный идентификатор платежной сессии	Нет
url-3DS	URL-адрес, на который нужно перенаправить клиента для проведения аутентификации 3DS	Нет
threeDSSessionData	Идентификатор сессии 3DSv2	Нет
creq	Тело запроса к ACS серверу эмитента 3DSv2	Нет

Если в ответе на запрос **/psp/external/api/payment/advanced** параметры отсутствуют, то участие клиента не требуется. Мерчанту необходимо произвести редирект на страницу результата платежа.

Если в ответе на запрос **/psp/external/api/payment/advanced** параметры были получены, то мерчант должен произвести редирект клиента на страницу прохождения

подтверждения платежа по 3DSv2. Для этого должна быть вызвана соответствующая форма, в которой заполнены следующие данные:

- **action** — значение из параметра **url-3DS**, который вернулся в ответе;
- **gatewaySessionId** — значение из параметра **gateway-session-id**, который вернулся в ответе;
- **threeDSSessionData** — значение из параметра **threeDSSessionData**, который вернулся в ответе;
- **creq** — значение из параметра **creq**, который вернулся в ответе.

После вызова формы (форма должна быть вызвана в течении 10 секунд после получения ответа) клиент будет перенаправлен на страницу прохождения 3DSv2-аутентификации, на которой вводит код из SMS-сообщения, полученного от банка. Затем производится редирект клиента на страницу результата списания денежных средств.

7.2 ВЫПЛАТЫ

7.2.1 ОБЩАЯ ИНФОРМАЦИЯ

В системе реализована возможность осуществлять выплату денежных средств клиенту, с помощью метода `/psp/external/api/payment/process`. Жизненный цикл процедуры выплаты денежных средств представлен на рисунке 7.2.1. Выплата денежных средств может производиться на разные платежные средства. Для проверки результата выплаты должен быть произведен запрос статуса операции (описание метода запроса статуса операции приведено в разделе [7.3](#)).

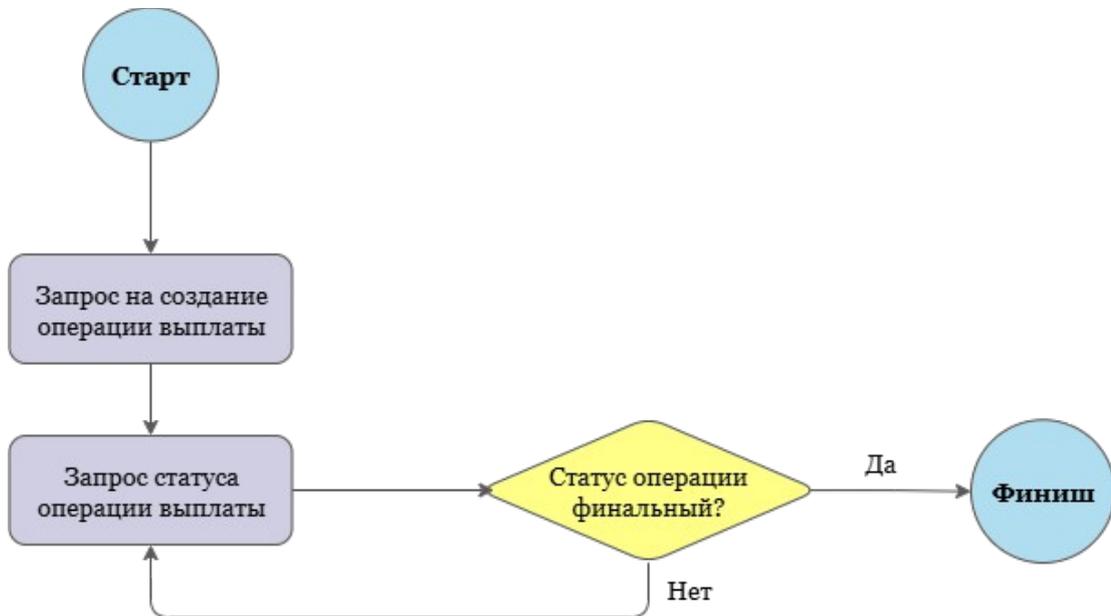


Рисунок 7.2.1 — Схема жизненного цикла операции выплаты

7.2.2 СОЗДАНИЕ ОПЕРАЦИИ ВЫПЛАТЫ

POST метод `/psp/external/api/payment/process` создает операцию выплаты денежных средств.

Структура запроса:

```
{
  "checkNumber": 0,
  "currency": "string",
  "date": "2024-02-28T05:27:18.192Z",
  "feeSum": 0,
  "id": "string",
  "params": [
    {
      "code": "string",
      "value": "string"
    }
  ]
}
```

```
    }  
  ],  
  "purchaseSum": 0,  
  "service": 0,  
  "source": {  
    "code": "string",  
    "params": [  
      {  
        "code": "string",  
        "value": "string"  
      }  
    ]  
  }  
}
```

Таблица 7.2.2.1 — Атрибуты запроса на создание операции выплаты

Параметр	Описание	Обяз.
id	Уникальный идентификатор транзакции на стороне мерчанта	Да
checkNumber	Номер операции, сообщается клиенту. Если параметр не используется на стороне мерчанта необходимо передавать «0»	Да
currency	Код валюты транзакции в формате ISO 4217	Да
date	Дата и время создания операции в формате уууу-ММ dd'T'HH:mm:ss	Да
feeSum	Сумма комиссии, взимаемая с клиента на стороне мерчанта, в минорной единице валюты (например, копейки, центы и т.д.). Не используется при расчетах или списаниях на стороне PSP	Да

params []	Объект со списком дополнительных параметров транзакции. Список параметров и их формат зависит от способа выплаты, см. раздел 7.2.3	Да
purchaseSum	Сумма выплаты, в минорной единице валюты (например, копейки, центы и т.д.)	Да
service	Идентификатор сервиса на стороне ПЦ. Предоставляется мерчанту при интеграции системы	Да
source []	Объект данных, содержащий параметры платежного инструмента	Да
source [] . code	Код источника оплаты, см. раздел 7.2.3	Да

Структура ответа:

```
{
  "id": "pltest117",
  "transaction": 1391241,
  "status": "PROCESSING",
  "errorDetail": null,
  "error": 0,
  "message": null,
  "bankPayment": null,
  "date": "2024-02-28T03:37:44.354+0000",
  "attributes": null,
  "menu": null
}
```

Таблица 7.2.2.2 — Атрибуты ответа на запрос создания операции выплаты

Параметр	Описание	Обяз.
id	Уникальный идентификатор транзакции на стороне	Да

	мерчанта	
transaction	Уникальный идентификатор транзакции на стороне PSP	Да
status	Статус операции. Возможные статусы приведены в таблице <u>I.1</u>	Да
errorDetail	Детализирующий код ошибки	Нет
error	Код ошибки	Нет
message	Описание ошибки	Нет
bankPayment	Объект данных о банковской транзакции. Описание объекта приведено в таблице <u>7.7.2</u>	Нет
date	Дата и время операции в формате уууу-ММ dd'T'HH:mm:ss	Да
attributes	Объект данных атрибутов основной операции. Описание объекта приведено в таблице <u>7.7.3</u>	Нет
menu	Объект данных сервиса, по которому осуществлялась оплата. Описание объекта приведено в таблице <u>7.7.4</u>	Нет

Примеры запроса и ответа приведены в приложении C. Примеры запроса и ответов для операции выплаты.

