



Банк России

ОКТАБРЬ 2020

ОТКРЫТЫЕ БАНКОВСКИЕ ИНТЕРФЕЙСЫ

Инициирование перевода денежных средств клиента третьей стороной в валюте Российской Федерации

Москва

2020

Оглавление

1	История изменений	4
2	Введение	5
2.1	Область применения	5
2.2	Термины и определения	5
2.3	Принципы архитектуры	6
2.3.1	Идентификаторы	7
2.3.2	Коды статусов	7
3	Основы	8
3.1	Роли участников	8
3.2	Кодировка символов	8
3.3	Формат даты	9
3.4	Структура пути URI ресурса	9
3.5	Заголовки сообщений (headers)	10
3.5.1	Заголовки запросов	10
3.5.2	Заголовки ответов	12
3.6	Коды статусов HTTP	13
3.6.1	400 (Bad Request) или 404 (Not Found)	16
3.6.2	403 (Forbidden)	16
3.6.3	401 (Unauthorized)	17
3.6.4	429 (Too Many Requests)	17
3.7	Идемпотентность	17
3.8	Фильтрация	18
3.9	Нумерация страниц	18
3.10	Архивирование	19
3.11	Дополнительные данные	19
4	Модель данных	20
4.1	Справочники и перечисления	20
4.2	Общая структура полезной нагрузки	20
4.2.1	Структура запроса	20
4.2.2	Структура ответа	21
4.2.3	Структура ответа с ошибками	21
4.2.4	Необязательные поля	23
4.2.5	Ссылки	24
4.2.6	Метаданные	24
5	Примеры использования	25
5.1	Потоки с нумерацией страниц	25
6	Инициирование переводов денежных средств, спецификация API - v1.2.1	26
6.1	Введение	26
6.1.1	Принципы проектирования	26
6.2	Основы	26
6.2.1	Общее описание процесса	26
6.2.2	Ограничения при инициировании платежей	29
6.2.3	Сохранение обратной совместимости между мажорными версиями	31
6.3	Конечные точки	32
6.4	Безопасность и контроль доступа	32
6.4.1	Scopes	32
6.4.2	Grants Types	32
6.4.3	Авторизация согласия	32
6.4.4	Информация по оценке рисков	34
6.5	Модель данных	35
6.5.1	Повторно используемые объекты	35
6.5.2	Поля для идентификаций	49
6.5.3	Виды платежей	51
6.5.4	Справочники	51
6.6	Внутренние переводы денежных средств, спецификация API - v1.2.1	54
6.6.1	Конечные точки	54
6.6.2	Модель данных	59

6.6.3	Примеры использования	92
7	Динамические справочники в пространствах имен - v1.0.0	105
7.1	Введение	105
7.2	Основы.....	105
7.2.1	Принципы проектирования	105
7.2.2	Управление версиями.....	105
7.3	Общие справочники.....	105

1 История изменений

Версия	Дата	Автор	Комментарий
1.0.0	21.08.2019	АФТ, Направление открытых API	Создана первая версия документа
1.0.1	27.11.2019	АФТ, Направление открытых API	Внесены минорные изменения.
1.0.2	10.02.2020	АФТ, Направление открытых API	Внесены минорные изменения по замечаниям Центрального банка Российской Федерации.
1.1.0	08.04.2020	АФТ, Направление открытых API	Внесены изменения по замечания Банка России и участников Ассоциации развития финансовых технологий. Для повторно используемого класса "Initiation" добавлены разделы: <ul style="list-style-type: none">• CreditorParty;• CreditorAgent.
1.2.1	17.06.2020	АФТ, Направление открытых API	Внесены изменения по замечаниям рабочей группы ТК122 ПКЗ. Добавлен динамичный справочник «OBRUErrorResponseErrorCode» для классификации низкоуровневых ошибок.

2 Введение

Настоящий Стандарт содержит описание элементов, которые являются общими для всех API на инициирование перевода денежных средств клиента третьей стороной в валюте Российской Федерации.

2.1 Область применения

Настоящий Стандарт рекомендован к использованию организациями при обмене финансовыми сообщениями, связанными с переводом денежных средств в валюте Российской Федерации.

Настоящий Стандарт предназначен для:

- участников получения информации о банковском счете (банки и их клиенты, а также Сторонние поставщики¹);
- участников перевода денежных средств (банки и их клиенты, а также Сторонние поставщики²);
- разработчиков информационного и программного обеспечения.

Положения настоящего Стандарта применяются на добровольной основе.

По предложениям участников настоящий Стандарт может дополняться ролями и сценариями, принятыми в международной практике.

2.2 Термины и определения

В настоящем Стандарте применяются следующие термины и определения:

Наименование	Описание
API (Application Programming Interface)	Набор процедур, протоколов и инструментов для создания программных приложений. API определяет, как взаимодействуют программные компоненты.
Открытые банковские интерфейсы	Общедоступные интерфейсы прикладного программирования (API), которые предоставляют разработчикам программный доступ к финансовым данным в финансовых сервисах.
Пользователь	Физическое или юридическое лицо, являющееся плательщиком или получателем средств.
Сторонний поставщик	Юридическое лицо, использующее Открытые банковские интерфейсы для доступа к банковскому счету Пользователя в целях предоставления информационных услуг (СПИУ) или для осуществления переводов денежных средств (платежей) (СППУ).
ППУ	Кредитная организация или ее филиал, обслуживающая счет Пользователя и публикующая Открытые банковские интерфейсы.
СПИУ	Юридическое лицо, предоставляющее Пользователю услугу получения информации о банковском счете (счетах) Пользователя.

¹ Получают доступ к информации с согласия владельца счета.

² Получают доступ к счету с согласия владельца счета.

Наименование	Описание
СППУ	Юридическое лицо, предоставляющее Пользователю услугу по инициированию перевода денежных средств.
Среда Открытых банковских интерфейсов	Комплекс стандартов Открытых банковских интерфейсов, управление, системы, процессы, безопасность и процедуры, используемые для поддержки участников.
Участники среды Открытых банковских интерфейсов	Пользователи, банки, поставщики финансовых услуг и разработчики программного обеспечения, которые участвуют в создании и развитии среды Открытых банковских интерфейсов.
Плательщик	Пользователь, осуществляющий перевод денежных средств (либо от имени которого осуществляется перевод денежных средств).
Получатель средств	Пользователь, в пользу которого осуществляется перевод денежных средств.
Стандарт ISO 20022	Международный стандарт обмена электронными сообщениями между организациями финансовой отрасли.
Многофакторная аутентификация Пользователя	Многофакторная аутентификация Пользователя – это аутентификация, которая основана на использовании двух или более элементов, классифицированных как знания, владение и неотъемлемость. Эти пункты являются независимыми, поскольку нарушение одного не угрожает надежности других.
Идемпотентность	Идемпотентность - это свойство объекта или операции при повторном применении операции к объекту давать тот же результат, что и при первом.
Ресурс	Ресурсом является представление любой сущности (например платеж, счет, транзакция) в определенном формате (например JSON). Каждый ресурс идентифицируется посредством постоянного идентификатора, который не меняется при изменении состояния ресурса.
Полезная нагрузка	Часть пакета данных (сообщения) без служебной информации (без заголовка, битов синхронизации и т. п.). Детальное описание структуры полезной нагрузки находится в разделе 4.2.

Таблица 1 - Термины и определения

2.3 Принципы архитектуры

Архитектура среды Открытых банковских интерфейсов соответствует концепции RESTful API³.

Данная концепция была выбрана на основании отзывов участников рынка, а также согласно опыту мировых практик.

Однако, приоритет заключается в том, чтобы API были просты для понимания и просты в использовании. В случаях, когда следование принципам RESTful является сложным, принципы не соблюдаются.

³ Детальное описание подхода можно посмотреть на сайте - <https://restfulapi.net/>.

2.3.1 Идентификаторы

Ресурс REST имеет уникальный идентификатор, который используется для идентификации ресурса. Эти уникальные идентификаторы используются для создания URL-адресов для идентификации и адресации конкретных ресурсов.

2.3.2 Коды статусов

API используют два кода состояния, которые служат двум различным целям:

- Код состояния HTTP отражает результат вызова API (операция HTTP на ресурсе).
- Поле состояния в некоторых полезных нагрузках ресурса отражает состояние ресурсов.

3 Основы

3.1 Роли участников

Среда Открытых банковских интерфейсов определяет следующих участников:

Пользователь – физическое или юридическое лицо, являющееся плательщиком или получателем средств.

Сторонний поставщик – юридическое лицо, использующее Открытые банковские интерфейсы для доступа к банковскому счету Пользователя в целях предоставления информационных услуг (СПИУ) или для осуществления переводов денежных средств (платежей) (СППУ).

ППУ (Поставщик Платежных Услуг) – кредитная организация или ее филиал, обслуживающая счет Пользователя и публикующая Открытые банковские интерфейсы.

Среда Открытых банковских интерфейсов определяет следующие роли участников:

СПИУ (Сторонний Поставщик Информационных Услуг о банковском счете) – юридическое лицо, предоставляющее Пользователю услугу получения информации о банковском счете (счетах) Пользователя.

СППУ (Сторонний Поставщик Платежных Услуг) – юридическое лицо, предоставляющее Пользователю услугу по инициированию перевода денежных средств.

Среда Открытых банковских интерфейсов определяет следующие взаимоотношения между участниками и ролями:

Роль \ Участник	Пользователь	ППУ	Сторонний поставщик
СПИУ	Нет	Да	Да
СППУ	Нет	Да	Да

Таблица 2 - Роли участников

Открытые банковские интерфейсы регламентируют взаимодействие только между следующими участниками:

- **ППУ** предоставляет Открытые банковские интерфейсы для Стороннего поставщика. Получает сообщения запросов через свои Открытые банковские интерфейсы и отправляет соответствующие ответные сообщения Стороннему поставщику.
- **Сторонний поставщик** может получить доступ к счету Пользователя, управляемому ППУ через Открытые банковские интерфейсы, при согласии Пользователя. Сторонний поставщик отправляет сообщения запроса через Открытые банковские интерфейсы ППУ и получает соответствующие ответные сообщения от этого ППУ.

3.2 Кодировка символов

Запросы и ответы API используют кодировку UTF-8. Это кодировка символов по умолчанию для JSON (RFC 7158 - [раздел 8.1](#)).

Однако нисходящая система ППУ может не принимать некоторые символы UTF-8, такие как символы emoji (например, идеограммы и смайлики могут не быть приемлемой ссылкой на платеж). Если ППУ отклоняет сообщение с символом UTF-8, которое не может быть обработано, то ППУ отвечает кодом состояния HTTP 400 (неверный запрос).

3.3 Формат даты

ППУ принимает в запросах все действующие форматы даты стандарта [ISO-8601](#), включая его разрешенные вариации.

Все даты в полезных нагрузках JSON представлены в формате date-time стандарта [ISO 8601](#). Все поля date-time в ответах включают часовой пояс. Например:

```
2019-07-08T11:23:03+00:00
2019-05-03T18:13:23Z
```

Таблица 3 - Поля dateTime с часовым поясом

Все даты в параметрах query имеют формат date-time стандарта ISO 8601 и не включают часовой пояс. Например:

```
2019-07-01T09:23:01
2019-02-07
```

Таблица 4 - Поля dateTime с обрезанием часового пояса

Все даты в заголовках HTTP представлены как полные даты RFC 7231. Пример:

```
Mon, 26 Aug 2019 14:23:51 GMT
```

Таблица 5 - Формат представления полной даты

Все даты в параметрах claims JWT имеют формат number JSON, представляющего количество секунд с 1970-01-01T0:0:0Z, измеренное в GMT до текущей даты (dateTime).

```
//Mon, 26 Aug 2019 14:23:51 GMT
1987646700
```

Таблица 6 - Формат представления полной даты с секундами

3.4 Структура пути URI ресурса

Путь URI соответствует следующей структуре:

- [participant-path-prefix]/open-banking/[version]/[resource]/[resource-id]/[sub-resource]

Структура URI пути состоит из следующих элементов:

- [participant-path-prefix]
Необязательный префикс ППУ.
- open-banking
Постоянное значение "open-banking".
- [version]
Версия API, выраженная в виде /v[major-version].[minor-version]/.
- [resource]/[resource-id]
Наименование ресурса и его идентификатор.
- [sub-resource]
Наименование подресурса (ресурса 2-го уровня).

ППУ использует один и тот же participant-path-prefix и host name для всех своих ресурсов.

Примеры:

<https://bank.ru/oapi-channel/open-banking/v1.1/payments>
<https://bank.ru/oapi-channel/open-banking/v1.1/account-consents>
<https://bank.ru/oapi-channel/open-banking/v1.1/accounts>
<https://bank.ru/oapi-channel/open-banking/v1.1/accounts/1234>
<https://bank.ru/oapi-channel/open-banking/v1.1/accounts/1234/transactions>

3.5 Заголовки сообщений (headers)

3.5.1 Заголовки запросов

Параметр header	Комментарий	POST запрос	GET запрос	DELETE запрос	PUT запрос
x-fapi-auth-date	Время последнего входа Пользователя в систему с TPP. Значение предоставляется в виде HTTP-date, как в разделе 7.1.1.1 [RFC7231] . Например, x-fapi-auth-date: Mon, 26 Aug 2019 12:23:11 GMT	Необязательно	Необязательно	Необязательно	Не используется
x-fapi-customer-ip-address	IP-адрес Пользователя, если Пользователь в данный момент подключен к Стороннему Поставщику (залогинен в приложении Стороннего Поставщика).	Необязательно	Необязательно	Необязательно	Не используется
x-fapi-interaction-id	RFC4122 UID, используемый в качестве идентификатора корреляции. Если необходимо, то ППУ передает обратно значение идентификатора корреляции в заголовке ответа x-fapi-interaction-id.	Обязательно	Обязательно	Обязательно	Обязательно
Authorization	Стандартный заголовок HTTP. Позволяет предоставлять учетные данные серверу авторизации и/или серверу ресурсов в зависимости от типа запрашиваемого ресурса. Для OAuth 2.0 / OIDC включает в себя Basic или Bearer схемы аутентификации.	Обязательно	Обязательно	Обязательно	Обязательно
Content-Type	Стандартный заголовок HTTP. Представляет формат полезной нагрузки в запросе. Устанавливается значение application/json, за исключением конечных точек, которые поддерживают Content-Type, отличный от application/json	Обязательно	Не используется	Не используется	Обязательно

Параметр header	Комментарий	POST запрос	GET запрос	DELETE запрос	PUT запрос
	<p>(например, POST /file-payment-consents/{consentId}/file).</p> <p>Устанавливается значение application/jose+jwe для зашифрованных запросов.</p> <p>Сторонний Поставщик может предоставлять дополнительную информацию.</p> <p>Если установлено другое значение, то ППУ присылает ответ: 415 Unsupported Media Type.</p>				
Ассепт	<p>Стандартный HTTP заголовок, определяющий тип контента, который требуется от сервера.</p> <p>Если Сторонний Поставщик ожидает незашифрованный ответ, то он указывает явно, что только ответ в формате JSON принимается (передавая значение application/json) в качестве заголовка контента для всех конечных точек, которые отвечают в формате JSON.</p> <p>Если Сторонний Поставщик ожидает зашифрованный ответ, то он указывает явно, что принимается только ответ JWT (передавая значение application/jose+jwe) в качестве заголовка контента для всех конечных точек, которые отвечают JSON.</p> <p>Для конечных точек, которые не отвечают в формате JSON (например, GET ../statements/{statementId}/file), ППУ указывает доступные параметры на своем портале для разработчиков.</p> <p>Сторонний Поставщик может предоставлять дополнительную информацию.</p> <p>Если установлено недопустимое значение, то ППУ отвечает: 406 (Not Acceptable).</p> <p>Если значение не указано, по умолчанию используется application/json.</p>	Необязательно	Необязательно	Не используется	Необязательно

Параметр header	Комментарий	POST запрос	GET запрос	DELETE запрос	PUT запрос
x-idempotency-key	Не стандартный HTTP заголовок. Уникальный идентификатор запроса для поддержки идемпотентности. Обязательно для запросов POST к конечным точкам идемпотентного ресурса. Для других запросов не указывается.	Необязательно	Не используется	Не используется	Не используется
x-jws-signature	Указывает, что тело запроса подписано. В документации на ресурсы отдельно определяется, когда это поле в заголовке указывается.	Условно (зависит от API)	Условно (зависит от API)	Условно (зависит от API)	Обязательно
x-customer-user-agent	В заголовке указывается тип устройства (user-agent), который использует Пользователь. Сторонний Поставщик может заполнить это поле значением типа устройства (user-agent), указанным Пользователем. Если Пользователь использует мобильное приложение Стороннего Поставщика, Сторонний Поставщик проверяет, что строка типа устройства (user-agent) отличается от строки типа устройства (user-agent) на основе браузера.	Необязательно	Необязательно	Необязательно	Необязательно

Таблица 7 - Заголовки запросов

Наличие или отсутствие Пользователя определяется через заголовок x-api-customer-ip-address. Если указан IP-адрес Пользователя, то предполагается, что Пользователь присутствует во время взаимодействия.

Последствием этого является, что ППУ полагаются на информацию, предоставленную СППУ.

3.5.2 Заголовки ответов

Параметр header	Комментарий	Обязательность
Content-Type	Стандартный параметр заголовка HTTP. Представляет формат полезной нагрузки, возвращаемой в ответе. ППУ возвращает значение Content-Type равное application/json в качестве заголовка для всех незашифрованных конечных точек.	Обязательно

Параметр header	Комментарий	Обязательность
	ППУ возвращает значение Content-Type равное application/jwe для всех зашифрованных конечных точек.	
Retry-After	Параметр заголовка, указывающий время (в секундах), в течение которого Сторонний Поставщик ожидает перед повторением операции. ППУ следует включать этот заголовок вместе с ответами с кодом состояния HTTP 429 (Too Many Requests).	Необязательно
x-fapi-interaction-id	RFC4122 UID, используемый в качестве идентификатора корреляции. ППУ заполняет параметр заголовка ответа x-fapi-interaction-id значением полученным в соответствующем параметре заголовка запроса или значением UID RFC4122 , если значение не было предоставлено в запросе для отслеживания взаимодействия.	Обязательно
x-jws-signature	Указывает, что тело ответа подписано. В документации на ресурсы отдельно определяется, когда указывается это поле в заголовке.	Условно (зависит от API)

Таблица 8 - Заголовки ответов

3.6 Коды статусов HTTP

Ниже приведены коды ответов HTTP для различных методов HTTP, для всех конечных точек API.