7.2.3 ПАРАМЕТРЫ ОПЕРАЦИИ ВЫПЛАТЫ

7.2.3.1 ПАРАМЕТРЫ ДЛЯ ВЫПЛАТЫ НА КАРТУ

При выплате денежных средств на банковскую карту параметр `source[] . code` имеет значение **BANKCARD**.

Таблица 7.2.3.1.1 — Параметры, передаваемые в запросе при выплате на банковскую карту

<code>id1</code>	Номер телефона получателя, в формате: +700000000000	Да
<code>pan</code>	PAN карты получателя	Да
<code>customerName</code>	ФИО получателя	Да

7.2.3.2 ПАРАМЕТРЫ ДЛЯ ВЫПЛАТЫ ЧЕРЕЗ СИСТЕМУ MOBILE MONEY

При выплате денежных средств через систему Mobile Money параметр `source[] . code` имеет значение **CASH**.

Таблица 7.2.3.2.1 — Параметры, передаваемые в запросе при выплате через систему MOBILE MONEY

<code>params[] . id1</code>	Номер телефона получателя, в формате: <ul style="list-style-type: none">• КОТ-Д'ИВУАР — 2250000000000;• СЕНЕГАЛ — 2217000000000;	Да
-----------------------------	---	----

	<ul style="list-style-type: none">• УГАНДА — 256709000000;• БЕНИН — 2290000000000;• МАЛИ — 223000000000	
params []. customerName	ФИО получателя	Да

7.3 СТАТУС ОПЕРАЦИИ

GET метод `/psp/external/api/payment/status` предназначен для запроса статуса операции в системе.

Запрос имеет один параметр: **id**, уникальный идентификатор операции.

Параметр является обязательным, тип данных — string.

Структура ответа:

```
{
  "attributes": [
    {
      "code": "string",
      "flags": 0,
      "title": "string",
      "value": "string",
      "valueTitle": "string"
    }
  ],
  "bankPayment": {
    "attributes": [
      {
        "code": "string",
        "flags": 0,
        "title": "string",
        "value": "string",
        "valueTitle": "string"
      }
    ]
  }
}
```

```
    }  
  ],  
  "error": 0,  
  "errorMessage": "string",  
  "status": "ERROR",  
  "transaction": "string"  
},  
"date": "string",  
"error": 0,  
"errorDetail": {  
  "errorFields": [  
    "string"  
  ],  
  "message": "string",  
  "type": "COMMISSION_ERROR"  
},  
"id": "string",  
"menu": {  
  "category": "string",  
  "service": 0,  
  "serviceName": "string"  
},  
"message": "string",  
"status": "ALREADY_EXISTS",  
"transaction": 0  
}
```

Таблица 7.3.1 — Атрибуты ответа на запрос статуса операции

Параметр	Описание	Обяз.
id	Уникальный идентификатор транзакции мерчанта	Да
status	Статус операции. Возможные статусы приведены в таблице I.1	Да
bankPayment. status	Статус банковской операции. Возможные статусы приведены в таблице I.1	Нет

bankPayment.transaction	Уникальный идентификатор транзакции банка	Нет
date	Дата создания операции, в формате уууу-ММ dd'T'HH:mm:ss	Нет
attributes	Объект данных атрибутов основной операции. Описание объекта приведено в таблице 7.7.3	Нет
menu	Объект данных сервиса, по которому осуществлялась оплата. Описание объекта приведено в таблице 7.7.4	Нет
transaction	Уникальный идентификатор транзакции на стороне PSP	Нет

Примеры запроса и ответа приведены в приложении [Е. Примеры запроса и ответов для метода запроса статуса операции](#).

7.4 ВОЗВРАТ

7.4.1 ОБЩАЯ ИНФОРМАЦИЯ

В системе предусмотрена процедура возврата денежных средств. Функционал возврата включает в себя два метода:

1. **/psp/external/api/payment/cancel/process** — метод операции возврата денежных средств;

2. `/psp/external/api/payment/cancel/status` — метод проверки статуса операции возврата.

Схема жизненного цикла возврата средств представлена на рисунке 7.4.1:

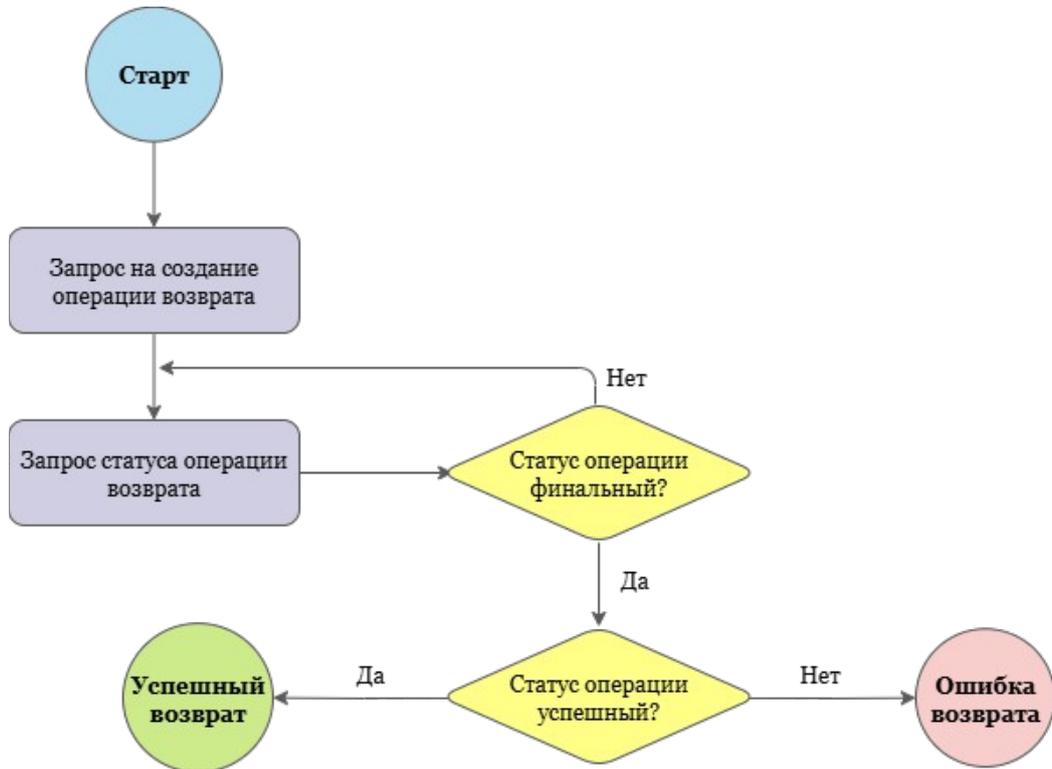


Рисунок 7.4.1 — Схема жизненного цикла операции возврата

7.4.2 СОЗДАНИЕ ОПЕРАЦИИ ВОЗВРАТА

POST метод `/psp/external/api/payment/cancel/process` предназначен для создания операции возврата денежных средств плательщику.

Таблица 7.4.2.1 — Атрибуты запроса на создание операции возврата

Параметр	Описание	Обяз.
id	Уникальный идентификатор транзакции на стороне мерчанта	Да
sum	Сумма транзакции, которую необходимо вернуть, в минимальных единицах валюты	Нет
comment	Комментарий	Нет



Внимание!

Частичный возврат недоступна, если по операциям была взята онлайн-комиссия.

Структура ответа:

```
{
  "comment": "string",
  "date": "string",
  "id": "string",
  "status": "ERROR",
  "sum": 0,
  "transaction": 0
}
```

Таблица 7.4.2.2 — Атрибуты ответа на запрос создания операции возврата

Параметр	Описание	Обяз.
id	Уникальный идентификатор транзакции на стороне мерчанта	Да

transaction	Уникальный идентификатор транзакции в системе PSP	Да
status	Статус выполнения запроса. Возможные статусы приведены в таблице I.2	Да
sum	Сумма возврата	Нет
date	Дата и время создания или последнего изменения состояния заявки на возврат, в формате уууу-ММ dd'T'HH:mm:ss	Нет

7.4.3 ЗАПРОС СТАТУСА ОПЕРАЦИИ ВОЗВРАТА

Для запроса статуса операции возврата используется метод `/psp/external/api/payment/cancel/status`.

Запрос имеет один параметр: `id`, уникальный идентификатор операции.

Параметр является обязательным, тип данных — `string`.

Структура ответа:

```
{
  "comment": "string",
  "date": "string",
  "id": "string",
  "status": "ERROR",
  "sum": 0,
  "transaction": 0
}
```

Таблица 7.4.3.1 — Атрибуты ответа на запрос статуса операции возврата

Параметр	Описание	Обяз.
id	Уникальный идентификатор транзакции на стороне мерчанта	Да
transaction	Уникальный идентификатор транзакции в системе PSP	Да
sum	Сумма возврата	Нет
comment	Комментарий	Нет
date	Дата и время создания или последнего изменения состояния заявки на возврат, в формате уууу-ММ dd'T'HH:mm:ss	Нет
status	Статус выполнения запроса. Возможные статусы приведены в таблице I.2	Да

Примеры запросов и ответов приведены в приложении [D. Примеры запроса и ответов для операции возврата](#).

7.5 РАСЧЕТ КОМИССИИ

Для расчета надбавочной комиссии необходимо использовать POST метод **/psp/check/calculationFees**. Метод вызывается перед созданием заказа в системе.

Структура запроса:

```
{
  "currency": "string",
  "service": 0,
  "source": "string"
  "sum": 0
}
```

Таблица 7.5.1 — Атрибуты запроса расчета комиссии

Параметр	Описание	Обяз.
currency	Валюта платежа	Да
service	Идентификатор сервиса на стороне PSP. Предоставляется мерчанту при интеграции системы	Да
source	Код источника оплаты. Возможные значения: <ul style="list-style-type: none">• SBP• SBERPAY• MOBILE_MONEY• BANKCARD-PS• HALYK_QR• ELQR	Нет
sum	Сумма заказа	Да

Структура ответа:

```
{
  "error": 0,
  "errorDetail": 0,
  "errorMessage": "string",
  "totalSums": {
```

```
"orderSum": 0,  
"customerSum": 0,  
"pspFee": 0,  
"currency": "string"  
"exchangeRate": 0  
}  
}
```

Таблица 7.5.2 — Атрибуты ответа на запрос расчета комиссии

Параметр	Описание	Обяз.
error	Код ошибки	Нет
errorDetail	Детализирующий код ошибки	Нет
errorMessage	Описание ошибки	Нет
totalSums []	Объект параметров суммы платежа и комиссии	-
totalSums [] . orderSum	Сумма заказа в валюте платежа	Да
totalSums [] . customerSum	Сумма к оплате клиентом, в валюте платежа	Да
totalSums [] . pspFee	Сумма комиссии, взимаемая с клиента, в валюте платежа	Да
totalSums [] . currency	Валюта мерчанта	Да

<code>totalSums [] . exchangeRate</code>	Курс валюты, который используется для конвертации если валюта платежа отличается от валюты мерчанта	Нет
--	---	-----

Примеры запроса и ответа приведены в приложении Г. Пример запроса и ответа для метода расчета комиссии.

7.6 БАЛАНС МЕРЧАНТА

Для запроса баланса необходимо реализовать поддержку метода `/psp/external/api/v2/balance`.

В ответе на запрос будет возвращена информация о счетах, включая: валюту, общий баланс, а также детализацию суммы общего баланса.

Структура запроса:

```
{
  "accounts": [
    {
      "awaitingEnrollment": 0,
      "balance": 0,
      "available": 0,
      "currency": "string",
      "rollingReserve": 0
    }
  ]
}
```

Таблица 7.6.1 — Параметры ответа на запрос баланса мерчанта

Параметр	Описание
----------	----------

currency	Валюта счета
balance	Общий баланс
rollingReserve	Удерживаемый резерв (<i>роллинг резерв</i>)
awaitingEnrollment	Блокированный баланс по операциям
available	Доступный для взаиморасчёта баланс (<i>Рассчитывается по формуле: Общий баланс – Блокированный баланс по операциям – Удерживаемый резерв – Взаиморасчеты в обработке</i>)

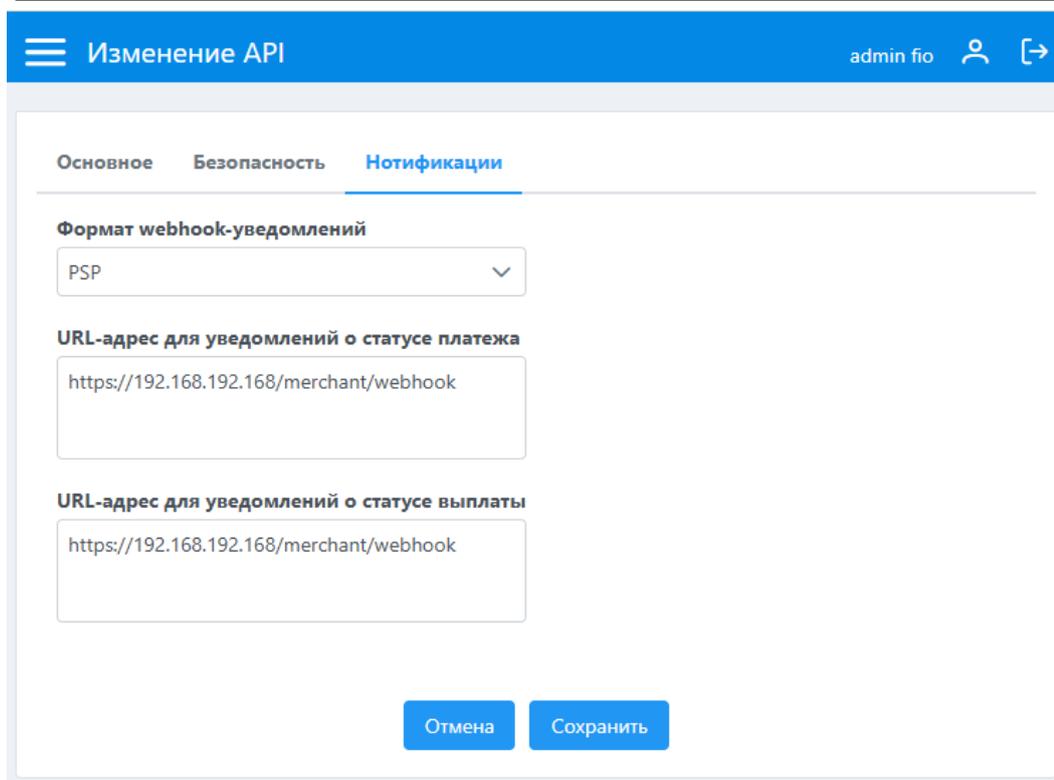
Пример ответа на запрос приведен в приложении G. Пример ответа на запрос баланса мерчанта.

7.7 УВЕДОМЛЕНИЕ ПАРТНЕРА О СТАТУСЕ ЗАКАЗА

В системе предусмотрен функционал callback-нотификации мерчанта о переходе операции транзакции в финальный статус (успех или ошибка). Функционал реализован с использованием технологии отправки webhook-оповещений.