Ситуация	HTTP статус	Комментарий	Для POST	Для GET	Для DELETE	Для PUT
Запрос успешно выполнен	200 OK		нет	да	нет	да
Операция создания выполнена успешно.	201 Created	Результатом операции является создание нового ресурса.	да	нет	нет	нет
Операция удаления успешно завершена.	204 No Content		нет	нет	да	нет
Запрос имеет неверные, отсутствующее или несовместимое тело JSON, параметры URL или поля заголовка.	400 Bad Request	Запрошенная операция не будет выполнена.	да	да	да	да
Заголовок авторизации отсутствует или неверный токен.	401 Unauthorized	Операции было отказано в доступе. Повторная аутентификация Пользователя может привести к созданию	да	да	да	да

Ситуация	HTTP статус	Комментарий	Для POST	Для GET	Для DELETE	Для PUT
		соответствующего токена, который может быть использован.				
Токен имеет неверную область действия или была нарушена политика безопасности.	403 Forbidden	Операции было отказано в доступе. Повторная аутентификация Пользователя может привести к созданию соответствующего токена, который может быть использован.	да	да	да	да
1. Сторонний Поставщик пытается получить ресурс, который указан в спецификации, но не реализован на стороне ППУ (например, ППУ решил не реализовывать конечную точку API статуса для внутренних запланированных платежей). 2. Сторонний Поставщик пытается получить ресурс, который не определен	404 (Not Found)		да	да	да	да
Сторонний Поставщик попытался получить доступ к ресурсу с помощью метода, который не поддерживается.	405 Method Not Allowed		да	да	да	да
Запрос содержал параметр заголовка Ассерта, отличный от разрешенных media types, и набор символов, отличный от UTF-8.	406 Not Acceptable		да	да	да	да
Операция была отклонена, поскольку полезная нагрузка находится в формате, не поддерживаемом этим методом на целевом ресурсе.	415 Unsupported Media Type		да	нет	нет	да

Ситуация	HTTP статус	Комментарий	Для POST	Для GET	Для DELETE	Для PUT
Операция была отклонена, так как слишком много запросов было сделано в течение определенного периода времени.	429 Too Many Requests	<p>ППУ ограничивают запросы, если они сделаны сверх их политики добросовестного использования.</p> <p>ППУ документируют свои политики добросовестного использования на своих порталах для разработчиков.</p> <p>ППУ отвечают этим статусом, если количество запросов в единицу времени было превышено.</p> <p>ППУ следует включать заголовок Retry-After в ответ, указывающий, как долго Сторонний Поставщик ожидает перед повторением операции.</p>	да	да	да	да
Что-то пошло не так на стороне ППУ.	500 Internal Server Error	Операция не удалась.	да	да	да	да
Устаревшая версия сервиса.	503 Service Unavailable	Если API устарел и больше не поддерживается ППУ, его путь URI все еще может быть активным и принимать запросы. В этом контексте рекомендуется вернуть 503 Service Unavailable, чтобы TPP знал, что версия API находится в offline режиме.	да	да	да	да

Таблица 9 - Коды статусов HTTP

ППУ возвращают другие стандартные коды состояния HTTP (например, от шлюзов и других периферийных устройств), как описано в [RFC 7231 - Раздел 6](#).

ППУ отвечают ошибкой в потоке OAuth / OIDC с обязательным выравниванием кодов ошибок, указанными в [разделе 3.1.2.6 OpenID Connect Core Specification](#).

ППУ отвечают на все некорректные запросы общей структурой ошибок Открытых банковских интерфейсов.

3.6.1 400 (Bad Request) или 404 (Not Found)

Если Сторонний Поставщик пытается запросить URL ресурса с идентификатором ресурса, который не существует, то ППУ отвечает 400 (неверный запрос), а не 404 (не найдено).

Например, если Сторонний Поставщик пытается выполнить запрос GET/payment/22289, где 22289 не является действительным paymentId, ППУ отвечает 400.

Если Сторонний Поставщик пытается получить доступ к URL-адресу ресурса, который не определен этими спецификациями (например, GET /card-accounts), то ППУ отвечает 404 (Not Found).

Если ППУ не реализовал конечную точку API, то он отвечает 404 (Не найдено) для запросов к этому URL.

Таблица ниже иллюстрирует некоторые примеры предсказуемого поведения:

Ситуация	Запрос	Ответ
Сторонний Поставщик пытается получить платеж с несуществующим идентификатором paymentId	GET /payments/1001	400 (Bad Request)
Сторонний Поставщик пытается получить ресурс, который указан в спецификации, но не реализован на стороне ППУ. Например, ППУ решил не реализовывать конечную точку API для получения транзакций по счету.	GET /accounts/{accountId}/transactions	404 (Not Found)
Сторонний Поставщик пытается получить ресурс, который не определен	GET /bulk	404 (Not Found)

Таблица 10 - Возможные ситуации для ответов 400 (Bad Request) и 404 (Not Found)

3.6.2 403 (Forbidden)

Когда Сторонний Поставщик пытается получить доступ к ресурсу, к которому у него нет разрешения, ППУ возвращает 403 (Forbidden) с необязательным телом ответа об ошибке.

Ситуация возможна в следующих случаях:

- Сторонний Поставщик использует токен доступа (access token), который не имеет соответствующего scope для доступа к запрошенному ресурсу.
- Сторонний Поставщик попытался получить доступ к ресурсу с идентификатором, к которому у него нет доступа. например, попытка получить доступ к GET /payments/1001, где платежный ресурс с идентификатором 1001 принадлежит другому Стороннему Поставщику.
- Сторонний Поставщик пытается получить доступ к ресурсу транзакции, а у Стороннего Поставщика нет согласия на авторизацию с правами доступа к запрашиваемому ресурсу.
- Сторонний Поставщик пытается получить доступ к ресурсу транзакции, а у Стороннего Поставщика нет согласия на авторизацию для accountId. Например, попытка получить доступ к GET /accounts/2001 или GET /accounts/2001/transactions, когда Пользователь не выбрал accountId 2001 для авторизации.
- Сторонний Поставщик пытается получить доступ к ресурсу, а ППУ решает повторно аутентифицировать Пользователя. ППУ отвечает соответствующим кодом ошибки, чтобы указать, что требуется повторная аутентификация.

3.6.3 401 (Unauthorized)

Когда Сторонний Поставщик использует токен доступа с истекшим сроком, ППУ возвращает 401 (Unauthorized) без какого-либо сообщения об ошибке.

Ситуация возникает, если ППУ завершил срок действия токена доступа по любой из следующих причин:

1. Истек срок действия согласия.
2. Подозрительное использование токена доступа или подозрение в мошенничестве
3. Плановая реализация многофакторной аутентификации.

Эта ошибка потенциально может быть исправлена, если Пользователь повторно пройдет аутентификацию или аутентифицируется с соответствующими разрешениями.

3.6.4 429 (Too Many Requests)

Если Сторонний Поставщик пытается получить доступ к ресурсу слишком часто, то ППУ может вернуть 429 (Too Many Requests). Это нефункциональное требование, и отдельные ППУ определяют метрику запросов в единицу времени.

Ситуация возникает когда:

- Сторонний Поставщик решает реализовать функцию «Статус платежа в реальном времени» для своих пользователей и делает это не корректно, опрашивая конечную точку методом GET.
- Сторонний Поставщик решает использовать конечную точку для разового единичного платежа, как если бы она была конечной точкой пакетной оплаты, и отправляет большое количество запросов на оплату в очень короткий промежуток времени, так что это превышает политику использования ППУ.

3.7 Идемпотентность

Ключ идемпотентности используется для защиты от создания дубликатов ресурсов при использовании метода POST для конечных точек API.

Если для конечной точки API требуется ключ идемпотентности:

- Параметр заголовка `x-idempotency-key`, содержит не более 40 символов. Если длина поля превышает 40 символов, то ППУ отклоняет запрос с кодом состояния 400 (Bad Request).
- Сторонний Поставщик не меняет тело запроса при использовании одинакового ключа `x-idempotency-key`. Если Сторонний Поставщик изменяет тело запроса, то ППУ не меняет конечный ресурс. ППУ рассматривает это как мошенническое действие.
- ППУ обрабатывает запрос как идемпотентный, если он получил запрос с существующим параметром `x-idempotency-key` от того же Стороннего Поставщика в течении 24 часов.
- ППУ не создает новый ресурс для запроса POST, если он определен как идемпотентный запрос.
- ППУ отвечает на запрос текущим статусом ресурса (или статусом максимально близким к текущему, который можно получить в данный момент времени на существующем онлайн канале) и кодом статуса HTTP 201 (Created).
- Сторонний Поставщик не использует ключ идемпотентности при опросе состояния ресурсов.

- ППУ использует подпись сообщения вместе с ключом идемпотентности, чтобы гарантировать, что тело запроса не изменилось.

Если ключ идемпотентности не требуется для конечной точки API, но содержится в запросе, то ППУ игнорирует ключ идемпотентности.

3.8 Фильтрация

ППУ обеспечивает ограниченную поддержку фильтрации для операций GET, которые возвращают множественные записи.

Параметры фильтра всегда разные для конкретного поля (полей) ресурса и следуют правилам/форматам, определенным в справочниках для ресурса.

Для параметров фильтра типа DateTime, значения соответствуют стандарту ISO8601. Если значение поля типа DateTime содержит часовой пояс, то ППУ игнорирует эту информацию.

Предполагается, что значения фильтра относятся к тому же часовому поясу, что и часовой пояс, в котором поддерживается ресурс.

3.9 Нумерация страниц

ППУ предоставляет постраничный ответ для операций GET, которые возвращают множественные записи.

В такой ситуации ППУ:

- Если существует следующая страница записей ресурсов, то ППУ предоставляет ссылку на следующую страницу ресурсов в поле Links.Next ответа. Отсутствие следующей ссылки будет означать, что текущая страница является последней страницей результатов.
- Если предыдущая страница записей ресурсов существует, то ППУ предоставляет ссылку на предыдущую страницу ресурсов в поле Links.Prev ответа. Отсутствие предыдущей ссылки указывает на то, что текущая страница является первой страницей результатов.

Для разбитых на страницы ответов ППУ гарантирует, что количество записей на странице находится в разумных пределах, минимум 25 записей (кроме последней страницы, где больше нет записей) и максимум 1000 записей.

Дополнительно, ППУ предоставляет:

- Ссылку на первую страницу результатов в поле Links.First.
- Ссылку на последнюю страницу результатов в поле Links.Last.
- Общее количество страниц в поле Meta.TotalPages.

ППУ включает «self» ссылку на ресурс в поле Links.Self, как описано в разделе «Ссылки».

Этот стандарт не определяет, каким образом параметры перелистывания страниц передаются ППУ, каждый ППУ может использовать свои собственные механизмы для разбивки ответа.

Если исходный запрос от СППУ включал параметры фильтра, то в ответе возвращаются только те результаты (разбитые на страницы), которые соответствуют фильтру.

3.10 Архивирование

Архивация ресурсов будет определяться для ППУ на основе их внутренних требований.

В дополнение стоит заметить, что:

- ППУ удаляют просроченные идентификаторы согласия (consentId) только через 24 часа после создания.
- ППУ могут архивировать просроченные идентификаторы согласия.

3.11 Дополнительные данные

Ряд ресурсов в спецификации включает в себя раздел для дополнительных данных (Supplementary Data). Данный раздел позволит ППУ принимать или передавать данные, которые не предусмотрены основной структурой ресурса.

Раздел дополнительных данных определяется как пустой объект JSON в данной спецификации.

Если ППУ использует структуру с дополнительными данными (Supplementary Data), то он выкладывает детальное описание структуры у себя на портале для разработчиков.

ППУ не используют структуру Supplementary Data, если добавляемый туда элемент уже существует в текущей версии документа Открытых банковских интерфейсов.

4 Модель данных

4.1 Справочники и перечисления

Спецификации Открытых банковских интерфейсов содержат поля со справочными данными.

Справочники бывают 2-х видов:

- фиксированные;
- гибкие.

В случае фиксированных справочников - все возможные значения будут задаваться статично в Стандарте Открытых банковских интерфейсов.

В случае гибких справочников - начальные возможные значения будут задаваться в Стандарте Открытых банковских интерфейсов, но каждый ППУ может динамично управлять ими, расширяя своими кастомизированными значениями. Кастомизированные значения, которые используют ППУ, могут в будущем включаться в список начальных возможных значений для гибких справочников.

И фиксированные и гибкие справочники находятся в разделе [Справочники и перечисления](#).

4.2 Общая структура полезной нагрузки

В этом разделе приводится обзор структуры верхнего уровня для полезных нагрузок Открытых банковских интерфейсов.

Данные, которые содержатся в разделе "Data", документируются для каждой отдельно взятой конечной точки API.

4.2.1 Структура запроса

Структура верхнего уровня для запросов Открытых банковских интерфейсов имеет следующий вид:

```
{
  "Data": {
    ...
  },
  "Risk": {
    ...
  }
}
```

Таблица 11 - Верхнеуровневая структура сообщений для запросов

Data

Раздел "Data" содержит данные запроса для конкретного запроса API.

Структура этого элемента отличается для каждой конечной точки API.

Risk

Раздел "Risk" содержит индикаторы риска для конкретного запроса API, предоставленного Сторонним поставщиком.

Индикаторы риска, содержащиеся в этом элементе, могут отличаться для каждой конечной точки API.

4.2.2 Структура ответа

Структура верхнего уровня для ответов Открытых банковских интерфейсов имеет следующий вид:

```
{
  "Data": {
    ...
  },
  "Risk": {
    ...
  },
  "Links": {
    ...
  },
  "Meta": {
    ...
  }
}
```

Таблица 12 - Верхнеуровневая структура сообщений для ответов

В соответствии с принципом API RESTful полный ресурс воспроизводится как часть ответа.

В ответ включаются следующие дополнительные разделы высокого уровня:

- Links
- Meta

4.2.3 Структура ответа с ошибками

Структура для ответов с ошибками Открытых банковских интерфейсов имеет следующий вид:

```
{
  "code": "...",
  "id": "...",
  "message": "...",
  "Errors": [
    {
      "errorCode": "...",
      "message": "...",
      "path": "...",
      "url": "..."
    }
  ]
}
```

Таблица 13 - Структура сообщений для ответов с ошибками

UML диаграмма



Рисунок 1 - UML диаграмма для структуры ответов с ошибками

Спецификация данных

Наименование	Кратность	XPath	Подробное описание	Тип данных	Значения	Шаблон
OBRUErrorResponse		OBRUErrorResponse	Массив подробных кодов ошибок, сообщений и URL-адресов к документации для помощи в исправлении.	OBRUErrorResponse		
code	1..1	OBRUErrorResponse/Code	Высокоуровневый текстовый код ошибки, необходимый для классификации.	Max40Text		
id	0..1	OBRUErrorResponse/Id	Уникальный идентификатор ошибки, для целей аудита, в случае неизвестных / не классифицированных ошибок.	Max40Text		
message	1..1	OBRUErrorResponse/Message	Краткое сообщение об ошибке. Например, «что-то не так с предоставленными параметрами запроса».	Max500Text		
Errors	1..n	OBRUErrorResponse/Errors		OBRUError		

Наименование	Краткость	XPath	Подробное описание	Тип данных	Значения	Шаблон
errorCode	1..1	OBRUErrorResponse/Errors/ErrorCode	Низкоуровневое текстовое описание ошибки. Например, RU.SBRF.Field.Missing.	OBRUErrorResponseErrorCode		
message	1..1	OBRUErrorResponse/Errors/Message	Описание ошибки. Например, "Обязательное поле не указано".	Max500Text		
path	0..1	OBRUErrorResponse/Errors/Path	Путь к элементу с ошибкой в JSON объекте. Рекомендуем, но не обязательное поле.	Max500Text		
url	0..1	OBRUErrorResponse/Errors/Url	URL для помощи в устранении проблемы. Также через URL можно предоставлять дополнительную информацию.	xs:anyURI		

Таблица 14 - Детальное описание элементов ответа с ошибками

4.2.4 Необязательные поля

В объектах, где значение для необязательного поля не указано, поле исключается из полезной нагрузки JSON.

В объектах, где поле массива определено как имеющее значения 0..n, поле массива включается в полезную нагрузку с пустым массивом.

```
{
  "name": "", // Неправильно. Поле "Name" нужно исключить из
              // полезной нагрузки.
  "age": 0, // Неправильно. Значение "0" не используется
            // для указания неопределенного возраста.
  "creditorAccount": {}, // Неправильно. Поле "CreditorAccount" нужно
                          // исключить.
  "balances": [] // Правильно. Таким образом передается пустой
                 // массив.
}
```

Таблица 15 - Примеры передачи необязательных полей

4.2.5 Ссылки

Раздел "Links" является обязательным и всегда будет содержать абсолютные URI для связанных ресурсов.

Ссылка "self" является обязательной.

```
"Links": {
  "self": "https://api.bank.ru/open-banking/v3.1/payments/58923"
}
```

Таблица 16 - Пример передачи одинарной ссылки "Self"

При передаче большого количества данных раздел Links может также содержать элементы First, Prev, Next и Last.

```
"Links": {
  "self": "http://rocks.ru/articles?page[number]=3&page[size]=25",
  "first": "http://rocks.ru/articles?page[number]=1&page[size]=25",
  "prev": "http://rocks.ru/articles?page[number]=2&page[size]=25",
  "next": "http://rocks.ru/articles?page[number]=4&page[size]=25",
  "last": "http://rocks.ru/articles?page[number]=6&page[size]=25"
}
```

Таблица 17 - Пример передачи всех элементов раздела Links

4.2.6 Метаданные

Раздел Meta обязателен, но может быть пустым. Необязательный элемент - «TotalPages», указывает на количество передаваемых страниц. Если передается более одной страницы, то элемент «TotalPages» обязательно присутствует.

```
"Meta": {
  "totalPages": 6
}
```

Таблица 18 - Пример передачи раздела Meta

5 Примеры использования

Примеры использования для отдельных API задокументированы на соответствующих страницах.

В этом разделе приведены примеры использования некоторых шаблонов.

5.1 Потоки с нумерацией страниц

Приведенный ниже пример иллюстрирует возможность ППУ возвращать многостраничный ответ.

```
GET /accounts/34566/transactions HTTP/1.1
Authorization: Bearer Az567AOJtyue
x-fapi-auth-date: Mon, 2 Sep 2019 12:33:12 GMT
x-fapi-customer-ip-address: 10.5.412.45
x-fapi-interaction-id: 11bac543-d5de-3446-b687-880a5018434d
Accept: application/json
```

Таблица 19 - Запрос списка транзакций

```
HTTP/1.1 200 OK
x-fapi-interaction-id: 11bac543-d5de-3446-b687-880a5018434d
Content-Type: application/json

{
  "Data": {
    ...
  },
  "Links": {
    "self": "https://bank.ru/open-banking/v3.1/accounts/98765/transactions/",
    "last": "https://bank.ru/open-banking/v3.1/accounts/98765/transactions?pg=6",
    "first": "https://bank.ru/open-banking/v3.1/accounts/98765/transactions/",
    "next": "https://bank.ru/open-banking/v3.1/accounts/98765/transactions?pg=2"
  },
  "Meta": {
    "totalPages": 6,
    "firstAvailableDateTime": "2019-05-03T00:00:00+00:00",
    "lastAvailableDateTime": "2019-12-03T00:00:00+00:00"
  }
}
```

Таблица 20 - Пример передачи всех элементов раздела Links

6 Инициирование переводов денежных средств, спецификация API - v1.2.1

6.1 Введение

В этой спецификации инициирования платежей описываются потоки и полезные данные для инициирования платежа на основании распоряжения о переводе денежных средств.

Описанные здесь конечные точки API позволяют СППУ:

- Создать ресурс согласия для инициирования платежа.
- Подтвердить достаточность средств для осуществления платежа (необязательная конечная точка).
- Подтвердить инициирование платежа и отправить его на обработку.
- Получить статус осуществления платежа.
- Получить статус согласия на проведение платежа.

6.1.1 Принципы проектирования

6.1.1.1 Агностичность к способу платежа

Открытые банковские интерфейсы разработаны таким образом, чтобы не зависеть от базовой схемы оплаты, которая определяет порядок осуществления платежа.

Определения полей и их длина установлены согласно стандарту ISO 20022.

6.1.1.2 Коды статусов

Открытые банковские интерфейсы используют следующие виды кодов статуса:

- Код состояния HTTP отражает результат вызова API (операция HTTP на ресурсе). Передается в заголовке сообщения и отображает техническую составляющую процесса.
- Поле "Status" для сообщения с согласием на проведение платежа отражает статус авторизации согласия Пользователем. Передается в сообщении с полезной нагрузкой и отображает бизнес-составляющую процесса.
- Поле "Status" для ресурса платежа отражает статус инициирования или проведения платежа. Передается в сообщении с полезной нагрузкой и отображает бизнес-составляющую процесса.

6.2 Основы

В данном разделе описываются потоки данных с позиции бизнес-процессов. Детальное описание потоков данных находится в стандарте «Открытые банковские интерфейсы. Безопасность финансовых (банковских) операций. Прикладные программные интерфейсы обеспечения безопасности финансовых сервисов на основе протокола OpenID».

6.2.1 Общее описание процесса

6.2.1.1 Пошаговое описание

Шаг 1: Согласие на инициирование платежа

- Этот поток начинается с согласия Пользователя на совершение платежа. Согласие направляется между Пользователем и СППУ.
- На этом этапе могут быть указаны реквизиты счета Получателя средств.

Шаг 2: Настройка согласия на платеж

- Между СППУ и сервером авторизации ППУ устанавливается защищенный канал связи в рамках стандарта «Открытые банковские интерфейсы. Безопасность финансовых (банковских) операций. Прикладные программные интерфейсы обеспечения безопасности финансовых сервисов на основе протокола OpenID».
- СППУ с помощью потока Client Credentials Grant получает на сервере авторизации ППУ токен доступа (access token).
- Между СППУ и сервером ресурсов ППУ устанавливается защищенный канал связи в рамках стандарта «Открытые банковские интерфейсы. Безопасность финансовых (банковских) операций. Прикладные программные интерфейсы обеспечения безопасности финансовых сервисов на основе протокола OpenID».
- СППУ подключается к ППУ, который обслуживает платежный счет Пользователя и создает новый ресурс согласия на платеж (payment-consent). Это информирует ППУ о том, что один из его Пользователей намерен выполнить платеж. ППУ отвечает идентификатором ресурса согласия на платеж (consentId).
- Этот шаг выполняется с помощью запроса POST к ресурсу согласия на платеж.

Шаг 3: Авторизация согласия

- СППУ запрашивает у Пользователя авторизацию согласия. СППУ выполняет этот запрос используя поток перенаправления (redirection flow):
 - В потоке перенаправления СППУ перенаправляет Пользователя на страницу ППУ.
 - Перенаправление содержит consentId, созданный на предыдущем шаге.
 - Это позволяет ППУ коррелировать согласие на проведение платежа, которое было установлено.
 - ППУ аутентифицирует Пользователя.
 - Пользователь выбирает счет Получателя средств на этом этапе (если счет Получателя средств не был выбран ранее на шаге 1).
 - ППУ обновляет статус ресурса согласия на платеж, фиксируя что согласие было авторизовано.
 - Как только согласие было авторизовано, Пользователь перенаправляется обратно к СППУ с кодом авторизации (authorization-code).
 - Между СППУ и сервером авторизации ППУ устанавливается защищенный канал связи в рамках стандарта «Открытые банковские интерфейсы. Безопасность финансовых (банковских) операций. Прикладные программные интерфейсы обеспечения безопасности финансовых сервисов на основе протокола OpenID».
 - СППУ обменивает на сервере авторизации ППУ код авторизации (authorization-code) на токен доступа (access token).

Шаг 4: Проведение платежа

- Между СППУ и сервером ресурсов ППУ устанавливается защищенный канал связи в рамках стандарта «Открытые банковские интерфейсы. Безопасность финансовых (банковских) операций. Прикладные программные интерфейсы обеспечения безопасности финансовых сервисов на основе протокола OpenID».

- СППУ создает ресурс платежа (payment) для указания того, что распоряжение о переводе средств, созданное на предыдущих шагах, направляется на обработку.
- Это выполняется с помощью запроса POST на создание ресурса платежа (payment).
- ППУ возвращает идентификатор ресурса платежа к СППУ (paymentId).

Шаг 5: Получение статуса согласия / статуса платежа / деталей платежа

- Между СППУ и сервером ресурсов ППУ устанавливается защищенный канал связи в рамках стандарта «Открытые банковские интерфейсы. Безопасность финансовых (банковских) операций. Прикладные программные интерфейсы обеспечения безопасности финансовых сервисов на основе протокола OpenID».
- СППУ проверяет статус согласия на проведение платежа (с consentId) или статус ресурса платежа (с paymentId), или детали платежа (с paymentId).
- Это выполняется с помощью запроса GET к ресурсу согласия на проведение платежа (**payment-consent**) или ресурсу платежа (**payment**).

6.2.1.2 Диаграмма последовательности

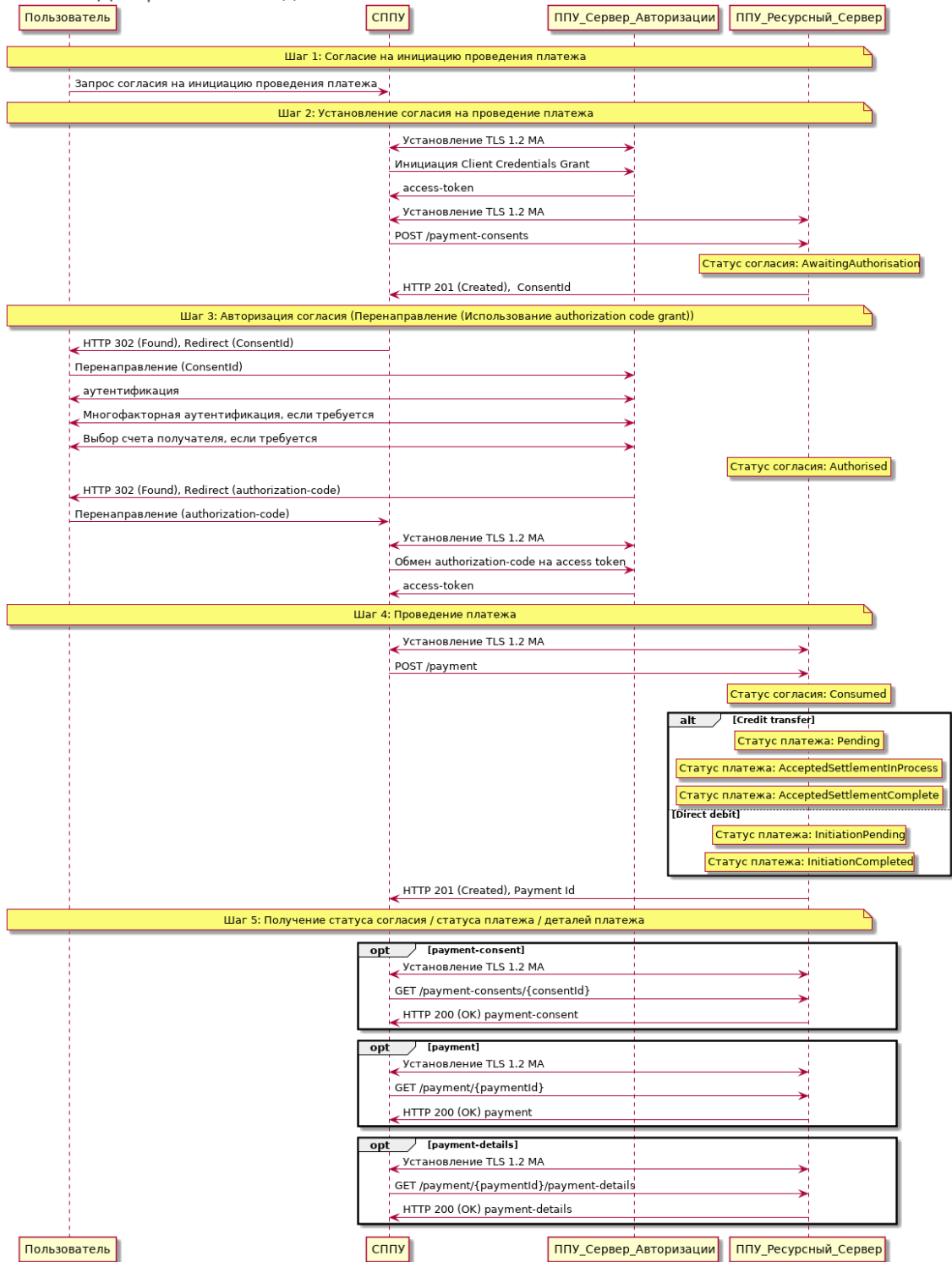


Рисунок 2 - Диаграмма последовательности для инициирования платежа

6.2.2 Ограничения при инициировании платежей

Стандарт не предусматривает единого набора ограничений для типов платежей, которые могут поддерживаться Открытыми банковскими интерфейсами.

Например:

- Максимально допустимое значение InstructedAmount.
- Максимальное значение даты в будущем для запланированного платежа.

Каждый ППУ определяет соответствующие ограничения, которые они поддерживают, основываясь на их индивидуальных методах, стандартах и ограничениях. Эти ограничения документируются на порталах ППУ для разработчиков.

ППУ отклоняет согласие на платеж, если ППУ не может обработать запрос.

6.2.2.1 Использование элемента CutOffDateTime

ППУ может вернуть конкретный CutOffDateTime при ответе на запрос о согласии на проведение платежа.

ППУ описывает поведение квитанции об оплате до и после CutOffDateTime для исполнения распоряжения о переводе средств.

Два варианта развития событий:

- Отклонить распоряжение о переводе средств (и шаги, связанные с созданием данного распоряжения), если оно получено после соответствующего CutOffDateTime.
- Принять распоряжение о переводе средств (и шаги, связанные с созданием данного распоряжения), если оно получено после соответствующего CutOffDateTime.

6.2.2.1.1 Отклонение распоряжения о переводе средств

В этом случае выполнение платежа явно определено для СППУ и для Пользователя:

- ППУ отклоняет согласие на исполнение распоряжения о переводе средств, если истек CutOffDateTime для определенного типа распоряжения.
- ППУ отклоняет запрос авторизации согласия на осуществление платежа, если авторизуемый ресурс согласия связан с истекшим CutoffDateTime. ППУ не выдает токен доступа в такой ситуации. ППУ устанавливает статус ресурса согласия на платеж в значение «Отклонено».
- ППУ отклоняет ресурс платежа, если CutOffDateTime для определенного ресурса платежа был установлен и истек.
- СППУ проверяет, что авторизация согласия Пользователя завершена и ресурс платежа создан до того, как истечет CutOffDateTime.

6.2.2.1.2 Принятие распоряжения о переводе средств

В этом случае выполнение платежа не является определенным для СППУ и Пользователя и платеж будет выполнен в следующий доступный рабочий день.

- ППУ принимает согласие на проведение платежа, если истек CutOffDateTime для определенного типа платежа.
- ППУ принимает запрос авторизации согласия на проведения платежа, когда дата CutoffDateTime истекла для ресурса согласия.
- ППУ принимает ресурс платежа, если CutOffDateTime для платежа был установлен и истек.
- ППУ обновляет согласие на проведение платежа или ресурс платежа с помощью CutOffDateTime, ExpectedExecutionDateTime и ExpectedSettlementDateTime, чтобы сообщить об ожидаемом поведении при исполнении распоряжения, если CutOffDateTime истек.

6.2.3 Сохранение обратной совместимости между мажорными версиями

В этом разделе дается обзор принципов сохранения обратной совместимости между версиями Открытых банковских интерфейсов для API инициирования платежей. Он применяется ко всем ресурсам согласия и ресурсам платежей, указанных в разделе «Конечные точки».

6.2.3.1 Ресурс согласия на проведение платежа

6.2.3.1.1 POST

- СППУ не создает ресурс согласия на проведение платежа с идентификатором `consentId` для более новой версии и не использует его для создания ресурса платежа предыдущей версии
 - Например, ресурс с идентификатором `consentId` созданный в версии 2 не используется для создания ресурса платежа версии 1
- СППУ не создает ресурс согласия на проведение платежа с идентификатором `consentId` для предыдущей версии и не использует его для создания ресурса платежа более новой версии
 - Например, ресурс с идентификатором `consentId` версии 1 не используется для создания ресурса платежа версии 2

6.2.3.1.2 GET

- СППУ не получает доступ к ресурсу согласия на проведение платежа с идентификатором `consentId`, созданному в более новой версии, через конечную точку предыдущей версии
- Поскольку для платежей используется краткосрочное согласие, то ППУ может давать или не давать доступ к ресурсу согласия с идентификатором `consentId` с разных версий конечных точек

6.2.3.2 Подтверждение наличия денежных средств

6.2.3.2.1 GET

СППУ не подтверждает наличие денежных средств, используя `consentId` согласия на проведение платежа, созданное в другой версии.

- Например, ресурс с идентификатором `consentId` созданный во 2-ой версии не используется для подтверждения денежных средств через конечную точку 1-ой версии.

6.2.3.3 Ресурс платежа

6.2.3.3.1 POST

- СППУ использует ресурс согласия на проведение платежа (`consentId`) такой же версии, как и версия ресурса проведения платежа (`paymentId`).
- ППУ не позволяет СППУ использовать `consentId` из предыдущей версии для инициирования платежа в более новой версии, и наоборот.

6.2.3.3.2 GET

- СППУ обращается к онлайн-порталу для разработчиков ППУ для получения рекомендаций по доступности ресурса платежа в более новой версии.
- СППУ не получает доступ к ресурсам платежей более новой версии через конечную точку более старой версии:

- ППУ документирует доступность ресурса платежа в более новой версии на онлайн-портале разработчиков ППУ.
- ППУ разрешает доступ к ресурсу платежа, созданному в предыдущей версии, на конечной точке более новой версии (в зависимости от требований ППУ по хранению данных), подробнее требования будут описаны в следующих версиях спецификации.

6.3 Конечные точки

В этом разделе приведен список доступных конечных точек API для выполнения потоков платежей. Если реализована конечная точка POST, конечная точка GET также реализовывается.

Ссылка	Ресурс	Конечная точка
Внутренние переводы денежных средств, спецификация API - v1.2.1	payment-consents	POST /payment-consents
	payment-consents	GET /payment-consents/{consentId}
	payments	POST /payments
	payments	GET /payments/{paymentId}
	payment-details	GET /payments/{paymentId}/payment-details

Таблица 21 - Конечные точки для инициирования платежа

6.4 Безопасность и контроль доступа

6.4.1 Scopes

Токен доступа, требуемый для доступа к платежным API, имеет, как минимум, следующий scope:

```
payments: Возможность проведения платежа
```

Таблица 22 - Scope для инициирования платежа

6.4.2 Grants Types

СППУ используют поток client credentials grant для получения токена доступа на создание ресурса согласия на проведение платежа (выполнения запроса POST к соответствующей конечной точке). В спецификации этот тип grant называется «Client Credentials».

СППУ используют поток authorization code grant (перенаправление или разъединенный поток) для получения токена доступа на создание ресурса платежа (выполнения запроса POST к соответствующей конечной точке). Этот токен также может использоваться для подтверждения наличия денежных средств. В спецификации этот тип grant называется «Authorization Code».

СППУ используют client credentials grant для получения токена доступа на выполнение запросов GET (за исключением подтверждения наличия денежных средств).