Для настройки функционала, необходимо указать URL-адрес, на который будут отправляться webhook-оповещения. В настройках API на вкладке Нотификации (рисунок 7.7.1) нужно заполнить необходимый параметр:

- **«URL-адрес для отправки уведомлений о статусе платежа»;**
- **«URL-адрес для отправки уведомлений о статусе выплаты».**



Изменение API

admin fio [→]

Основное Безопасность **Нотификации**

Формат webhook-уведомлений

PSP

URL-адрес для уведомлений о статусе платежа

https://192.168.192.168/merchant/webhook

URL-адрес для уведомлений о статусе выплаты

https://192.168.192.168/merchant/webhook

Отмена Сохранить

Рисунок 7.7.1 — Заполнение URL-адреса для отправки webhook-оповещений в настройках API

Структура запроса отправки webhook-оповещения:

```
{
  "id" : string,
  "transaction" : 21799382,
  "status" : string,
  "errorDetail" : null,
  "error" : 0,
  "message" : null,
  "bankPayment" : null,
  "date" : string,
  "attribute" : [ {
    "code" : string,
    "value" : string,
    "title" : null,
    "valueTitle" : null,
  } ]
}
```

```

    "flags" : null
  } ],
  "menu" : {
    "service" : 8,
    "serviceName" : string,
    "category" : string
  },
  "sumOutcome" : 10000
}

```

Таблица 7.7.1 — Атрибуты запроса отправки webhook-оповещения

Параметр	Описание	Обязательность
id	Уникальный идентификатор транзакции мерчанта	Да
transaction	Уникальный идентификатор транзакции системы	Да
status	Статус операции, может принимать значения: <ul style="list-style-type: none"> • SUCCESS — Успех • ERROR — Ошибка 	Да
sumOutcome	Сумма операции, в копейках	Да
error	Код ошибки	Нет
errorDetail	Детализирующий код ошибки	Нет
message	Описание ошибки	Нет
bankPayment	Объект данных о банковской транзакции. Описание объекта приведено в таблице 7.7.2	Нет
date	Дата и время транзакции, в формате уууу-ММ	Нет

	dd'T'HH:mm:ss	
attribute	Объект данных атрибутов основной операции. Описание объекта приведено в таблице 7.7.3	Нет
menu	Объект данных сервиса. Описание объекта приведено в таблице 7.7.4	Нет

Таблица 7.7.2— Атрибуты объекта **bankPayment**

Параметр	Описание	Обязательность
status	Статус операции, может принимать значения: <ul style="list-style-type: none"> • SUCCESS — Успех • ERROR — Ошибка 	Нет
transaction	Уникальный идентификатор транзакции на стороне банка	Нет
attributes	Атрибуты банковской операции	Нет

Таблица 7.7.3— Атрибуты объекта **attribute**

Параметр	Описание	Обязательность
code	Код атрибута	Нет
value	Значение	Нет
title	Заголовок	Нет
valueTitle	Отображаемый заголовок	Нет

flags	Флаг	Нет

Таблица 7.7.4— Атрибуты объекта **menu**

Параметр	Описание	Обязательность
service	Код сервиса	Нет
serviceName	Название сервиса	Нет
category	Категория сервиса	Нет

В ответ от мерчанта системой будет ожидаться HTTP-ответ:

- **200** — webhook-оповещение успешно принято, система прекратит отправку;
- **401, 403** — доступ запрещен, система прекратит отправку;
- **404** — URL не найден, система прекратит отправку;
- **любой код, отличный от 200, 401, 403 и 404** — ошибка, система повторит отправку позже. Webhook-оповещение будет находиться в очереди на отправку 24 часа и будет периодически отправляться повторно. Если по истечению 24 часов, от мерчанта не будет получен успешный ответ, то система прекратит отправку.

Пример запроса приведен в приложении Н. Пример запроса отправки webhook-оповещения.

Для авторизации запросов webhook-оповещений используется проверка электронной подписи. Для этого в запросе передается в HTTP-заголовок **X-Sign**, содержащий в себе идентификатор API мерчанта, присвоенный в системе Paylogic.

Электронная подпись формируется от строки <REQUEST METHOD>+<REQUEST FULL URI>+< REQUEST BODY> BODY> по алгоритму SHA256+RSA в формате Base64 (UTF8).

Пример:

Webhook-оповещение отправляется мерчанту по адресу —
https://192.168.192.168/merchant/webhook

Подпись в данном примере будет иметь вид:

```
signBase64 (POST/merchant/webhook { "body" : "body" } )
```

Проверка подписи осуществляется по публичной части ключа, полученной от системы.

Пример генерации и проверки подписи приведен в приложении [А. Примеры генерации и проверки ЭП на java.](#)

8 ПРИЛОЖЕНИЕ

А. ПРИМЕРЫ ГЕНЕРАЦИИ И ПРОВЕРКИ ЭП НА JAVA

Пример генерации и проверки подписи API:

```
import org.apache.commons.codec.binary.Base64;
import org.apache.commons.codec.binary.Hex;
import org.bouncycastle.openssl.PEMReader;
import org.bouncycastle.openssl.PasswordFinder;
import java.io.FileReader;
import java.io.IOException;
import java.security.*;

public class Signer {
    public static final String defaultAlgorithm = "SHA256withRSA";
    public static final String defaultCharset = "UTF-8";

    private PrivateKey privateKey;
    private PublicKey publicKey;
    private String algorithm;
    private String charset;

    public Signer(String publicKeyPath, String privateKeyPath) throws IOException
    {
        try (FileReader fr = new FileReader(publicKeyPath);
            PEMReader pr = new PEMReader(fr)) {
            publicKey = (PublicKey) pr.readObject();
        }

        try (FileReader fr = new FileReader(privateKeyPath);
            PEMReader pr = new PEMReader(fr)) {
            KeyPair kp = (KeyPair) pr.readObject();
            privateKey = kp.getPrivate();
        }
    }

    public boolean verifyBase64(String message, String signature) throws
    SignatureException {
        try {
            Signature sign = Signature.getInstance(defaultAlgorithm);
            sign.initVerify(publicKey);
            sign.update(message.getBytes(defaultCharset));
            return sign.verify(Base64.decodeBase64(signature.getBytes(charset)));
        }
    }
}
```

```
    } catch (Exception ex) {  
        throw new SignatureException(ex);  
    }  
}  
  
public String signBase64(String message) throws SignatureException {  
    try {  
        Signature sign = Signature.getInstance(defaultAlgorithm);  
        sign.initSign(privateKey);  
        sign.update(message.getBytes(defaultCharset));  
        return new String(Base64.encodeBase64(sign.sign()), charset);  
    } catch (Exception ex) {  
        throw new SignatureException(ex);  
    }  
}
```

Пример подписи генерации и проверки подписи при отправке webhook-оповещения:

```
import org.apache.commons.codec.binary.Base64;  
import org.apache.commons.codec.binary.Hex;  
import org.bouncycastle.openssl.PEMReader;  
import org.bouncycastle.openssl.PasswordFinder;  
  
import java.io.FileReader;  
import java.io.IOException;  
import java.security.*;  
  
public class Signer {  
  
    public static final String defaultAlgorithm = "SHA256withRSA";  
    public static final String defaultCharset = "UTF-8";  
  
    private PrivateKey privateKey;  
    private PublicKey publicKey;  
    private String algorithm;  
    private String charset;  
  
    public Signer(String publicKeyPath) throws IOException {  
        try (FileReader fr = new FileReader(publicKeyPath);  
            PEMReader pr = new PEMReader(fr)) {  
            publicKey = (PublicKey) pr.readObject();  
        }  
    }  
}
```

```
public boolean verifyBase64(String message, String signature) throws
SignatureException {
    try {
        Signature sign = Signature.getInstance(defaultAlgorithm);
        sign.initVerify(publicKey);
        sign.update(message.getBytes(defaultCharset));
        return sign.verify(Base64.decodeBase64(signature.getBytes(charset)));
    } catch (Exception ex) {
        throw new SignatureException(ex);
    }
}
```