6.4.3 Авторизация согласия

СППУ начинает выполнение платежа с создания ресурса согласия на проведение платежа (payment-consent) с помощью операции POST. Этот ресурс определяет

разрешения (permissions), которые присылает СППУ от имени Пользователя. На начальном этапе согласие не авторизовано, поскольку ППУ еще не верифицировал разрешения во взаимодействии непосредственно с самим Пользователем.

ППУ отвечает сообщением, которое содержит идентификатор ресурса согласия consentId. Далее, этот идентификатор используется при инициации потока authorization code grant, который нужен для подтверждения Пользователем разрешений.

Во время authorization code grant:

- ППУ аутентифицирует Пользователя.
- ППУ направляет согласие, полученное от СППУ, обратно Пользователю для непосредственной авторизации (Пользователь подтверждает то, что давал такие разрешения СППУ, но теперь на странице банка). Пользователь может принять или отклонить согласие только целиком (нет выбора параметров).
- Если в согласии не был указан счет Плательщика, то ППУ предоставляет Пользователю список счетов на выбор.

Как только эти шаги выполнены, согласие авторизованно Пользователем.

6.4.3.1 Множественная авторизация

В контексте множественной авторизации те же этапы авторизации согласия выполняются для первого Пользователя, чтобы авторизовать или подготовить согласие на проведение платежа.

Для ресурса согласия на проведение платежа:

- СППУ запрашивает AuthorisationType для платежа (Single, Any или Multiple). Если значение не указано, то ППУ будет интерпретировать AuthorisationType как "Any".
- СППУ запрашивает CompletionDateTime для завершения авторизации платежа. Если значение не указано, то ППУ будет интерпретировать значение CompletionDateTime как неограниченное.
- ППУ отклоняет согласие на проведение платежа, если запрашиваемый СППУ тип AuthorisationType не совпадает с DebtorAccount в запросе.
- ППУ устанавливает статус согласия платежа на Rejected, если запрашиваемый СППУ тип AuthorisationType не может быть удовлетворен после аутентификации Пользователя:
 - ППУ отвечает ошибкой OAuth, указанной как invalid_request и error_description, содержащей соответствующее описание.
- ППУ ограничивает выбор DebtorAccount (в онлайн-канале ППУ) счетами, которые соответствуют типу AuthorisationType, запрошенному СППУ.

Для ресурса платежа:

- ППУ отвечает объектом MultiAuthorisation, если распоряжение о переводе средств требует множественной авторизации. Объект MultiAuthorisation указывает СППУ, что распоряжение о переводе средств требует множественной авторизации.
- ППУ заполняет поле Status объекта MultiAuthorisation значениями, соответствующими множественной авторизации.
- ППУ может заполнить объект MultiAuthorisation дополнительно необязательными полями для множественной авторизации:
 - Количество требуемых авторизаций (общее количество, которое необходимо с начала множественной авторизации).
 - Количество выполненных авторизаций (завершенных на данный момент).
 - Дата и время последнего обновления авторизации.

- Дата и время, когда процесс авторизации завершается.

6.4.3.2 Условия возникновения ошибок

Если Пользователь не завершает авторизацию согласия успешно (например, если пользователь не прошел аутентификацию успешно), поток authorization code grant заканчивается перенаправлением на сторону СППУ с ответом об ошибке, как описано в Разделе 4.1.2.1 RFC 6749. Пользователь перенаправляется на сторону СППУ с параметром ошибки, указывающим на возникшую ситуацию.

6.4.3.3 Отзыв согласия

Пользователь не может отозвать согласие на проведение платежа после его авторизации.

6.4.3.4 Изменения выбранного счета

Для согласия на проведение платежа выбранный счет Плательщика не может быть изменен после того, как согласие было авторизовано.

6.4.3.5 Повторная аутентификация

Согласие на проведение платежа является краткосрочным согласием и не может быть повторно аутентифицировано Пользователем.

6.4.4 Информация по оценке рисков

Из мирового опыта была взята практика оценки рисков для платежей, инициированных через API платежей.

Информация по оценке рисков располагается в:

- FAPI HTTP заголовках. Это определено в [разделе 6.3 спецификации FAPI](#) и в разделе заголовков выше.
- Дополнительные поля, определенные отраслью как проблемы безопасности бизнес-логики, будут переданы в разделе "Risk" полезной нагрузки в объекте JSON.

Это набор дополнительных полей в разделе риска полезной нагрузки для v1.0.0, который будет определен на стороне СППУ:

- PaymentContextCode.
- MerchantCategoryCode.
- MerchantCustomerIdentification.
- DeliveryAddress.

PaymentContextCode описывает контекст платежа и может иметь следующие значения:

- BillPayment.
- EcommerceGoods.
- EcommerceServices.
- Other.
- PartyToParty.

В платежах EcommerceGoods и EcommerceServices могут быть заполнены MerchantCategoryCode и MerchantCustomerIdentification (рекомендовано заполнять). В платежах EcommerceGoods также рекомендуется заполнять адрес доставки.

Эти поля описаны в разделе «Модель данных».

6.5 Модель данных

6.5.1 Повторно используемые объекты

6.5.1.1 Risk

В этом разделе описывается класс Risk, который используется в ресурсах согласия на проведение платежа и непосредственно платежа.

6.5.1.1.1 Диаграмма UML



Рисунок 3 - Диаграмма объекта Risk

6.5.1.1.2 Состав данных объекта

Наименование	Кратность	Путь	Описание	Тип	Значения	Шаблон
Risk		Risk	Раздел "Risk" отправляется инициатором в ППУ. Он используется для указания дополнительных деталей для оценки рисков для платежей.	RiskComplexType		
paymentContextCode	0..1	Risk/paymentContextCode	Контекст проведения платежа.	PaymentContextStaticType	BillPayment EcommerceGoods EcommerceServices Other PartyToParty	
merchantCategoryCode	0..1	Risk/merchantCategoryCode	Код категории соответствует ISO 18245 и относится к типу услуг или товаров, которые продавец предоставляет для транзакции.	Min3Max4Text		
merchantCustomerIdentification	0..1	Risk/merchantCustomerIdentification	Уникальный идентификатор покупателя,	Max70Text		

Наименование	Кратность	Путь	Описание	Тип	Значения	Шаблон
			который продавец присвоил Пользователю.			
DeliveryAddress	0..1	Risk/DeliveryAddress	Информация, которая находит и идентифицирует определенный адрес, как определено почтовыми службами или в произвольном текстовом формате.	PostalAddress		
addressLine	0..2	Risk/DeliveryAddress/addressLine[]	Информация, которая определяет и идентифицирует определенный адрес, как это определено почтовыми службами, в форме свободного текста.	Max70Text		

Наименование	Кратность	Путь	Описание	Тип	Значения	Шаблон
streetName	0..1	Risk/DeliveryAddress/streetName	Наименование улицы или проспекта.	Max70Text		
buildingNumber	0..1	Risk/DeliveryAddress/buildingNumber	Номер, который определяет положение здания на улице.	Max16Text		
postCode	0..1	Risk/DeliveryAddress/postCode	Идентификатор, состоящий из группы букв и/или цифр, которые добавляются к почтовому адресу для упрощения сортировки почты.	Max16Text		
townName	1..1	Risk/DeliveryAddress/townName	Наименование застроенного пространства с определенными границами и органами местного самоуправления.	Max35Text		
countrySubDivision	0..2	Risk/DeliveryAddress/countrySubDivision[]	Определяет часть страны,	Max35Text		

Наименование	Кратность	Путь	Описание	Тип	Значения	Шаблон
			например, край, область, республика.			
country	1..1	Risk/DeliveryAddress/country	Государство.	CountryCode		^[A-Z]{2,2}\$

Таблица 23 - Состав данных объекта Risk

6.5.1.2 Charge

В этом разделе описывается класс Charge, который используется в полезных нагрузках ответа в ресурсах согласия на проведение платежа и платежа.

6.5.1.2.1 Диаграмма UML

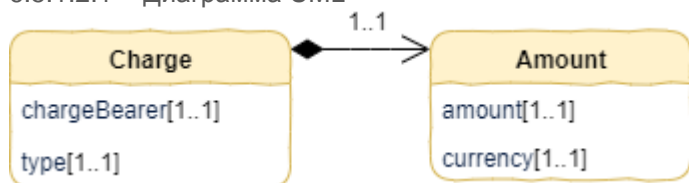


Рисунок 4 - Диаграмма объекта Charge

6.5.1.2.2 Состав данных объекта

Наименование	Кратность	Путь	Описание	Тип	Значения	Шаблон
Charge		Charge	Набор элементов, используемый для предоставления деталей о плате за инициацию платежа. Сколько будет стоить для Пользователя проведение платежа.	ChargeComplexType		

Наименование	Кратность	Путь	Описание	Тип	Значения	Шаблон
chargeBearer	1..1	Charge/chargeBearer	Указывается, какая сторона/стороны будут нести (оплачивать) расходы, связанные с обработкой платежной инструкции.	ChargeBearerType1Code	CRED DEBT SLEV SHAR	
type	1..1	Charge/type	Тип комиссии за проведение платежа.	ExternalChargeType1Code	BRKF BTCH COMM SUMM	
Amount	1..1	Charge/Amount	Сумма комиссии за проведение платежа.	ActiveOrHistoricCurrencyAndAmount		
amount	1..1	Charge/Amount/amount	Сумма комиссии.	OBAActiveCurrencyAndAmount_SimpleType		^d{1,13}\.d{1,5}\$
currency	1..1	Charge/Amount/currency	Код валюты, соответствующий международному стандарту ISO 4217.	ActiveOrHistoricCurrencyCode		^[A-Z]{3,3}\$

Таблица 24 - Состав данных объекта Charge

6.5.1.3 Authorisation

В этом разделе описывается класс Authorisation, который используется в полезных нагрузках запроса и ответа к ресурсу согласия на проведение платежа.

6.5.1.3.1 Диаграмма UML

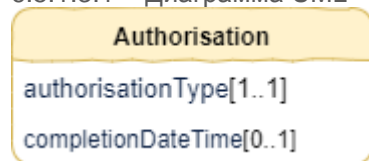


Рисунок 5 - Диаграмма объекта Authorisation

6.5.1.3.2 Состав данных объекта

Наименование	Кратность	Путь	Описание	Тип	Значения	Шаблон
Authorisation		Authorisation	Запрос типа авторизации от СППУ.	AuthorisationComplexType		
authorisationType	1..1	Authorisation/authorisationType	Тип запрашиваемого потока авторизации.	AuthorisationStaticType	Any Single Multiple	
completionDateTime	0..1	Authorisation/completionDateTime	Дата и время, когда запрашиваемый поток авторизации завершается.	ISODateTime		

Таблица 25 - Состав данных объекта Authorisation

6.5.1.4 MultiAuthorisation

В этом разделе описывается класс MultiAuthorisation, который используется в полезных нагрузках ответов ресурсов платежей.

6.5.1.4.1 Диаграмма UML

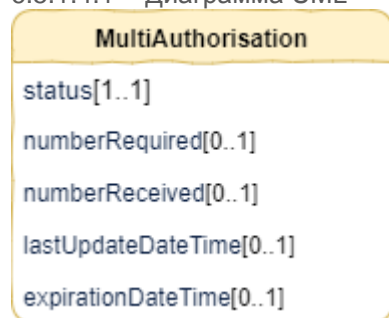


Рисунок 6 - Диаграмма объекта MultiAuthorisation

6.5.1.4.2 Состав данных объекта

Наименование	Кратность	Путь	Описание	Тип	Значения	Шаблон
MultiAuthorisation		MultiAuthorisation	Ответ потока множественной авторизации от ППУ.	MultiAuthorisationComplexType		
status	1..1	MultiAuthorisation/status	Состояние потока авторизации.	MultiAuthorisationStatusStaticType	Authorised AwaitingFurtherAuthorisation Rejected	
numberRequired	0..1	MultiAuthorisation/numberRequired	Количество авторизаций, необходимых для инициирования проведения платежа (общее количество, необходимое при старте мультиавторизации).	Number		
numberReceived	0..1	MultiAuthorisation/numberReceived	Количество полученных авторизаций.	Number		
lastUpdateDateTime	0..1	MultiAuthorisation/lastUpdateDateTime	Дата и время последнего обновления авторизационного потока.	ISODatetime		

Наименование	Кратность	Путь	Описание	Тип	Значения	Шаблон
expirationDateTimе	0..1	MultiAuthorisation/expirationDateTime	Дата и время, когда поток авторизации завершается.	ISODateTime		

Таблица 26 - Состав данных объекта MultiAuthorisation

6.5.1.5 PaymentDetails

В этом разделе описывается класс PaymentDetails, который используется в полезных нагрузках ответов подресурсов деталей платежа.

6.5.1.5.1 Диаграмма UML

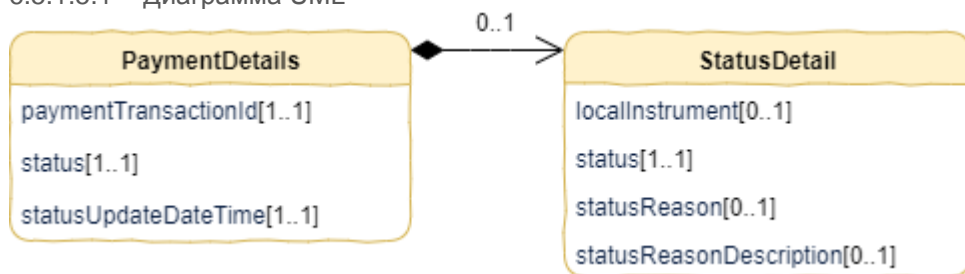


Рисунок 7 - Диаграмма объекта PaymentDetails

6.5.1.5.2 Состав данных объекта

Наименование	Кратность	Путь	Описание	Тип	Значения	Шаблон
PaymentDetails		PaymentDetails	Детали статуса оплаты.	PaymentDetailsComplexType		
paymentTransactionId	1..1	PaymentDetails/paymentTransactionId	Уникальный идентификатор транзакции	Max210Text		

Наименование	Кратность	Путь	Описание	Тип	Значения	Шаблон
			внутри обслуживающего учреждения. Этот идентификатор является уникальным и неизменным.			
status	1..1	PaymentDetails/status	Статус платежа, назначенный администратором транзакции.	ExternalPaymentTransactionStatus1 Code	ACCC ACCP ACFC ACSC ACSP ACTC ACWC ACWP CANC PATC PDNG RCVD RJCT	
statusUpdateDateTime	1..1	PaymentDetails/statusUpdateDateTime	Дата и время, когда статус был присвоен переводу.	ISODateTime		
StatusDetail	0..1	PaymentDetails/StatusDetail	Детали статуса оплаты.	PaymentStatusDetailComplexType		

Наименование	Кратность	Путь	Описание	Тип	Значения	Шаблон
localInstrument	0..1	PaymentDetails/StatusDetail/localInstrument	<p>Специальный инструмент, используемый сообществом.</p> <p>Использование: Этот элемент используется для указания местного инструмента, варианта локального клиринга и/или для дополнительного описания сервиса или уровня обслуживания.</p>	ExternalLocalInstrument1Code		
status	1..1	PaymentDetails/StatusDetail/status	Статус платежа, назначенный администратором транзакции.	Max128Text		

Наименование	Кратность	Путь	Описание	Тип	Значения	Шаблон
statusReason	0..1	PaymentDetails/StatusDetail/statusReason	Код причины статуса платежа.	TransactionIndividualStatusReasonStaticType	Cancelled PendingFailingSettlement PendingSettlement Proprietary ProprietaryRejection Suspended Unmatched	
statusReasonDescription	0..1	PaymentDetails/StatusDetail/statusReasonDescription	Детальное описание причины статуса платежа.	Max256Text		

Таблица 27 - Состав данных объекта PaymentDetails

6.5.1.6 SCASupportData

В этом разделе описывается класс SCASupportData, который используется во всех ресурсах запроса согласия на проведение платежа, что позволяет СППУ предоставлять вспомогательные данные при запросе ППУ для исключения SCA.

6.5.1.6.1 Диаграмма UML

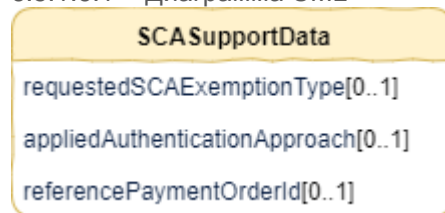


Рисунок 8 - Диаграмма объекта SCASupportData

6.5.1.6.2 Состав данных объекта

Наименование	Кратность	Путь	Описание	Тип	Значения	Шаблон
SCASupportData		SCASupportData	Вспомогательные данные, предоставленные СППУ, при запросе SCA.	SCASupportData		
requestedSCAExemptionType	0..1	SCASupportData/requestedSCAExemptionType	Поле позволяет СППУ запрашивать конкретное исключение SCA для инициирования платежа.	SCAExemptionTypeStaticType	BillPayment ContactlessTravel EcommerceGoods EcommerceServices Kiosk Parking PartyToParty	
appliedAuthenticationApproach	0..1	SCASupportData/appliedAuthenticationApproach	Поле показывает, подвергался ли Пользователь SCA, выполняемый СППУ.	AppliedAuthenticationApproachStaticType	CA SCA	
referencePaymentOrderId	0..1	SCASupportData/referencePaymentOrderId	Использование: если платеж повторяется,	Max128Text		

Наименование	Кратнос ть	Путь	Описание	Тип	Значения	Шабл он
			то идентификато р транзакции предыдущего вхождения платежа позволяет ППУ убедиться, что СППУ, сумма и получатель платежа совпадают с предыдущим вхождением.			

Таблица 28 - Состав данных объекта SCASupportData

6.5.2 Поля для идентификаций

В этом разделе описываются идентификаторы, используемые в потоках API платежей, направление потока и как они используются.