В. ПРИМЕРЫ ЗАПРОСА И ОТВЕТОВ ДЛЯ ОПЕРАЦИИ ПЛАТЕЖА

Пример запроса на создание операции платежа с вводом данных на стороне банка:

```
{
  "checkNumber": 10043,
  "currency": "RUB",
  "date": "2025-05-18T07:49:20.946Z",
  "feeSum": 0,
  "id": "external-id-123456",
  "params": [{
    "code": "id1",
    "value": "790000000000"
  }],
  "purchaseSum": 1,
  "service": 5,
  "source": {
    "code": "SBP",
    "params": [{
      "code": "pma",
      "value": "790000000000"
    }],
    {
      "code": "redirectUrlSuccess",
      "value": "https://merchant.ru/"
    },
    {
      "code": "client-ipAddress",
      "value": "2a01:4f8:c2c:37c1::1"
    }
  ],
}
```

```
{
  "code": "email",
  "value": "support.m@merchant.ru"
}]
}
```

Пример успешного ответа на запрос с вводом данных на стороне банка:

```
{
  "id": "external-id-123456",
  "transaction": 1391191,
  "status": "PROCESSING",
  "errorDetail": null,
  "error": 0,
  "message": null,
  "bankPayment": {
    "status": "PROCESSING",
    "transaction": "ps_1027240831",
    "attributes": [
      {
        "code": "order-id",
        "title": "order-id",
        "value": "ps_1027240831",
        "valueTitle": "ps_1027240831",
        "flags": 0
      },
      {
        "code": "payload",
        "title": "payload",
        "value": "https://merchant.ru/",
        "valueTitle": "https://merchant.ru/",
        "flags": 0
      }
    ]
  },
  "error": null,
  "errorMessage": null
},
  "date": "2025-05-19T04:20:28.139+0000",
  "attributes": null,
  "menu": null
}
```

Пример ошибочного ответа на запрос с вводом данных на стороне банка:

```
{
  "id": "20250522140659",
```

```
"transaction": 54575293,  
"status": "ERROR",  
"errorDetail": null,  
"error": -2,  
"message": null,  
"bankPayment": {  
  "status": "ERROR",  
  "transaction": null,  
  "attributes": [],  
  "error": 9,  
  "errorMessage": "Авторизация не выполнена"  
},  
"date": "2025-05-22T07:07:00.666+0000",  
"attributes": null,  
"menu": null  
}
```

Пример запроса на создание операции платежа через систему SberPay:

```
{  
  "checkNumber": 10002,  
  "currency": "RUB",  
  "date": "2025-06-30T06:54:29.136Z",  
  "feeSum": 0,  
  "id": "external-id-123456",  
  "params": [{  
    "code": "id1",  
    "value": "79600000000"  
  }],  
  "purchaseSum": 1,  
  "service": 13,  
  "source": {  
    "code": "SBERPAY",  
    "params": [{  
      "code": "pma",  
      "value": "79600000000"  
    },  
    {  
      "code": "serviceCode",  
      "value": "111-19318-39"  
    },  
    {  
      "code": "frontType",  
      "value": "back2app"  
    }  
  ]  
}  
}
```

```
}
```

Пример успешного ответа на запрос создания операции платежа через систему SberPay:

```
{
  "id": "external-id-123456",
  "transaction": 61110152,
  "status": "PROCESSING",
  "errorDetail": null,
  "error": 0,
  "message": null,
  "bankPayment": {
    "status": "PROCESSING",
    "transaction": "192183256",
    "attributes": [
      {
        "code": "order-id",
        "title": "order-id",
        "value": "192183256",
        "valueTitle": "192183256",
        "flags": 0
      },
      {
        "code": "rrn",
        "title": "rrn",
        "value": "192183256",
        "valueTitle": "192183256",
        "flags": 0
      }
    ],
    "error": null,
    "errorMessage": null
  },
  "date": "2025-06-30T07:54:52.424+0000",
  "attributes": null,
  "menu": null
}
```

Пример запроса на создание операции платежа через HALYK QR:

```
{
  "checkNumber": 10001,
  "currency": "KZT",
  "date": "2025-11-07T11:07:28.893",
  "feeSum": 0,
}
```

```
"id": "20251107110728",
"params": [{
  "code": "id1",
  "value": "1234567890"
}],
"purchaseSum": 100,
"service": 2,
"source": {
  "code": "HALYK_QR",
  "params": [
    {
      "code": "pma",
      "value": "790000000000"
    }
  ]
}
}
```

Пример успешного ответа на запрос создания операции платежа через систему HALYK QR:

```
{
  "id": "20251107110728",
  "transaction": 1030557,
  "status": "PROCESSING",
  "errorDetail": null,
  "error": 0,
  "message": null,
  "bankPayment": {
    "status": "PROCESSING",
    "transaction": "47848222211",
    "attributes": [
      {
        "code": "token",
        "title": "token",
        "value": "VPNNYV08JT1XWCQ5WF2TX2",
        "valueTitle": "VPNNYV08JT1XWCQ5WF2TX2",
        "flags": 0
      },
      {
        "code": "token-expires-in",
        "title": "token-expires-in",

```

```
        "value": "1200",
        "valueTitle": "1200",
        "flags": 0
    },
    {
        "code": "token-generated-date",
        "title": "token-generated-date",
        "value": "2025-11-07T04:07:29.623Z",
        "valueTitle": "2025-11-07T04:07:29.623Z",
        "flags": 0
    },
    {
        "code": "rrn",
        "title": "rrn",
        "value": "47848222211",
        "valueTitle": "47848222211",
        "flags": 0
    },
    {
        "code": "payload",
        "title": "payload",
        "value": "https://bank.kz/payments/halykqr/47848222211?
AMT=1&CUR=KZT&pointName=EPAY-TEST-
PAYMENT&message=&billId=47848222211&qrCode=000305010212021645989595979091
3127400012KZ.HALYKBANK0108HSBKKZKX02089691363428660011478482222110120KZ49
60171310000239340208969136340301N0501006010520445115303398540115802KZ5917
EPAY-TEST-
PAYMENT6006ALMATY6238011147848222211080319110121307400062786304D810",
        "valueTitle":
        "https://bank.kz/payments/halykqr/47848222211?
AMT=1&CUR=KZT&pointName=EPAY-TEST-
PAYMENT&message=&billId=47848222211&qrCode=000305010212021645989595979091
3127400012KZ.HALYKBANK0108HSBKKZKX02089691363428660011478482222110120KZ49
60171310000239340208969136340301N0501006010520445115303398540115802KZ5917
EPAY-TEST-
PAYMENT6006ALMATY6238011147848222211080319110121307400062786304D810",
        "flags": 0
    },
    {
        "code": "status-QR",
        "title": "status-QR",
        "value": "NEW",
```

```
        "valueTitle": "NEW",
        "flags": 0
    }
],
"error": null,
"errorMessage": null
},
"date": "2025-11-07T04:07:29.372+0000",
"attributes": null,
"menu": null,
"conversionResult": null
}
```