Создание	Идентификатор	Бизнес описание
Продавец/СППУ Отправляется в полезной нагрузке API	endToEndIdentification	Уникальная идентификация, присвоенная иницирующей стороной для однозначной идентификации операции. Эта идентификация передается без изменений по всей цепочке от начала до конца. Использование: Сквозной идентификатор может использоваться для выверки расчетов для связки задач, относящихся к операции. Она может включаться в несколько сообщений, относящихся к операции.
Продавец/СППУ Отправляется в полезной нагрузке API	instructionIdentification	СППУ генерирует InstructionIdentification, который является уникальным идентификатором транзакции, и передает его в ППУ (это обязательно), но это не идет дальше в потоке платежей. Уникальная идентификация, присвоенная инструктирующей стороной для проинструктированной стороны, для однозначной идентификации инструкции. Использование: Идентификация инструкции является сквозным идентификатором, который может использоваться между инструктирующей стороной и проинструктированной стороной для ссылки на индивидуальную инструкцию. Она может включаться в несколько сообщений, относящихся к инструкции. Ожидается, что это уникально на неопределенный срок для всех периодов времени. СППУ может гарантировать, что это неопределенно уникально, включив в поле элемент даты или даты и времени, или вставив уникальный идентификатор.
Продавец/СППУ Отправляется в полезной нагрузке API	remittanceInformation	RemittanceInformation - это информация о платеже, которую согласовывает Получатель средств, для обеспечения квитанции с позициями, которые должен обеспечить перевод, например, оплата по коммерческим счетам, по которым ведется учет в системе ожидаемых поступлений.
ППУ / API система	consentId	Уникальная идентификация, назначенная ППУ для однозначного определения ресурса согласия на проведение платежа.
ППУ / API система	paymentId	Уникальная идентификация, назначенная ППУ для однозначного определения платежа (ресурса платежа).

Создание	Идентификатор	Бизнес описание
		<ul style="list-style-type: none"> paymentId
ППУ / Способ платежа	scheme Payment ID	Создается ППУ для однозначной идентификации способа платежа при его проведении.

Таблица 29 - Идентификаторы, используемые в потоках API платежей

В приведенных ниже таблицах указан участник (эктор), который изначально создает каждый из идентификаторов сообщений, а также их передача и видимость другим участникам.

Эти потоки являются ориентировочными и будут зависеть от того, какие платежные схемы могут поддерживать.

Приняты следующие обозначения:

+ - определяет участника, который создает идентификатор

>> - направление прямого потока

<< - направление обратного потока

6.5.2.1 Поток продавца

Идентификатор	Пользователь	Продавец	СППУ	ППУ	Платежная схема	Бенифициар
endToEndIdentification		+	>>	>>	>>	>>
remittanceInformation		+	>>	>>	>>	>>
instructionIdentification			+	>>		
consentId			<<	+		
payment Id			<<	+		
scheme Payment ID				+	>>	>>

Таблица 30 - Поток продавца

6.5.2.2 Поток между пользователями

Идентификатор	Пользователь	Продавец	СППУ	ППУ	Платежная схема	Бенифициар
endToEndIdentification			+	>>	>>	>>
remittanceInformation	+		>>	>>	>>	>>
instructionIdentification			+	>>		
consentId			<<	+		
payment Order Id			<<	+		
scheme Payment ID				+	>>	>>

Таблица 31 - Поток между Пользователями

6.5.3 Виды платежей

Каждый из типов платежей задокументирован на подстраницах данной спецификации. Каждый тип платежа в документации содержит:

- Конечные точки
 - Конечные точки API доступные для ресурса.
- Модель данных
 - Определение ресурса
 - Диаграмму UML
 - Разрешения, связанные с доступом к ресурсу
 - Спецификация данных, которая определяет поля, повторно используемые классы, обязательные и необязательные поля
- Примеры использования

6.5.4 Справочники

PaymentContextStaticTypeAcceptedSettlementCompleted

№	Значение	Описание
1	BillPayment	Контекстом инициирования платежа является оплата по счету.
2	EcommerceGoods	Контекст инициирования платежа относится к товарам через канал электронной коммерции.
3	EcommerceServices	Контекст инициирования платежа относится к услугам через канал электронной коммерции.
4	PartyToParty	Контекстом инициирования платежа является перевод между двумя сторонами.
5	Other	Контекст инициирования платежа относится к другому типу.

Таблица 32 - Контекст инициирования платежа

TransactionStatusStaticType

№	Значение	Описание
1	AcceptedSettlementCompleted	Расчет по счету Плательщика завершен. СППУ не используют этот статус в качестве подтверждения завершения расчетов по счету Получателя средств.
2	AcceptedSettlementInProgress	Все предыдущие проверки, такие как техническая проверка и профиль клиента, были успешными, и поэтому запрос об инициировании платежа был принят к исполнению.
3	Pending	Ожидается инициирование платежа или отдельной транзакции, включенной в инициирование платежа.
4	Rejected	Инициирование платежа или отдельная транзакция, включенная в инициирование платежа, были отклонены.

№	Значение	Описание
5	AcceptedWithoutPosting	Платежная инструкция, включенная в перевод, принимается без зачисления на счет Получателя средств.
6	AcceptedCreditSettlementCompleted	Расчет по счету Получателя средств завершен.

Таблица 33 - Статус транзакции

ConsentStatusStaticType

№	Значение	Описание
1	AwaitingAuthorisation	Ресурс согласия на проведение платежа ожидает авторизации пользователя.
2	Rejected	Ресурс согласия на проведение платежа был отклонен.
3	Authorised	Ресурс согласия на проведение платежа был успешно авторизован.
4	Consumed	Ресурс согласия на проведение платежа был успешно задействован.

Таблица 34 - Статус согласия на проведение платежа

AuthorisationStaticType

№	Значение	Описание
1	Any	Любой тип авторизации запрашивается.
2	Multiple	Требуется несколько типов авторизации.
3	Single	Запрашивается один тип авторизации.

Таблица 35 - Тип авторизации

6.5.4.1 Справочники ISO

Следующие справочники ISO используются в API инициирования платежей.

№	Значение	Описание
1	ActiveOrHistoricCurrencyCode	https://www.iso20022.org/external_code_list.page
2	CountryCode	Стандарт ГОСТ 7.67.2003 (ИСО 3166-1:1997), alpha-2 code - http://docs.cntd.ru/document/1200035671 . Таблица 1, двухзначный буквенный код на латинице.
3	Min3Max4Text	https://www.iso.org/standard/33365.html

Таблица 36 - Международные и общероссийские справочники

ChargeBearerType1Code (ISO 20022)

№	Значение	Описание
1	CRED	Все расходы по сделке несет Получатель средств.
2	DEBT	Все расходы по сделке несет Плательщик.
3	SLEV	Плата взимается в соответствии с правилами, согласованными на уровне обслуживания и / или схеме.

№	Значение	Описание
4	SHAR	В контексте перевода денежных средств по инициативе Плательщика означает, что расходы по сделке на стороне отправителя несет Плательщик, а расходы по сделке на стороне получателя несет Получатель средств. В контексте прямого дебета означает, что расходы по сделке на стороне отправителя несет Получатель средств, а расходы по сделке на стороне получателя несет Плательщик.

Таблица 37 - Расходы по операции

6.6 Внутренние переводы денежных средств, спецификация API - v1.2.1

6.6.1 Конечные точки

Ресурс	Метод HTTP	Конечная точка	Scope	Grand type	Подписание	Ключ идемпотентности	Объект запроса	Объект ответа
payment-consents	POST	POST /payment-consents	payments	Client Credentials	Подписывается запрос Подписывается ответ	Да	ConsentRequest	ConsentResponse
payment-consents	GET	GET /payment-consents/{consentId}	payments	Client Credentials	Подписывается ответ	Нет		ConsentResponse
payments	POST	POST /payments	payments	Authorization Code	Подписывается запрос Подписывается ответ	Да	PaymentRequest	PaymentResponse
payments	GET	GET /payments/{paymentId}	payments	Client Credentials	Подписывается ответ	Нет		PaymentResponse

Таблица 38 - Конечные точки для ресурса платежа

6.6.1.1 POST /payment-consents

```
POST /payment-consents
```

Таблица 39 - Создание ресурса согласия на проведение платежа

Данная конечная точка позволяет выполнять следующее взаимодействие:

- Конечная точка позволяет с помощью метода POST создавать на стороне ППУ ресурс согласия на проведение платежа (payment-consent) без предварительной идентификации Пользователя.
- Полезная нагрузка запроса может не содержать отдельных необязательных полей, например, она может не содержать информации о том, какой именно счет Пользователя, открытый в банке (ППУ) будет в итоге использован для списания денежных средств.
- Конечная точка позволяет СППУ отправлять согласие (между Пользователем и СППУ) в качестве информации для ППУ по поддержке процесса авторизации Пользователя.
- Конечная точка позволяет ППУ отправлять СППУ идентификатор созданного ресурса согласия на проведение платежа (consentId).
- Сразу после создания ресурса согласия на проведение платежа ППУ присваивает ему статус по умолчанию "AwaitingAuthorisation".

6.6.1.2 GET /payment-consents/{consentId}

```
GET /payment-consents/{consentId}
```

Таблица 40 - Получение статуса ресурса согласия на проведение платежа

Данная конечная точка позволяет СППУ получать детали и статус созданного им ресурса согласия на проведение платежа по его идентификатору.

6.6.1.2.1 Статус ресурса согласия на проведение платежа

После того, как Пользователь авторизовал согласие на проведение платежа, ППУ изменяет статус ресурса на "Authorised".

При отклонении согласия Пользователем, ППУ изменяет статус ресурса на "Rejected".

При возникновении иных причин препятствующих проведению платежа, ППУ изменяет статус ресурса на "Rejected".

После того, как ресурс согласия был задействован при выполнении платежа, ППУ изменяет статус ресурса на "Consumed".

Статус ресурса согласия на проведение платежа имеет следующие значения:

№	Статус	Описание
1	Rejected	Согласие на проведение платежа отклонено.
2	AwaitingAuthorisation	Согласие на проведение платежа ожидает авторизации.
3	Authorised	Согласие на проведение платежа успешно авторизовано.
4	Consumed	Согласие на проведение платежа было задействовано и больше не может использоваться при проведении платежа.

Таблица 41 - Статусы ресурса согласия на проведение платежа

6.6.1.3 POST /payments

```
GET /payments
```

Таблица 42 - Создание ресурса платежа

Конечная точка используется после того, как Пользователь авторизовал ресурс согласия на проведение платежа.

Конечная точка используется для подтверждения инициирования проведения платежа.

Данная конечная точка позволяет выполнять следующее взаимодействие:

- Конечная точка позволяет с помощью метода POST создавать на стороне ППУ ресурс платежа (payment).
- Запрос, выполненный с помощью конечной точки, является инструкцией для ППУ на выполнение платежа.
- СППУ обеспечивает соответствие разделов "Initiation" и "Risk" ресурса платежа соответствующим разделам "Initiation" и "Risk" ресурса согласия на проведение платежа (значения всех присутствующих и там и там элементов должны быть равны). Если есть хотя бы одно несовпадение, ППУ отчает ошибкой с кодом 400 (Bad Request).
- Любые операции с ресурсом платежа не приводят к изменению статуса ресурса платежа.

6.6.1.4 GET /payments/{paymentId}

```
GET /payments/{paymentId}
```

Таблица 43 - Получение статуса ресурса платежа

Данная конечная точка позволяет СППУ получать статус созданного ресурса платежа по его идентификатору.

6.6.1.4.1 Статус

Статус ресурса платежа может иметь следующие значения:

№	Статус	Наименование статуса	Описание
1	PDNG	Pending	Начальный статус выполнения платежа.
2	RJCT	Rejected	Платеж отклонен.
3	ACSP	AcceptedSettlementInProgress	Все проверки, такие как техническая проверка и профиль клиента, были успешными, и поэтому инициирование платежа было принято к исполнению.
4	ACSC	AcceptedSettlementCompleted	Расчет по счету Плательщика выполнен.
5	ACWP	AcceptedWithoutPosting	Платежная инструкция, включенная в перевод, принимается без зачисления на счет Получателя средств.
6	ACCC	AcceptedCreditSettlementCompleted	Расчет по счету Получателя средств завершен.

Таблица 44 - Статусы ресурса платежа

6.6.1.4.2 Модель статусов обработки

6.6.1.4.3 Статус операций для ресурса согласия на проведение платежа
Модель состояний для ресурса согласия на проведение платежа соответствует следующей модели:

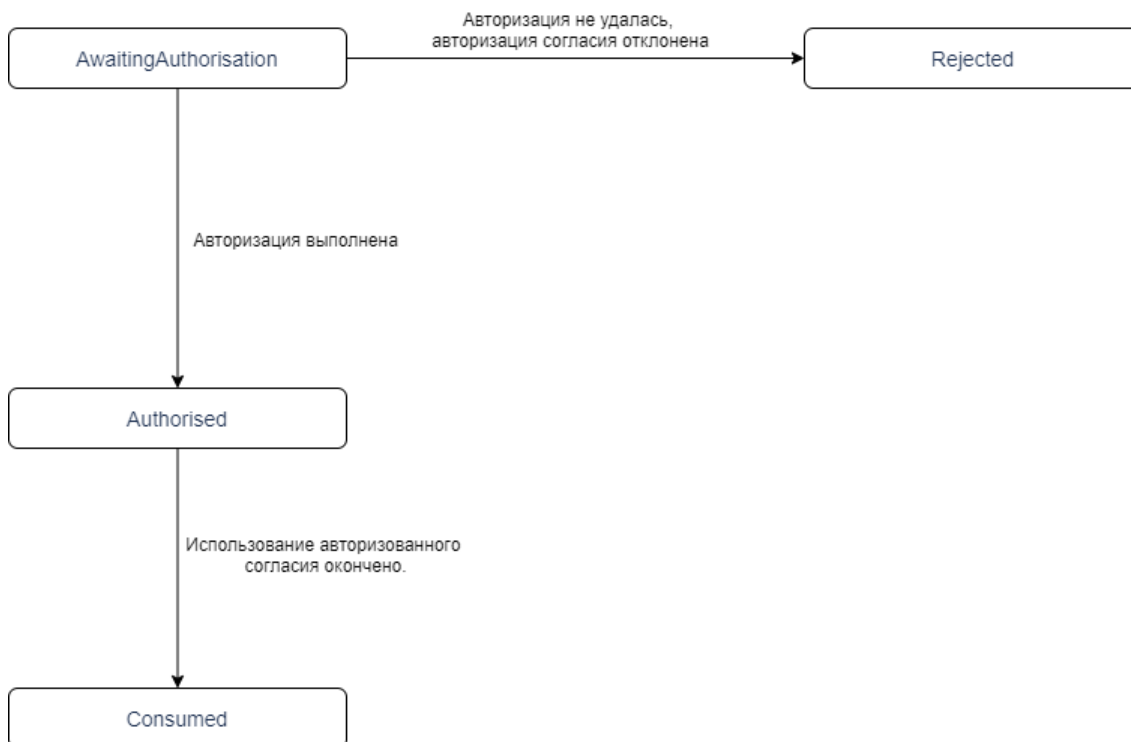


Рисунок 9 - Диаграмма состояний ресурса согласия на проведение платежа

Описание статусов:

№	Статус	Описание
1	Rejected	Согласие на проведение платежа отклонено.
2	AwaitingAuthorisation	Согласие на проведение платежа ожидает авторизации.
3	Authorised	Согласие на проведение платежа успешно авторизовано.
4	Consumed	Согласие на проведение платежа было задействовано и больше не может использоваться при проведении платежа.

Таблица 45 - Статусы ресурса согласия на проведение платежа

6.6.1.4.4 Статус операций для ресурса платежа

Модель состояния для ресурса платежа соответствует статусам из стандарта ISO 20022.

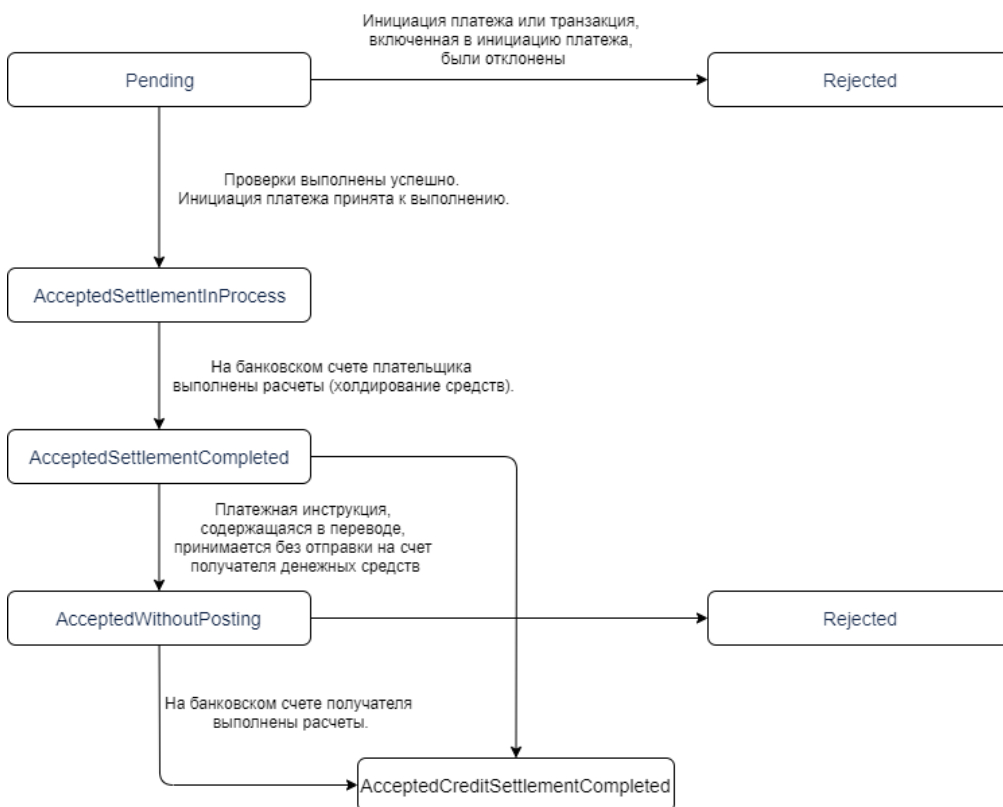


Рисунок 10 - Диаграмма состояний статусов ресурса платежа

Описание статусов:

№	Статус	Наименование статуса	Описание
1	PDNG	Pending	Начальный статус выполнения платежа.
2	RJCT	Rejected	Платеж отклонен.
3	ACSP	AcceptedSettlementInProgress	Все проверки, такие как техническая проверка и профиль клиента, были успешными, и поэтому инициирование платежа было принято к исполнению.
4	ACSC	AcceptedSettlementCompleted	Расчет по счету Плательщика выполнен.
5	ACWP	AcceptedWithoutPosting	Платежная инструкция, включенная в перевод, принимается без зачисления на счет Получателя средств.
6	ACCC	AcceptedCreditSettlementCompleted	Расчет по счету Получателя средств завершен.