Пример запроса на создание операции платежа через систему ELQR:

```
{
  "checkNumber": 0,
  "currency": "RUB",
  "date": "2025-11-24T14:58:29.983",
  "feeSum": 0,
  "id": "external-id-123456",
  "params": [{
    "code": "id1",
    "value": "1234567890"
  }],
  "purchaseSum": 1000,
  "service": 2,
  "source": {
    "code": "ELQR",
    "params": [{
      "code": "pma",
      "value": "1234567890"
    }]
  }
}
```

Пример успешного ответа на запрос создания операции платежа через систему ELQR:

```
{
  "id": "external-id-123456",
  "transaction": 73530300,
  "status": "PROCESSING",
}
```

```
"errorDetail": null,  
"error": 0,  
"message": null,  
"bankPayment": {  
  "status": "PROCESSING",  
  "transaction": "005226398",  
  "attributes": [  
    {  
      "code": "rrn",  
      "title": "rrn",  
      "value": "005376398",  
      "valueTitle": "005376398",  
      "flags": 0  
    },  
    {  
      "code": "orderId",  
      "title": "orderId",  
      "value": "26327268",  
      "valueTitle": "26327268",  
      "flags": 0  
    },  
    {  
      "code": "payload",  
      "title": "payload",  
      "value":  
"https://test.ru/BD10001400BCNDKV9QUAS80BVUEKEK90",  
      "valueTitle":  
"https://test.ru/BD10001400BCNDKV9QUAS80BVUEKEK90",  
      "flags": 0  
    }  
  ],  
  "error": null,  
  "errorMessage": null  
},  
"date": "2025-11-24T09:00:17.370+0000",  
"attributes": null,  
"menu": null  
}
```

Пример запроса на создание операции платежа через систему Mobile Money:

```
{  
  "checkNumber": 10001,  
}
```

```
"currency": "UGX",
"date": "2025-08-06T16:21:16.511",
"feeSum": 0,
"id": "20250307162217",
"params": [{
  "code": "id1",
  "value": "0900000001"
}],
"purchaseSum": 100,
"service": 2,
"source": {
  "code": "MOBILE_MONEY",
  "params": [{
    "code": "pma",
    "value": "0900000001"
  }],
  {
    "code": "country",
    "value": "UG"
  },
  {
    "code": "providerId",
    "value": "14"
  },
  {
    "code": "customerName",
    "value": "Johnson John"
  },
  {
    "code": "customerEmail",
    "value": "test@mail.com"
  },
  {
    "code": "redirectUrlProcess",
    "value": "https://merchant.ru/"
  }
]}
}
```

Пример успешного ответа на запрос создания операции платежа через систему Mobile Money:

```
{
  "id": "20250806162013",
  "transaction": 1030580,
  "status": "PROCESSING",
  "errorDetail": null,
  "error": 0,
  "message": null,
  "bankPayment": {
    "status": "PROCESSING",
    "transaction": "284319940",
    "attributes": [
      {
        "code": "rrn",
        "title": "rrn",
        "value": "284319940",
        "valueTitle": "284319940",
        "flags": 0
      },
      {
        "code": "waitCallback",
        "title": "waitCallback",
        "value": "true",
        "valueTitle": "true",
        "flags": 0
      }
    ],
    "error": null,
    "errorMessage": null
  },
  "date": "2025-08-06T09:20:14.762+0000",
  "attributes": null,
  "menu": null
}
```

Пример запроса на создание операции платежа с вводом данных на стороне мерчанта:

```
{
  "checkNumber": 10001,
  "currency": "RUB",
  "date": "2024-10-21T13:11:03.582Z",
  "feeSum": 0,
}
```

```
"id": "external-id-123456",
"params": [{
  "code": "id1",
  "value": "790000000000"
}],
"purchaseSum": 51.01,
"service": 6,
"source": {
  "code": "BANKCARD-PS",
  "params": [{
    "code": "client-ipAddress",
    "value": "10.192.168.1"
  },
  {
    "code": "client-browserInfo",
    "value": "Y2hyb211"
  },
  {
    "code": "client-agentName",
    "value": "AgentName"
  },
  {
    "code": "pan",
    "value": "2201382000000039"
  },
  {
    "code": "expiry",
    "value": "1224"
  },
  {
    "code": "holder",
    "value": "Ivanov Ivan"
  },
  {
    "code": "cvv",
    "value": "123"
  },
  {
    "code": "terminal-id",
    "value": "merchant-test"
  }
  ],
  }
```

```
        "code": "redirectUrlExecuteIframe2",
        "value": "https://merchant.ru/server/"
    },
    {
        "code": "redirectUrlExecuteAdvanced",
        "value": "https://merchant.ru/server/about.seam"
    },
    {
        "code": "redirectUrlSuccess",
        "value": "https://processing.merchant.ru/result/success"
    },
    {
        "code": "redirectUrlError",
        "value": "https://processing.merchant.ru /result/error"
    }
]
}
```

Пример успешного ответа на запрос с вводом данных на стороне мерчанта с аутентификацией по протоколу 3DSv1:

```
{
  "id": "external-id-123456",
  "transaction": 12345678,
  "status": "PROCESSING",
  "errorDetail": null,
  "error": 0,
  "message": null,
  "bankPayment": {
    "status": "PROCESSING",
    "transaction": null,
    "attributes": [
      {
        "code": "gateway-session-id",
        "title": "gateway-session-id",
        "value": "461a60f7-e31a-41ce-9341-abf8d37ae0f0",
        "valueTitle": "461a60f7-e31a-41ce-9341-abf8d37ae0f0",
        "flags": 0
      },
      {
        "code": "url-3DS",
```

```
        "title": "url-3DS",
        "value":
"https://acs2.sbrf.ru:443/ncs_06/acs/pareq/02_06ce40bebc20ee405094f868acb
5fd0d29",
        "valueTitle":
"https://acs2.sbrf.ru:443/ncs_06/acs/pareq/02_06ce40bebc20ee405094f868acb
5fd0d29",
        "flags": 0
    },
    {
        "code": "md",
        "title": "md",
        "value": "b45c3799-80aa-4310-bf94-f52c92fe17e8",
        "valueTitle": "b45c3799-80aa-4310-bf94-f52c92fe17e8",
        "flags": 0
    },
    {
        "code": "paReq",
        "title": "paReq",
        "value":
"eJxVUu9v2jAQ/VeifB/+mYygi6u2aKybKLSlm+DL5NguRCWGOs4G/etrU7Jukj/cuzu9e/
fOCHFotslv49p6Z8uUDHCAgKt2urbrMn1cfPk0TJPWS6vldmdNmR5Nm14IWGycMeMHozpnBEx
N28q1SWpdpqXgKl0a6aqqXGNWZeQzZbTQ7ImxospTAFPLe/
Mi4DxVhKEDCqiHgc2pjbRegFQvVze3IuN5gQmgM4TGujuxKIYEY0yy8HJA7zmwsjHi+vuvyWR
8P5sD0mFQu856dxQ5Z4B6AJ3bio33+3aEkHqWYyWB69BeHhtjPQIU64A+1My7GLWB71BrMW1W
9e3PVT1bPPJ1s2TL1/
Vh1Uz5bLEuAcU00NIbQTH1BFOS4GJE8hENCk55kE0UIrKwftztHcE+DrnsS7HybwaC3y7cp1+
1R2AO+3Ce0BGM/BsD+pB8/
TXaqXwwKeec0GFBsx95fqefzbj64yffqo5ROosmn5oiYx1cohWPT5QRAIo06Hw/
dD57iP77Dm+UmcGt",
        "valueTitle":
"eJxVUu9v2jAQ/VeifB/+mYygi6u2aKybKLSlm+DL5NguRCWGOs4G/etrU7Jukj/cuzu9e/
fOCHFotslv49p6Z8uUDHCAgKt2urbrMn1cfPk0TJPWS6vldmdNmR5Nm14IWGycMeMHozpnBEx
N28q1SWpdpqXgKl0a6aqqXGNWZeQzZbTQ7ImxospTAFPLe/
Mi4DxVhKEDCqiHgc2pjbRegFQvVze3IuN5gQmgM4TGujuxKIYEY0yy8HJA7zmwsjHi+vuvyWR
8P5sD0mFQu856dxQ5Z4B6AJ3bio33+3aEkHqWYyWB69BeHhtjPQIU64A+1My7GLWB71BrMW1W
9e3PVT1bPPJ1s2TL1/
Vh1Uz5bLEuAcU00NIbQTH1BFOS4GJE8hENCk55kE0UIrKwftztHcE+DrnsS7HybwaC3y7cp1+
1R2AO+3Ce0BGM/BsD+pB8/
TXaqXwwKeec0GFBsx95fqefzbj64yffqo5ROosmn5oiYx1cohWPT5QRAIo06Hw/
dD57iP77Dm+UmcGt",
        "flags": 0
    }
}
```

```
    },
    {
      "code": "order-id",
      "title": "order-id",
      "value": "123456809853644",
      "valueTitle": "123456809853644",
      "flags": 0
    },
    {
      "code": "redirect-url-3DS",
      "title": "redirect-url-3DS",
      "value":
"https://merchant.ru/external/payment-tools/callback/26379485/37",
      "valueTitle": "https://merchant.ru/external/payment-
tools/callback/26379485/37",
      "flags": 0
    }
  ],
  "error": null,
  "errorMessage": null
},
"date": "2024-10-21T09:13:45.108+0000",
"attributes": null,
"menu": null
}
```