Таблица 46 - Статусы ресурса платежа

6.6.1.4.5 Множественная авторизация

Если распоряжение о переводе средств требует множественной авторизации, то статус авторизации от нескольких Пользователей обновляется в объекте "MultiAuthorisation".

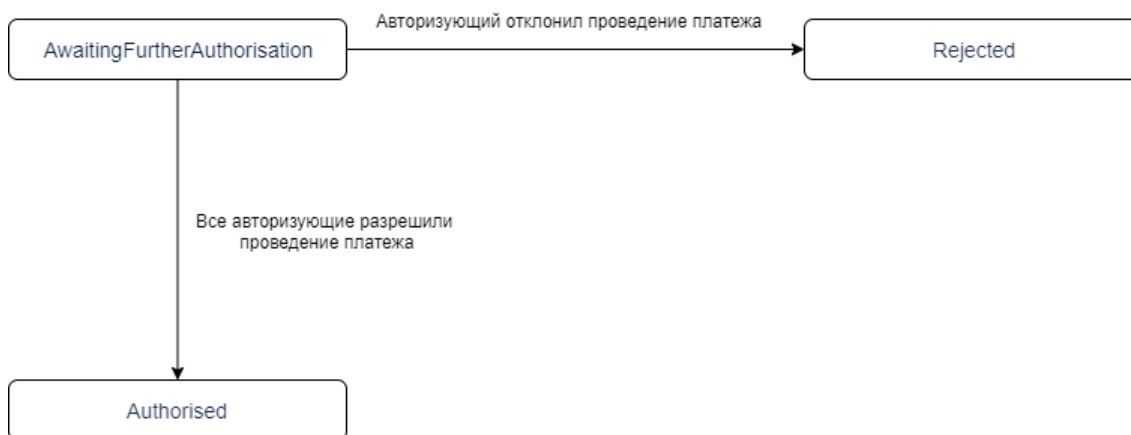


Рисунок 11 - Диаграмма статусов ресурса платежа для множественной авторизации

Описание статусов:

№	Статус	Описание состояния
1	AwaitingFurtherAuthorisation	Ресурс платежа ожидает дальнейшей авторизации. Не все необходимые участники авторизовались.
2	Rejected	Ресурс платежа был отклонен, как минимум одним Пользователем, выполняющим авторизацию.
3	Authorised	Ресурс платежа был успешно авторизован всеми необходимыми для этого Пользователями.

Таблица 47 - Статусы ресурса платежа для множественной авторизации

6.6.2 Модель данных

В данном разделе приведены подробные описания полезных нагрузок запросов и ответов для потоков API платежей, осуществляемых на территории Российской Федерации.

6.6.2.1 Повторно используемые объекты

6.6.2.1.1 Initiation

В данном разделе описывается объект Initiation, который повторно используется в ресурсах согласия на проведение платежа (payment-consent) и платежа (payment).

6.6.2.1.1.1 UML Диаграмма

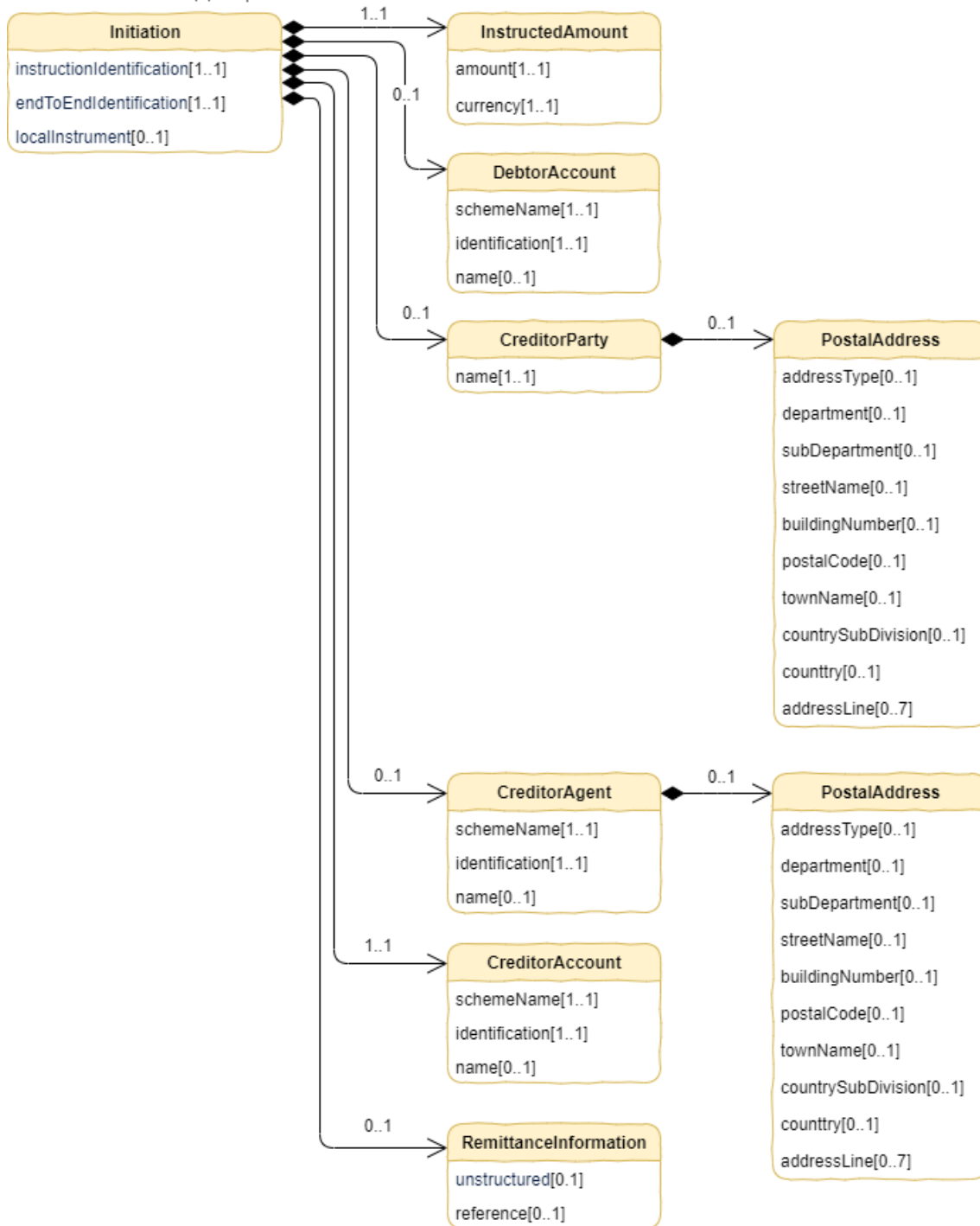


Рисунок 12 - Диаграмма объекта Initiation

Требования к объекту "Initiation", которые выполняют участники:

- Все элементы в объекте "Initiation", которые заполнялись на стороне СППУ, не изменяются на стороне ППУ, поскольку это является официальным согласием Пользователя.
- Если ППУ определяет проблему с полезной нагрузкой во время вызова API, то ППУ немедленно отклоняет запрос получения согласия на проведение платежа.

- Если ППУ определяет проблему с согласием на платеж после вызова API, то ППУ изменяет статус ресурса согласия на проведение платежа на "Rejected".
- Раздел "DebtorAccount" является необязательным, поскольку СППУ может не знать идентификационные данные банковского счета Пользователя.
- Если значения элементов объекта "DebtorAccount" заданы на стороне СППУ и являются неверными для Пользователя, то после аутентификации Пользователя статус согласия на проведение платежа меняется на "Rejected".
- Элемент "CreditorReferenceInformation" был переименован на "Reference", поскольку это уникальный элемент ISO 20022, используемый в сообщении pain.001, а не компонент сообщения ISO 20022.
- Поскольку продавец (merchant, а не Пользователь) может инициировать платеж через СППУ, в полезную нагрузку включены два идентификатора:
 - Идентификатор "instructionIdentification" генерируется СППУ.
 - СППУ гарантирует уникальность идентификатора "instructionIdentification" на неопределенный промежуток времени.
 - Идентификатор "endToEndIdentification" генерируется продавцом.
- Идентификатор "instructionIdentification" не используется в качестве идентификатора "consentId", поскольку последний создается на стороне ППУ.
- Идентификатор "endToEndIdentification" не используется в качестве идентификатора "consentId", поскольку последний создается на стороне ППУ.

6.6.2.1.2 Состав данных объекта Initiation

Наименование	Кратность	Путь	Описание	Тип	Значения	Шаблон
Initiation		Initiation	Полезная нагрузка отправляется иницирующей стороной в ППУ. Используется для запроса о переводе средств со счета Плательщика на счет Получателя средств для одиночного внутреннего платежа	InitiationComplexType		
instructionIdentification	1..1	Initiation/instructionIdentification	Уникальная идентификация, присвоенная инструктирующей стороной для проинструктированной стороны, для однозначной	Max35Text		

Наименование	Кратность	Путь	Описание	Тип	Значения	Шаблон
			<p>идентификации инструкции.</p> <p>Использование: Идентификация инструкции является сквозным идентификатором, который может использоваться между инструктирующей стороной и проинструктированной стороной для ссылки на индивидуальную инструкцию. Она может включаться в несколько сообщений, относящихся к инструкции.</p>			
endToEndIdentification	1..1	Initiation/endToEndIdentification	Уникальная идентификация, присвоенная иницилирующей стороной для	Max35Text		

Наименование	Кратность	Путь	Описание	Тип	Значения	Шаблон
			<p>однозначной идентификации операции. Эта идентификация передается без изменений по всей цепочке от начала до конца.</p> <p>Использование: Сквозной идентификатор может использоваться для выверки расчетов для связки задач, относящихся к операции. Она может включаться в несколько сообщений, относящихся к операции.</p>			
localInstrument	0..1	Initiation/localInstrument	Специальный инструмент, используемый сообществом.	ExternalLocalInstrument1Code		

Наименование	Кратность	Путь	Описание	Тип	Значения	Шаблон
			Использование: Этот элемент используется для указания местного инструмента, варианта локального клиринга и/или для дополнительного описания сервиса или уровня обслуживания.			
InstructedAmount	1..1	Initiation/InstructedAmount	Сумма денег, подлежащая переводу между Плательщиком и Получателем средств, до вычета комиссии, выраженная в валюте поручения инициатора.	AmountComplexType		
amount	1..1	Initiation/InstructedAmount/amount	Сумма денежных средств, подлежащая	ActiveCurrencyAndAmount_SimpleType		^d{1,13}\\.d{1,5}\$

Наименование	Кратность	Путь	Описание	Тип	Значения	Шаблон
			перемещению между Плательщиком и Получателем средств, до вычета сборов, с указанием валюты, которую определила иницирующая сторона.			
currency	1..1	Initiation/InstructedAmount/currency	Валюта, согласно стандарта ISO 4217.	ActiveOrHistoricCurrencyCode		^[A-Z]{3,3}\$
DebtorAccount	0..1	Initiation/DebtorAccount	Однозначная идентификация счета Плательщика, по которому будет отражена дебетовая запись в результате операции.	CashAccountComplexType		
schemeName	1..1	Initiation/DebtorAccount/schemeName	Наименование схемы	AccountIdentificationDynamicType		

Наименование	Кратность	Путь	Описание	Тип	Значения	Шаблон
			идентификации			
identification	1..1	Initiation/DebtorAccount/identification	Уникальная и однозначная идентификация счета, используемая между владельцем счета и организацией, обслуживающей счет.	Max256Text		
name	0..1	Initiation/DebtorAccount/name	Имя владельца или владельцев учетной записи (банковского счета, карты, телефона).	Max70Text		
CreditorParty	0..1	Initiation/CreditorParty	Информация о контрагенте	PartyIdentificationComplexType		
name	1..1	Initiation/CreditorParty/name	Наименование контрагента	Max160Text		
PostalAddress	0..1	Initiation/CreditorParty/PostalAddress	Информация, которая находит и идентифицирует определенный	PostalAddressComplexType		

Наименование	Кратность	Путь	Описание	Тип	Значения	Шаблон
			адрес, как определено почтовыми службами.			
addressType	0..1	Initiation/CreditorParty/PostalAddress/addressType	Тип адреса.	AddressTypeStaticType	Business Correspondence DeliveryTo MailTo POBox Postal Residential Statement	
department	0..1	Initiation/CreditorParty/PostalAddress/department	Департамент или строение	Max70Text		
subDepartment	0..1	Initiation/CreditorParty/PostalAddress/subDepartment	Идентификация подразделения организации или идентификация здания	Max70Text		
streetName	0..1	Initiation/CreditorParty/PostalAddress/streetName	Наименование улицы или проспекта.	Max70Text		

Наименование	Кратность	Путь	Описание	Тип	Значения	Шаблон
buildingNumber	0..1	Initiation/CreditorParty/PostalAddress/buildingNumber	Номер, который определяет положение здания на улице.	Max16Text		
postCode	0..1	Initiation/CreditorParty/PostalAddress/postCode	Идентификатор, состоящий из группы букв и/или цифр, которые добавляются к почтовому адресу для упрощения сортировки почты.	Max16Text		
townName	0..1	Initiation/CreditorParty/PostalAddress/townName	Наименование застроенного пространства с определенными границами и органами местного самоуправления.	Max35Text		
countrySubDivision	0..1	Initiation/CreditorParty/PostalAddress/countrySubDivision	Определяет часть страны, например, край, область, республика.	Max35Text		

Наименование	Кратность	Путь	Описание	Тип	Значения	Шаблон
country	0..1	Initiation/CreditorParty/PostalAddress/country	Государство.	CountryCode		^[A-Z]{2,2}\$
addressLine	0..7	Initiation/CreditorParty/PostalAddress/addressLine	Информация, которая определяет и идентифицирует определенный адрес, как это определено почтовыми службами, в форме свободного текста.	Max70Text		
CreditorAgent	0..1	Initiation/CreditorAgent	Финансовое организация, обслуживающая счет Получателя средств	BranchAndFinancialInstitutionIdentificationComplexType		
schemeName	1..1	Initiation/CreditorAgent/schemeName	БИК/SWIFT банка контрагента	FinancialInstitutionIdentificationDynamicType		
identification	1..1	Initiation/CreditorAgent/identification	БИК/SWIFT банка контрагента	Max35Text		

Наименование	Кратность	Путь	Описание	Тип	Значения	Шаблон
name	0..1	Initiation/CreditorAgent//name	Наименование банка контрагента	Max160Text		
PostalAddress	0..1	Initiation/CreditorAgent/PostalAddresses	Информация, которая находит и идентифицирует определенный адрес, как определено почтовыми службами.	PostalAddressComplexType		
addressType	0..1	Initiation/CreditorAgent/PostalAddress/addressType	Тип адреса.	AddressTypeStaticType	Business Correspondence DeliveryTo MailTo POBox Postal Residential Statement	
department	0..1	Initiation/CreditorAgent/PostalAddress/department	Департамент или строение	Max70Text		
subDepartment	0..1	Initiation/CreditorAgent/PostalAddress/subDepartment	Идентификация подразделения организации	Max70Text		

Наименование	Кратность	Путь	Описание	Тип	Значения	Шаблон
			или идентификация здания			
streetName	0..1	Initiation/CreditorAgent/PostalAddress/streetName	Наименование улицы или проспекта.	Max70Text		
buildingNumber	0..1	Initiation/CreditorAgent/PostalAddress/buildingNumber	Номер, который определяет положение здания на улице.	Max16Text		
postCode	0..1	Initiation/CreditorAgent/PostalAddress/postCode	Идентификатор, состоящий из группы букв и/или цифр, которые добавляются к почтовому адресу для упрощения сортировки почты.	Max16Text		
townName	0..1	Initiation/CreditorAgent/PostalAddress/townName	Наименование застроенного пространства с определенными границами и органами местного	Max35Text		

Наименование	Кратность	Путь	Описание	Тип	Значения	Шаблон
			самоуправления.			
countrySubDivision	0..1	Initiation/CreditorAgent/PostalAddress/countrySubDivision	Определяет часть страны, например, край, область, республика.	Max35Text		
country	0..1	Initiation/CreditorAgent/PostalAddress/country	Государство.	CountryCode		^[A-Z]{2,2}\$
addressLine	0..7	Initiation/CreditorAgent/PostalAddress/addressLine	Информация, которая определяет и идентифицирует определенный адрес, как это определено почтовыми службами, в форме свободного текста.	Max70Text		
CreditorAccount	1..1	Initiation/CreditorAccount	Идентификатор счета получателя, на котором будет проведена запись о кредите в результате	CashAccountComplexType		

Наименование	Кратность	Путь	Описание	Тип	Значения	Шаблон
			платежной транзакции.			
schemeName	1..1	Initiation/CreditorAccount/schemeName	Наименование схемы идентификации	AccountIdentificationDynamicType		
identification	1..1	Initiation/CreditorAccount/identification	Идентификатор счета Получателя средств, на котором будет проведена кредитная запись в результате платежной транзакции.	Max256Text		
name	0..1	Initiation/CreditorAccount/name	Имя владельца или владельцев учетной записи (банковского счета, карты, телефона). ППУ могут выполнять проверку имени для подтверждения получателя, но	Max70Text		

Наименование	Кратность	Путь	Описание	Тип	Значения	Шаблон
			это не обязательно.			
RemittanceInformation	0..1	Initiation/RemittanceInformation	Предоставляемая информация, позволяющая сопоставить запись с позициями, для которых предназначен перевод, такими как коммерческие счета в системе дебиторской задолженности.	RemittanceInformationComplexType		
unstructured	0..1	Initiation/RemittanceInformation/unstructured	Информация, позволяющая сопоставлять / сверять записи с позициями, для которых предназначен платеж, такими как коммерческие счета в системе дебиторской задолженности,	Max140Text		

Наименование	Кратность	Путь	Описание	Тип	Значения	Шаблон
			В неструктурированной форме.			
reference	0..1	Initiation/RemittanceInformation/reference	Уникальная ссылка, присвоенная получателем, для однозначной ссылки на платежную транзакцию.	Max35Text		

Таблица 48 - Состав данных объекта Initiation

6.6.2.2 ConsentRequest

Объект ConsentRequest используется в запросе при вызове:

- POST /payment-consents

6.6.2.2.1 UML Диаграмма

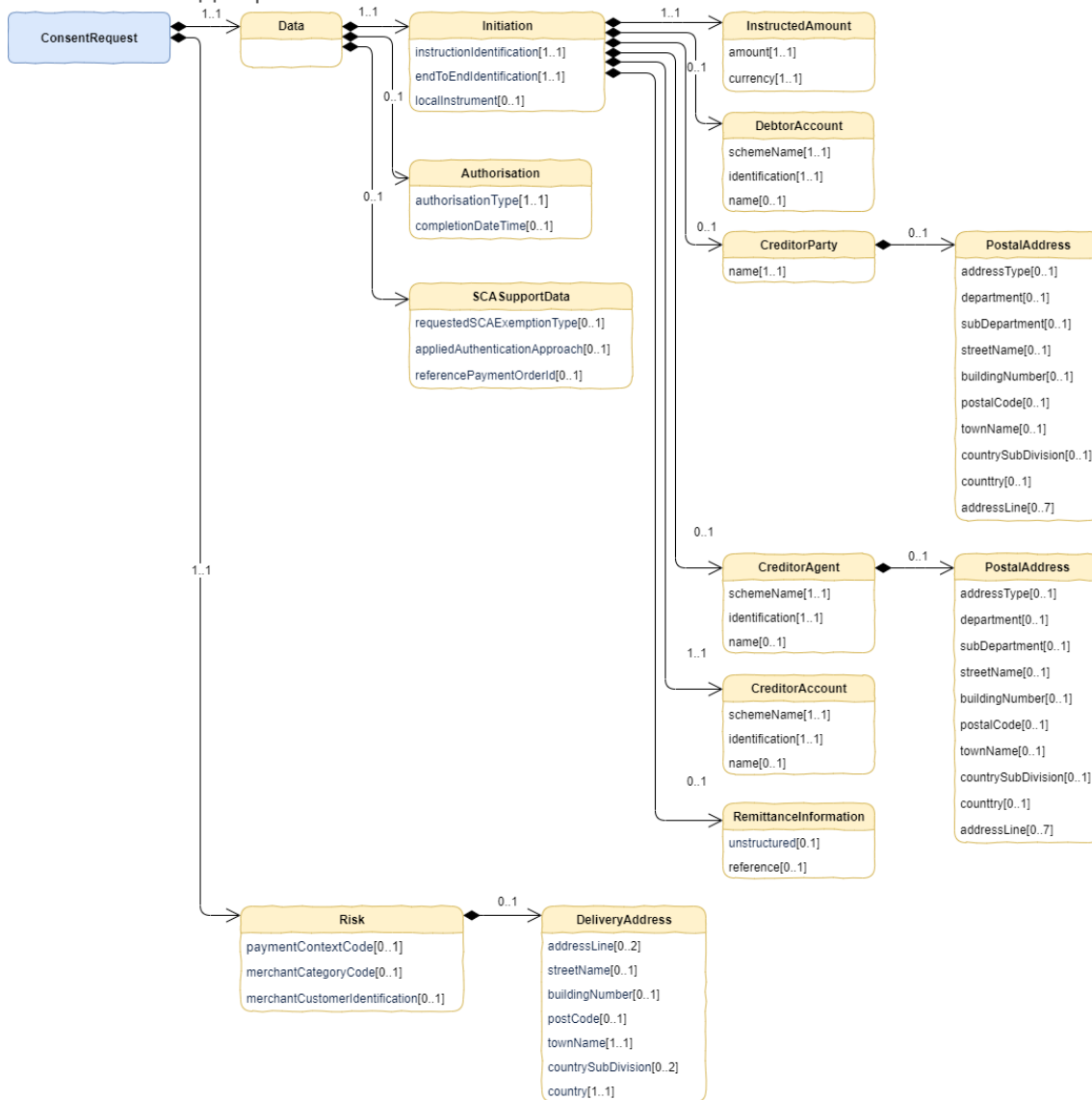


Рисунок 13 - Диаграмма объекта ConsentRequest

Объект ConsentRequest состоит из следующих разделов (объектов):

- Initiation
- Authorisation
- SCASupportData
- Risk

6.6.2.2.2 Состав данных объекта ConsentRequest

Наименование	Кратность	Путь	Описание	Тип	Значения	Шаблон
ConsentRequest				ConsentRequestComplexType		
Data	1..1	ConsentRequest/Data		DataConsentRequestComplexType		
Initiation	1..1	ConsentRequest/Data/Initiation	Полезная нагрузка отправляется иницирующей стороной в ППУ. Используется для запроса перевода средств со счета Плательщика на счет Получателя средств для одиночного внутреннего платежа.	InitiationComplexType		
Authorisation	0..1	ConsentRequest/Data/Authorisation	Запрос типа авторизации от СППУ.	AuthorisationComplexType		
SCASupportData	0..1	ConsentRequest/Data/SCASupportData	Вспомогательные данные, предоставленные СППУ, при запросе SCA.	SCASupportDataComplexType		
Risk	1..1	ConsentRequest/Risk	Раздел "Risk" отправляется инициатором в ППУ, используется для указания дополнительных деталей для оценки рисков при проведении платежей.	RiskComplexType		

Таблица 49 - Состав данных объекта ConsentRequest

6.6.2.3 ConsentResponse

Класс объекта ConsentResponse используется при ответе на вызов:

- POST /payment-consents
- GET /payment-consents/{consentId}

6.6.2.3.1 UML Диаграмма

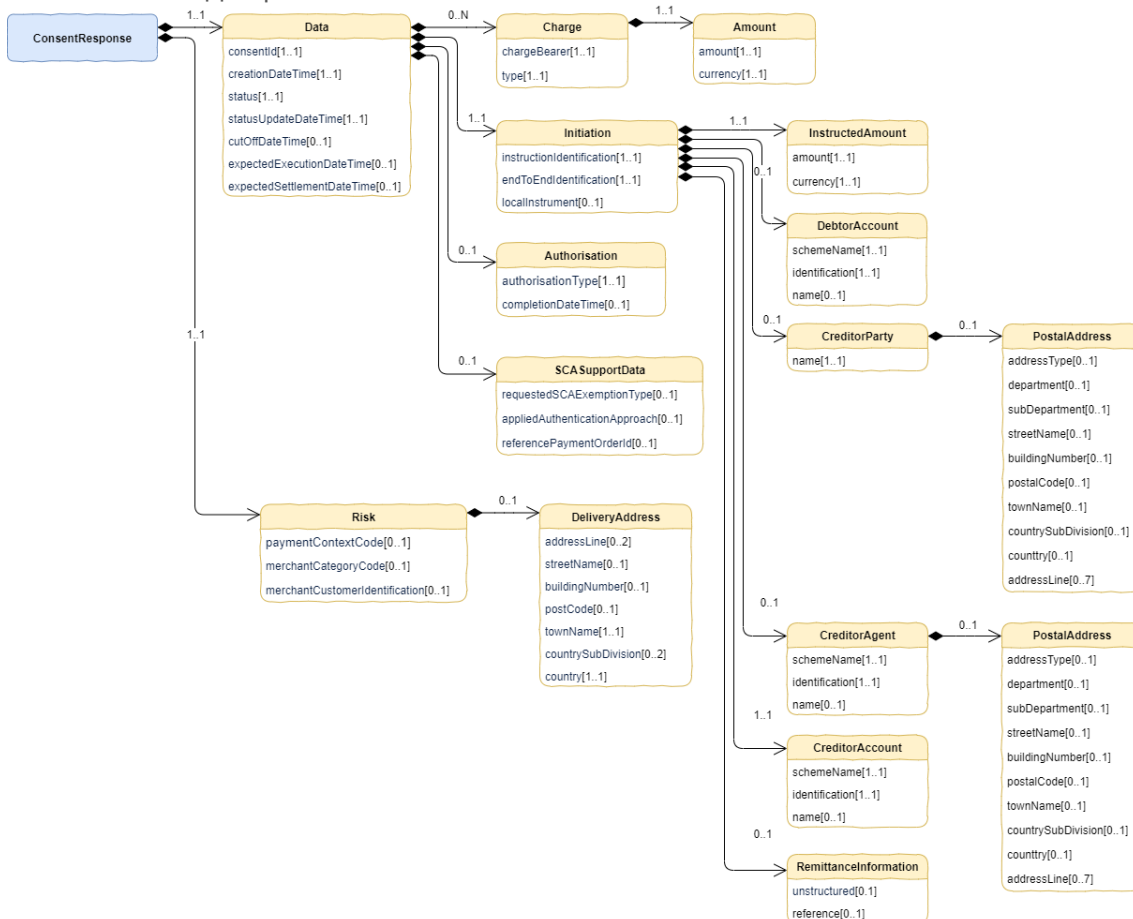


Рисунок 14 - Диаграмма объекта ConsentResponse

Ответ согласия на проведение платежа содержит полную полезную нагрузку запроса согласия на проведение платежа с дополнительными элементами:

- Идентификатор ресурса согласия на проведение платежа consentId, созданного на стороне ППУ.
- Элемент creationDateTime - дата и время создания ресурса согласия на проведение платежа.
- Элементы status и statusUpdateDateTime - статус, дата и время изменения статуса ресурса согласия на проведение платежа.
- Элемент CutOffDateTime и его использование описаны в разделе "[Использование элемента CutOffDateTime](#)"
- Элемент "expectedExecutionDateTime" означает прогнозируемую дату и время выполнения платежа и предназначен для ресурса платежа. Если ресурс платежа создан до даты "cutOffDateTime", то "expectedExecutionDateTime" означает ожидаемую дату и время списания средств со счета Плательщика. ППУ

обновляет значение этого поля при любых изменениях (например, после авторизации Пользователя).

- Элемент "expectedSettlementDateTime" означает прогнозируемую дату и время выполнения платежа и предназначен для ресурса платежа. Если ресурс платежа создан до даты "cutOffDateTime", то "expectedSettlementDateTime" означает ожидаемую дату и время зачисления средств на счет Получателя средств. ППУ обновляет значение этого поля при любых изменениях (например, после авторизации Пользователя).
- Элемент "Charges" - это массив для разбивки применимых сборов ППУ (разбиение комиссий).

6.6.2.3.2 Состав данных объекта ConsentResponse

Наименование	Кратность	Путь	Описание	Тип	Значения	Шаблон
ConsentResponse				ConsentResponseComplexType		
Data	1..1	ConsentResponse/Data		DataConsentRequestComplexType		
consentId	1..1	ConsentResponse/Data/consentId	Идентификатор ресурса согласия на проведение платежа, присвоенный на стороне ППУ.	Max128Text		
creationDateTime	1..1	ConsentResponse/Data/creationDateTime	Дата и время создания ресурса согласия на проведение платежа.	ISODateTime		
status	1..1	ConsentResponse/Data/status	Текущий статус ресурса согласия на проведение платежа.	ConsentStatusStaticType	Authorised AwaitingAuthorisation Consumed Rejected	
statusUpdateDateTime	1..1	ConsentResponse/Data/statusUpdateDateTime	Дата и время обновления статуса ресурса	ISODateTime		

Наименование	Кратность	Путь	Описание	Тип	Значения	Шаблон
			согласия на проведение платежа.			
cutOffDateTime	0..1	ConsentResponse/Data/cutOffDateTime	Дата и время актуальности согласия на проведение платежа (после этого согласие является не действующим)	ISODatetime		
expectedExecutionDateTime	0..1	ConsentResponse/Data/expectedExecutionDateTime	Ожидаемая дата и время исполнения платежа (платежного ресурса).	ISODatetime		
expectedSettlementDateTime	0..1	ConsentResponse/Data/expectedSettlementDateTime	Ожидаемая дата и время расчета платежа (платежного ресурса).	ISODatetime		
Charges	0..N	ConsentResponse/Data/Charges	Сборы при проведении платежа.	ChargeComplexType		

Наименование	Кратность	Путь	Описание	Тип	Значения	Шаблон
Initiation	1..1	ConsentResponse/Data/Initiation	Полезная нагрузка отправляется иницирующей стороной в ГПУ. Используется для запроса перевода средств со счета плательщика на счет получателя для одиночного внутреннего платежа.	InitiationComplexType		
Authorisation	0..1	ConsentResponse/Data/Authorisation	Запрос типа авторизации от СППУ.	AuthorisationComplexType		
SCASupportData	0..1	ConsentResponse/Data/SCASupportData	Вспомогательные данные, предоставленные СППУ, при запросе SCA.	SCASupportDataComplexType		
Risk	1..1	ConsentResponse/Risk	Раздел "Risk" отправляется инициатором в ГПУ,	RiskComplexType		

Наименование	Кратнос ть	Путь	Описание	Тип	Значения	Шабл он
			используется для указания дополнительных деталей для оценки рисков при проведении платежей.			

Таблица 50 - Состав данных объекта ConsentResponse

6.6.2.4 PaymentRequest

Объект PaymentRequest используется в запросе при вызове:

- POST /payments
- GET /payments/{paymentId}

6.6.2.4.1 UML Диаграмма

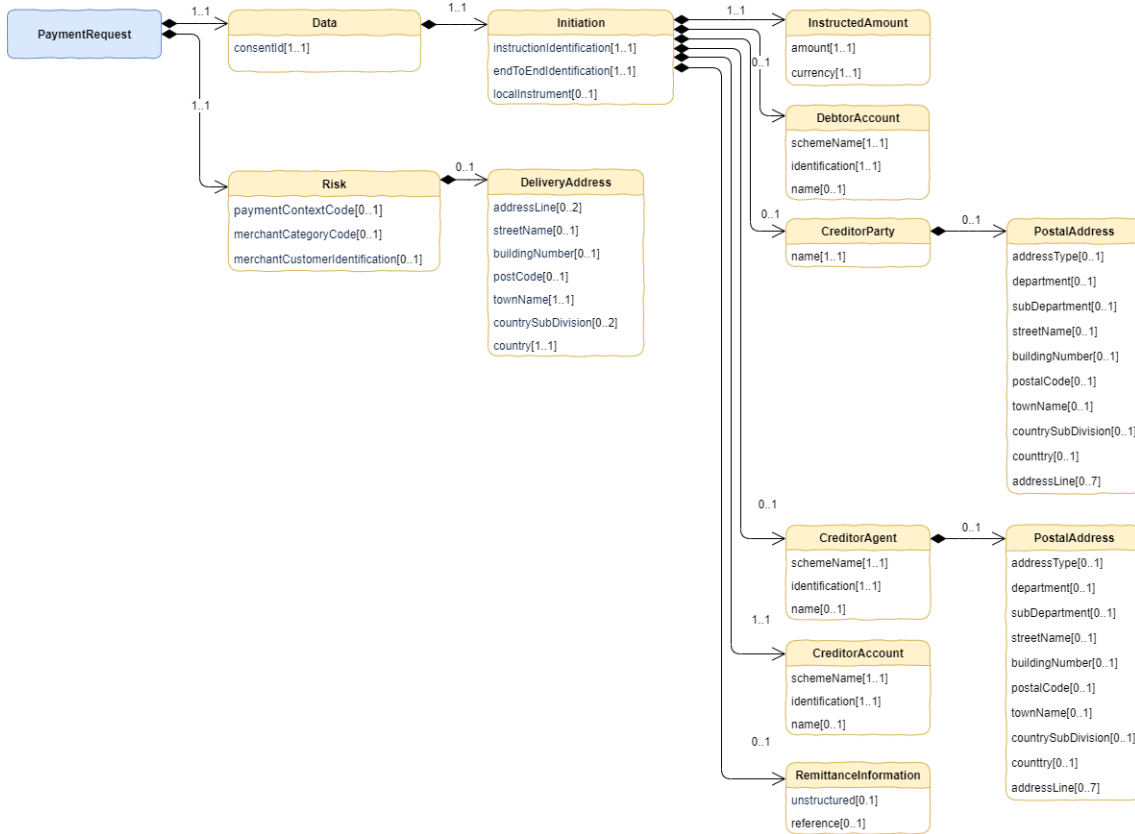


Рисунок 15 - Диаграмма объекта PaymentRequest

Объект "PaymentRequest" содержит следующие элементы:

- Идентификатор соответствующего ресурса согласия на проведение платежа consentId.
- Объекты "Initiation" и "Risk" и все их элементы, которые содержались в запросе на согласие о проведении платежа.

Все элементы объектов "Initiation" и "Risk" в запросе на проведение платежа должны совпадать с соответствующими элементами объектов "Initiation" и "Risk", которые были в запросе согласия на проведение платежа. Проверка соответствия осуществляется, как на стороне СППУ, так и на стороне ППУ. В случае несовпадения ресурс платежа не создается, ресурс согласия переводится в статус "Rejected" и ППУ отвечает СППУ сообщением об ошибке несовпадения данных.