Пример успешного ответа на запрос с вводом данных на стороне мерчанта с аутентификацией по протоколу 3DSv2:

```
{
  "id": "external-id-123456",
  "transaction": 12345678,
  "status": "PROCESSING",
  "errorDetail": null,
  "error": 0,
  "message": null,
  "bankPayment": {
    "status": "PROCESSING",
    "transaction": null,
    "attributes": [
      {
        "code": "gateway-session-id",
```

```
        "title": "gateway-session-id",
        "value": "183e1bc5-168b-4173-a784-0d55a74edc33",
        "valueTitle": "183e1bc5-168b-4173-a784-0d55a74edc33",
        "flags": 0
    },
    {
        "code": "order-id",
        "title": "order-id",
        "value": "123456809853683",
        "valueTitle": "123456809853683",
        "flags": 0
    },
    {
        "code": "redirect-url-3DS",
        "title": "redirect-url-3DS",
        "value":
"https://merchant.ru/external/payment-tools/callback/26380074/37",
        "valueTitle": "https://merchant.ru/external/payment-
tools/callback/26380074/37",
        "flags": 0
    },
    {
        "code": "bank-operation-id",
        "title": "bank-operation-id",
        "value": "26380074",
        "valueTitle": "26380074",
        "flags": 0
    },
    {
        "code": "hidden-request-2-url",
        "title": "hidden-request-2-url",
        "value": "https://3ds-ds1.mirconnect.ru/ma",
        "valueTitle": "https://3ds-ds1.mirconnect.ru/ma",
        "flags": 0
    },
    {
        "code": "threeDSMethodData",
        "title": "threeDSMethodData",
        "value":
"eyJ0aHJlZURTU2VydMvYVhJhbnNJRCI6IjBjZjI3OWQwLWMxYWVtNDEzYS1iZWU4LTk3MwYx
NDYyM2E2ZiIsInRocmVlRFNNZXRob2R0b3RpZmljYXRpb25VUkwiOiJodHRwczovL2FwcC5vb
```

```
mxpcGF5LmNvbS9leHRlcm5hbC9wYXltZW50LXRvb2xzL2NhbGxiYWNRlZlI2MzgwMDc0LzM3In
OK",
      "valueTitle":
"eyJ0aHJlZURTU2VydmVyVHJhbnNJRCI6IjBjZjI3OWQwLWMxYWVhbnNDEzYS1iZWU4LTk3MmYx
NDYyM2E2ZiIsInRocmVlRFNNZXRob2ROb3RpZmljYXRpb25VUkwiOiJodHRwczovL2FwcC5vb
mxpcGF5LmNvbS9leHRlcm5hbC9wYXltZW50LXRvb2xzL2NhbGxiYWNRlZlI2MzgwMDc0LzM3In
OK",
      "flags": 0
    }
  ],
  "error": null,
  "errorMessage": null
},
"date": "2024-10-21T09:21:47.217+0000",
"attributes": null,
"menu": null
}
```

С. ПРИМЕРЫ ЗАПРОСА И ОТВЕТОВ ДЛЯ ОПЕРАЦИИ ВЫПЛАТЫ

Пример запроса на создание операции выплаты:

```
{
  "checkNumber": 1,
  "date": "2024-02-28T06:44:34.106Z",
  "feeSum": 0,
  "id": "pltest117",
  "params": [
    {
      "code": "id1",
      "value": "*****3057"
    },
    {
      "code": "pan",
      "value": "2200356884553057"
    },
    {
      "code": "fio",
      "value": "Иванов Иван Иванович"
    }
  ],
}
```

```
"purchaseSum": 1,  
"service": 3,  
"source": {  
  "code": "BANKCARD",  
  "params": null  
}  
}
```

Пример ответа на создание операции выплаты:

```
{  
  "id": "pltest117",  
  "transaction": 1391241,  
  "status": "PROCESSING",  
  "errorDetail": null,  
  "error": 0,  
  "message": null,  
  "bankPayment": null,  
  "date": "2024-02-28T03:37:44.354+0000",  
  "attributes": null,  
  "menu": null  
}
```

Д. ПРИМЕРЫ ЗАПРОСА И ОТВЕТОВ ДЛЯ ОПЕРАЦИИ ВОЗВРАТА

Примеры запроса на создание операции возврата:

```
Curl:  
curl -X POST "https://{host}/psp/external/api/payment/cancel/process?  
id=ext123456&sum=100&comment=cancel" -H "accept: */*" -d ""  
  
Request URL:  
https://{host}/psp/external/api/payment/cancel/process?  
id=ext123456&sum=100&comment=cancel
```

Пример ответа на создание операции возврата:

```
{  
  "id": "ext123456",  
  "transaction": 30778238,  
  "sum": 1.00,  
}
```

```
"comment": "cancel",  
"date": "2024-11-27T09:52:44.009+0000",  
"status": "PROCESSING"  
}
```

Е. ПРИМЕРЫ ЗАПРОСА И ОТВЕТОВ ДЛЯ МЕТОДА ЗАПРОСА СТАТУСА ОПЕРАЦИИ

Пример запроса статуса операции списания:

```
/psp/external/api/payment/status?id=external-id-123456
```

Пример ответа на запрос статуса операции списания:

```
{  
  "id": "external-id-123456",  
  "transaction": 1391191,  
  "status": "SUCCESS",  
  "errorDetail": null,  
  "error": 0,  
  "message": null,  
  "bankPayment": {  
    "status": "SUCCESS",  
    "transaction": "ps_1027240831",  
    "attributes": null,  
    "error": null,  
    "errorMessage": null  
  },  
  "date": "2025-05-18T07:48:37.264+0000",  
  "attributes": [  
    {  
      "code": "id1",  
      "title": null,  
      "value": "790000000000",  
      "valueTitle": null,  
      "flags": 0  
    }  
  ],  
  "menu": {  
    "service": 5,  
    "serviceName": "Оплата через СБП",  
    "category": "Списание СБП"  
  }  
}
```

```
}  
}
```

Е. ПРИМЕР ЗАПРОСА И ОТВЕТА ДЛЯ МЕТОДА РАСЧЕТА КОМИССИИ

Пример запроса расчета комиссии:

```
{  
  "currency": "USD",  
  "service": 4341,  
  "source": "BANKCARD-PS",  
  "sum": 100  
}
```

Пример успешного ответа:

```
{  
  "error": 0,  
  "errorDetail": 0,  
  "totalSums": {  
    "exchangeRate": 83.63,  
    "currency": "RUB",  
    "orderSum": 1.2,  
    "customerSum": 1.42,  
    "pspFee": 0.22  
  }  
}
```

Г. ПРИМЕР ОТВЕТА НА ЗАПРОС БАЛАНСА МЕРЧАНТА

```
{  
  "accounts": [  
    {  
      "awaitingEnrollment": 0,  
      "balance": 5793817,  
      "available": 3898167,  
      "currency": "RUB",  
      "rollingReserve": 20000  
    },  
  ],  
}
```

```
{
  "awaitingEnrollment": 0,
  "balance": 1605200,
  "available": 1605200,
  "currency": "USD",
  "rollingReserve": 0
},
{
  "awaitingEnrollment": 0,
  "balance": 10000,
  "available": 10000,
  "currency": "USDT",
  "rollingReserve": 0
},
{
  "awaitingEnrollment": 0,
  "balance": 0,
  "available": 0,
  "currency": "USDC",
  "rollingReserve": 0
}
]
```

Н. ПРИМЕР ЗАПРОСА ОТПРАВКИ WEBHOOK-ОПОВЕЩЕНИЯ

```
{
  "id" : "external-id-123456",
  "transaction" : 1391191,
  "status" : "SUCCESS",
  "errorDetail" : null,
  "error" : 0,
  "message" : null,
  "bankPayment" : {
    "status" : "SUCCESS",
    "transaction" : "ps_13755949",
    "attributes" : null
  },
  "date" : "2025-05-18",
  "attribute" : [ {
    "code" : "merch-url",
```

```
"value" : "https://test.ru",
"title" : null,
"valueTitle" : null,
"flags" : null
}, {
  "code" : "id1",
  "value" : "",
  "title" : null,
  "valueTitle" : null,
  "flags" : null
}, {
  "code" : "id2",
  "value" : null,
  "title" : null,
  "valueTitle" : null,
  "flags" : null
} ],
"menu" : {
  "service" : 5,
  "serviceName" : "Оплата через СВП",
  "category" : "Списание СВП"
},
"sumOutcome" : 24796
}
```

I. СТАТУСЫ ОПЕРАЦИЙ

Таблица I.1 – Статусы операций платежа и выплаты

Статус	Описание
SUCCESS	Операция успешно обработана. Финальный статус
ERROR	Ошибка проведения операции. Финальный статус
NOT_FOUND	Не найден id операции
PROCESSING	Проведение
VALIDATION_ERROR	Ошибка валидации

Таблица I.2 – Статусы операции возврата

Статус	Описание
PROCESSING	Заявка на отмену создана и обрабатывается
SUCCESS	Заявка на отмену успешно обработана, статус основной транзакции изменен на "CANCELLED". Финальный статус
ERROR	Заявка на отмену завершилась ошибкой, статус основной транзакции остался "SUCCESS". Финальный статус
PAYMENT_NOT_FOUND	Транзакция для отмены не найдена
PAYMENT_WRONG_STATUS	Транзакция для отмены найдена, но её статус не позволяет произвести отмену
SUM_PARTIAL_FORBIDDEN	Частичная отмена транзакции запрещена
SUM_IS_NOT_VALID	Переданная для частичной отмены транзакции сумма меньше или больше допустимой. Важно! Частичная отмена так же недоступна, если по операциям была взята онлайн-комиссия
VALIDATION_ERROR	Ошибка валидации данных запроса
SYSTEM_ERROR	Системная ошибка

1. ПРИМЕРЫ ФОРМ ДЛЯ РЕДИРЕКТА ДЛЯ АВТОРИЗАЦИИ ПО ПРОТОКОЛУ 3DS

Пример формы с параметрами `paReq` и `md` для редиректа по протоколу 3DSv1:

```
<form class="pos-relative d-flex align-items-center justify-content-center mt30 mb30" target="_self" name="3dsform" id="3dsform" method="post" action="XXXXX">
  <input type="hidden" name="MD" value="XXXXX">
  <input type="hidden" name="PaReq" value="XXXXX">
  <input type="hidden" name="TermUrl" value="XXXXX">
  <button class="card-button waves-effect waves-light" id="btnSend" type="submit" th:text="#{goToBank.btn}">Перейти</button>
  <div class="card-button__shadow"></div>
</form>
```

Пример формы с параметрами paReq для редиректа по протоколу 3DSv1:

```
<form class="pos-relative d-flex align-items-center justify-content-center mt30 mb30" target="_self" name="3dsform" id="3dsform" method="post" action="XXXXX">
  <input type="hidden" name="PaReq" value="XXXXX">
  <button class="card-button waves-effect waves-light" id="btnSend" type="submit" th:text="#{goToBank.btn}">Перейти</button>
  <div class="card-button__shadow"></div>
</form>
```

Пример формы с параметрами url-3DS, creq и threeDSSessionData для редиректа по протоколу 3DSv2:

```
<form class="pos-relative d-flex align-items-center justify-content-center mt30 mb30" target="_self" name="3dsform" id="3dsform" method="post" action="XXXXX">
  <input type="hidden" name="gatewaySessionId" value="XXXXX">
  <input type="hidden" name="threeDSSessionData" value="XXXXX">
  <input type="hidden" name="creq" value="XXXXX">
  <button class="card-button waves-effect waves-light" id="btnSend" type="submit" th:text="#{goToBank.btn}">Перейти</button>
  <div class="card-button__shadow"></div>
</form>
```

Пример формы url-3DS и creq для редиректа по протоколу 3DSv2:

```
<form class="pos-relative d-flex align-items-center justify-content-center mt30 mb30" target="_self" name="3dsform" id="3dsform" method="post" action="XXXXX">
  <input type="hidden" name="creq" value="XXXXX">
```

```
<button class="card-button waves-effect waves-light" id="btnSend"  
type="submit" th:text="#{goToBank.btn}">Перейти</button>  
<div class="card-button__shadow"></div>  
</form>
```

Пример формы для редиректа при выполнении скрытого *iframe1*:

```
<form id="tdsMethod3DsServer" method="post"  
action="threeDSMethodURLServer" target="tdsMethod3DsServerFrame"></form>  
<iframe name="https://docs.bisys.ru/doc/3ds-acq/#auth" width="0"  
height="0" style="display: none;"></iframe>
```

Пример формы для редиректа при выполнении скрытого *iframe2*:

```
<form name="test" method="post" action="XXXXX">  
  <input type="text" name="threeDSMethodData" size="300"  
maxlength="300" value="XXXXX">  
</form>  
  
<script type="text/javascript">  
  document.test.submit();  
</script>
```