6.6.2.4.2 Состав данных объекта PaymentRequest

Наименование	Кратность	Путь	Описание	Тип	Значения	Шаблон
PaymentRequest				PaymentRequestComplexType		
Data	1..1	PaymentRequest/Data		DataPaymentRequestComplexType		
consentId	1..1	PaymentRequest/Data/consentId	Идентификатор ресурса согласия на проведение платежа, присвоенный на стороне ППУ.	Max128Text		
Initiation	1..1	PaymentRequest/Data/Initiation	Полезная нагрузка отправляется иницирующей стороной в ППУ. Используется для запроса перевода средств со счета плательщика на счет получателя для одиночного внутреннего платежа.	InitiationComplexType		
Risk	1..1	PaymentRequest/Risk	Раздел "Risk" отправляется инициатором в ППУ, используется для указания дополнительных деталей для оценки рисков при проведении платежей.	RiskComplexType		

Таблица 51 - Состав данных объекта PaymentRequest

6.6.2.5 PaymentResponse

Объект PaymentResponse используется при ответе на вызов:

- POST /domestic-payments

6.6.2.5.1 UML Диаграмма

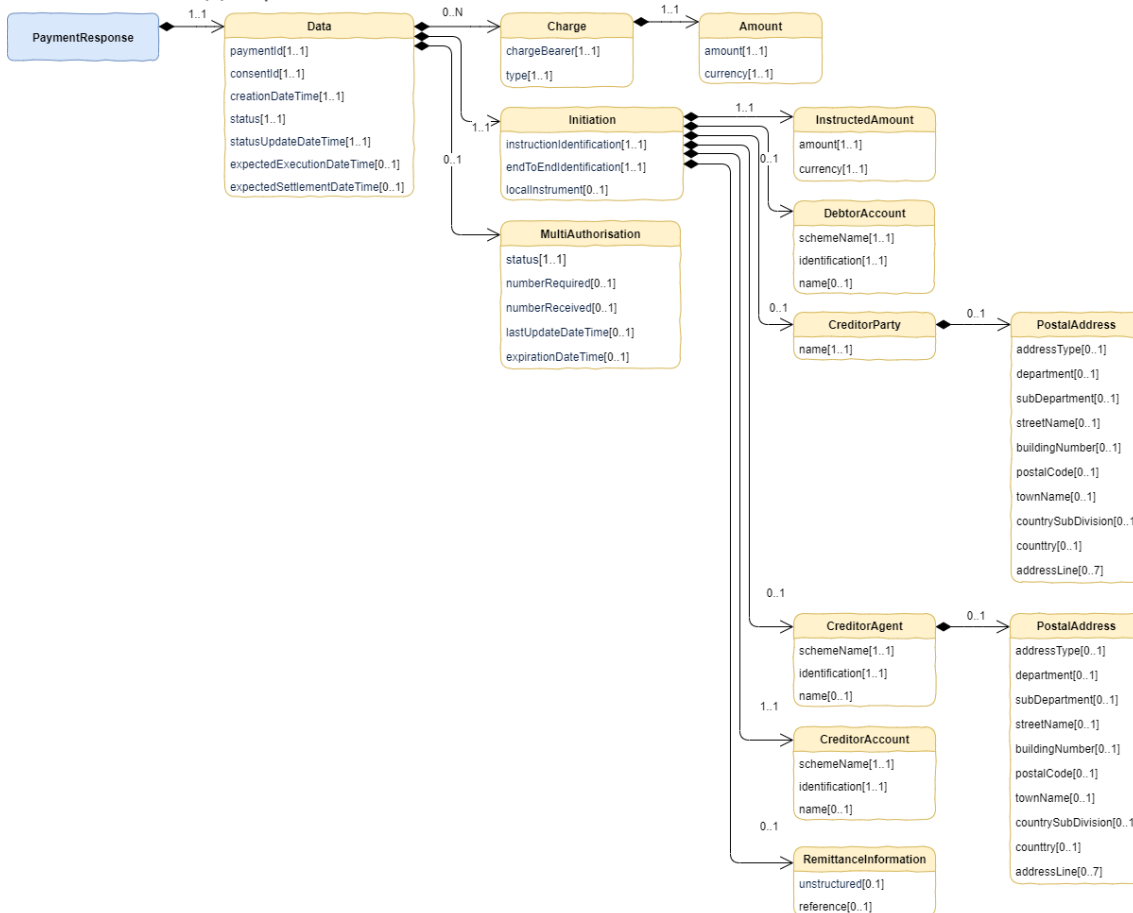


Рисунок 16 - Диаграмма объекта PaymentResponse

Объект PaymentResponse содержит следующие элементы:

- Элемент "paymentId" - идентификатор ресурса платежа, созданный на стороне ППУ.
- Элемент "consentId" - идентификатор ресурса согласия на проведение платежа, созданный на стороне ППУ.
- Элемент creationDateTime - дата и время создания ресурса платежа.
- Элементы status и statusUpdateDateTime - статус, дата и время изменения статуса ресурса платежа.
- Элемент "expectedExecutionDateTime" означает ожидаемую дату и время списания средств со счета плательщика.
- Элемент "expectedSettlementDateTime" означает ожидаемую дату и время зачисления средств на счет получателя.
- Элемент "Charges" - это массив для разбивки применимых сборов ППУ (разбиение комиссий).
- Объект "Initiation", соответствующий одноименному объекту из запроса согласия на проведение платежа.

- Объект "MultiAuthorisation", если для ресурса платежей требуется множественная авторизация.

6.6.2.5.2 Состав данных объекта PaymentResponse

Наименование	Кратность	Путь	Описание	Тип	Значения	Шаблон
PaymentResponse				PaymentResponseComplexType		
Data	1..1	PaymentResponse/Data		DataPaymentRequestComplexType		
consentId	1..1	PaymentResponse/Data/consentId	Идентификатор ресурса согласия на проведение платежа, присвоенный на стороне ППУ.	Max128Text		
creationDateTime	1..1	PaymentResponse/Data/creationDateTime	Дата и время создания ресурса согласия на проведение платежа.	ISODateTime		
status	1..1	PaymentResponse/Data/status	Текущий статус ресурса согласия на проведение платежа.	TransactionStatusStaticType	AcceptedCreditSettlementCompleted AcceptedWithoutPosting AcceptedSettlementCompleted AcceptedSettlementInProcess	

Наименование	Кратность	Путь	Описание	Тип	Значения	Шаблон
					Pending Rejected	
statusUpdateDateTime	1..1	PaymentResponse/Data/statusUpdateDateTime	Дата и время обновления статуса ресурса согласия на проведение платежа.	ISODateTime		
expectedExecutionDateTime	0..1	PaymentResponse/Data/expectedExecutionDateTime	Ожидаемая дата и время исполнения платежа (платежного ресурса).	ISODateTime		
expectedSettlementDateTime	0..1	PaymentResponse/Data/expectedSettlementDateTime	Ожидаемая дата и время расчета платежа (платежного ресурса).	ISODateTime		
Charges	0..N	PaymentResponse/Data/Charges	Сборы при проведении платежа.	ChargeComplexType		
Initiation	1..1	PaymentResponse/Data/Initiation	Полезная нагрузка	InitiationComplexType		

Наименование	Кратность	Путь	Описание	Тип	Значения	Шаблон
			отправляет инициирующей стороной в ППУ. Используется для запроса перевода средств со счета плательщика на счет получателя для одиночного внутреннего платежа.			
MultiAuthorisation	0..1	PaymentResponse/Data/MultiAuthorisation	Ответ потока множественной авторизации от ППУ.	MultiAuthorisationComplex Type		

Таблица 52 - Состав данных объекта PaymentResponse

6.6.3 Примеры использования

6.6.3.1 Инициирование продавцом через СППУ

Для данного случая оплаты:

- Продавец не указывает реквизиты счета Плательщика.
- Пользователь (Плательщик) выберет реквизиты своего счета во время авторизации согласия.

Пользователь Иван Иванов, зарегистрированный в магазине электронной коммерции "MERCHANT Inc" под идентификатором 053598653254, приобретает товары на сумму 23463.00 рублей с указанием адреса доставки. Продавец присваивает данной платежной инструкции сквозной идентификатор (endToEndIdentification) MERCHANT.256702.IDN.12 и намеривается инициировать через СППУ, в которой зарегистрирован Пользователь, проведение оплаты.

6.6.3.1.1 Создание согласия на проведение платежа

Запрос POST /payment-consents. СППУ присваивает иницируемой платежной инструкции номер (instructionIdentification) PISP412 и создает на стороне ППУ ресурс согласия на проведения платежа вызовом конечной точки POST /payment-consents, указывая идентификаторы ("instructionIdentification": "PISP412", "endToEndIdentification": "MERCHANT.256702.IDN.12"), реквизиты "MERCHANT Inc" и указания дополнительных деталей для оценки рисков платежей в разделе Risk.

Ответ POST /payment-consents. ППУ производит обработку запроса и возвращает идентификатор согласия (consentId) "58923" со статусом "AwaitingAuthorisation", указав дату создания (creationDateTime) (дата обновления статуса (statusUpdateDateTime) в данном случае идентична дате создания).

Далее СППУ перенаправляет Пользователя на страницу ППУ и аутентифицирует его. Пользователь производит авторизацию согласия предоставления своих персональных данных от ППУ к СППУ для создания возможности проведения на стороне СППУ инициирования платежа с реквизитами Пользователя.

```
POST /payment-consents HTTP/1.1
Authorization: Bearer 3YotnDFGEjrlzCsicMWpSS
x-idempotency-key: MERCHANT.256702.IDN.12
x-jws-signature:
UTOmZSdzIGEgam91ckgfyH343YSBkZXN0aW5hdGlvbiA=.D3ghkjk84rk14fIG1yIHR5bGVyI
GdvaW5nIGRvd24gPw==
x-fapi-auth-date: Sun, 29 Sep 2019 12:42:08 GMT
x-fapi-customer-ip-address: 106.32.21.32
x-fapi-interaction-id: 32bae548-f4de-4874-b184-880a4363460d
Content-Type: application/json
Accept: application/json

{
  "Data": {
    "Initiation": {
      "instructionIdentification": "PISP412",
      "endToEndIdentification": "MERCHANT.256702.IDN.12",
      "InstructedAmount": {
        "amount": "23463.00",
        "currency": "RUB"
      },
      "CreditorAccount": {
        "schemeName": "RU.CBR.AccountNumber",
        "identification": "40817810621234567890",
        "name": "MERCHANT Inc"
      },
      "RemittanceInformation": {
        "Reference": "CBR-130",
        "Unstructured": "Назначение платежа - оплата за товары. Внутренний код операции 1234567"
      }
    },
    "Risk": {
      "paymentContextCode": "EcommerceGoods",
      "merchantCategoryCode": "5967",
      "merchantCustomerIdentification": "053598653254",
      "DeliveryAddress": {
        "addressLine": [
          "Шлюзовая наб., 4, Москва, 115114",
          "Rosso Riva"
        ],
        "streetName": "Шлюзовая наб.",
        "buildingNumber": "4",
        "postCode": "115114",
        "townName": "Moscow",
        "countySubDivision": [
          "Moscow"
        ],
        "country": "RU"
      }
    }
  }
}
```

Таблица 53 - Запрос POST /payment-consents

```
HTTP/1.1 201 Created
x-jws-signature:
D4hhdCB3Zfsefe234xQgaGVyZQ0K..dWQgZmFpbHVyZSB0bdsRft5erewf40ZQ0K
x-fapi-interaction-id: 32bae548-f4de-4874-b184-880a4363460d
Content-Type: application/json

{
  "Data": {
    "consentId": "58923",
    "status": "AwaitingAuthorisation",
    "creationDateTime": "2019-06-05T15:15:13+00:00",
    "statusUpdateDateTime": "2019-06-05T15:15:13+00:00",
    "Initiation": {
      "instructionIdentification": "PISP412",
      "endToEndIdentification": "MERCHANT.256702.IDN.12",
      "InstructedAmount": {
        "amount": "23463.00",
        "currency": "RUB"
      },
      "CreditorAccount": {
        "schemeName": "RU.CBR.AccountNumber",
        "identification": "40817810621234567890",
        "name": "MERCHANT Inc"
      },
      "RemittanceInformation": {
        "Reference": "CBR-130",
        "Unstructured": "Назначение платежа - оплата за товары. Внутренний код операции 1234567"
      }
    },
    "Risk": {
      "paymentContextCode": "EcommerceGoods",
      "merchantCategoryCode": "5967",
      "merchantCustomerIdentification": "053598653254",
      "DeliveryAddress": {
        "addressLine": [
          "Шлюзовая наб., 4, Москва, 115114",
          "Rosso Riva"
        ],
        "streetName": "Шлюзовая наб.",
        "buildingNumber": "4",
        "postCode": "115114",
        "townName": "Moscow",
        "countySubDivision": [
          "Moscow"
        ],
        "country": "RU"
      }
    }
  }
}
```

Таблица 54 - Ответ POST /payment-consents

6.6.3.1.2 Создание платежа

Запрос POST /payment. СППУ, получив данные Пользователя, дополняет платежную инструкцию данными Пользователя, указав его идентификатор счета и имя ("Identification": "40817810621234567232", "Name": "Иван Иванов"), а также идентификатор согласия "consentId": "58923" и создает на стороне ППУ ресурс платежа вызовом конечной точки POST /payments.

Ответ на POST /payment-consents. ППУ производит обработку (и процесс исполнения) платежа и возвращает СППУ его идентификатор (paymentId) 58923-001.

```
POST /payments HTTP/1.1
Authorization: Bearer 3YotnDFGEjrlzCsicMWpSS
x-idempotency-key: MERCHANT.256702.IDN.12
x-jws-signature:
UTOmZSdzIGEgam91ckgfyH343YSBkZXN0aW5hdGlvbiA=.D3ghkjk84rk14fIG1yIHR5bGVyI
GdvaW5nIGRvd24gPw==
x-fapi-auth-date: Sun, 29 Sep 2019 12:42:08 GMT
x-fapi-customer-ip-address: 106.32.21.32
x-fapi-interaction-id: 32bae548-f4de-4874-b184-880a4363460d
Content-Type: application/json
Accept: application/json
{
  "Data": {
    "consentId": "58923",
    "Initiation": {
      "instructionIdentification": "PISP412",
      "endToEndIdentification": "MERCHANT.256702.IDN.12",
      "InstructedAmount": {
        "amount": "23463.00",
        "currency": "RUB"
      },
      "DebtorAccount": {
        "SchemeName": "RU.CBR.AccountNumber",
        "Identification": "40817810621234567232",
        "Name": "Иван Иванов"
      },
      "CreditorAccount": {
        "schemeName": "RU.CBR.AccountNumber",
        "identification": "40817810621234567890",
        "name": "MERCHANT Inc"
      },
      "RemittanceInformation": {
        "Reference": "CBR-130",
        "Unstructured": "Назначение платежа - оплата за товары. Внутренний код операции 1234567"
      }
    },
    "Risk": {
      "paymentContextCode": "EcommerceGoods",
      "merchantCategoryCode": "5967",
      "merchantCustomerIdentification": "053598653254",
      "DeliveryAddress": {
        "addressLine": [
          "Шлюзовая наб., 4, Москва, 115114",
          "Rosso Riva"
        ],
        "streetName": "Шлюзовая наб.",
        "buildingNumber": "4",
        "postCode": "115114",
        "townName": "Moscow",
        "countySubDivision": [
          "Moscow"
        ],
        "country": "RU"
      }
    }
  }
}
```

Таблица 55 - Запрос POST /payment

```
HTTP/1.1 201 Created
x-jws-signature:
D4hhdCB3Zfsefe234xQgaGVyZQ0K..dWQgZmFpbHVyZSB0bdsRft5erewf40ZQ0K
x-fapi-interaction-id: 32bae548-f4de-4874-b184-880a4363460d
Content-Type: application/json
{
  "Data": {
    "paymentId": "58923-001",
    "consentId": "58923",
    "status": "AcceptedSettlementInProgress",
    "creationDateTime": "2019-06-05T15:15:13+00:00",
    "statusUpdateDateTime": "2019-06-05T15:15:13+00:00",
    "Initiation": {
      "instructionIdentification": "PISP412",
      "endToEndIdentification": "MERCHANT.256702.IDN.12",
      "InstructedAmount": {
        "amount": "23463.00",
        "currency": "RUB"
      },
      "DebtorAccount": {
        "SchemeName": "RU.CBR.AccountNumber",
        "Identification": "40817810621234567232",
        "Name": "Иван Иванов"
      },
      "CreditorAccount": {
        "schemeName": "RU.CBR.AccountNumber",
        "identification": "40817810621234567890",
        "name": "MERCHANT Inc"
      },
      "RemittanceInformation": {
        "Reference": "CBR-130",
        "Unstructured": "Назначение платежа - оплата за товары. Внутренний код операции 1234567"
      }
    }
  },
}
```

Таблица 56 - Ответ POST /payment

6.6.3.1.3 Получение согласия на проведение платежа

Запрос GET /payment-consents. СППУ выполняет запрос к ППУ на получение списка ранее данных Пользователем согласий на проведения платежей вызовом конечной точки GET /payment-consents без параметров.

Ответ на GET /payment-consents. ППУ возвращает СППУ согласие на проведение платежа с идентификатором (consentId) "58923" имеющего статус "Authorised".

```
GET /payment-consents/58923 HTTP/1.1
Authorization: Bearer 3YotnDFGEjrlzCsicMwPSS
x-fapi-auth-date: Sun, 29 Sep 2019 12:42:08 GMT
x-fapi-customer-ip-address: 106.32.21.32
x-fapi-interaction-id: 32bae548-f4de-4874-b184-880a4363460d
Accept: application/json
```

Таблица 57 - Запрос GET /payment-consents


```
HTTP/1.1 200 OK
x-jws-signature:
D4hhdCB3Zfsefe234xQgaGVyZQ0K..dWQgZmFpbHVyZSB0bdsRft5erewf40ZQ0K
x-fapi-interaction-id: 32bae548-f4de-4874-b184-880a4363460d
Content-Type: application/json

{
  "Data": {
    "consentId": "58923",
    "status": "Authorised",
    "creationDateTime": "2019-06-05T15:15:13+00:00",
    "statusUpdateDateTime": "2019-06-05T15:15:13+00:00",
    "Initiation": {
      "instructionIdentification": "PISP412",
      "endToEndIdentification": "MERCHANT.256702.IDN.12",
      "InstructedAmount": {
        "amount": "23463.00",
        "currency": "RUB"
      },
      "CreditorAccount": {
        "schemeName": "RU.CBR.AccountNumber",
        "identification": "40817810621234567890",
        "name": "MERCHANT Inc"
      },
      "RemittanceInformation": {
        "Reference": "CBR-130",
        "Unstructured": "Назначение платежа - оплата за товары. Внутренний код операции 1234567"
      }
    },
    "Risk": {
      "paymentContextCode": "EcommerceGoods",
      "merchantCategoryCode": "5967",
      "merchantCustomerIdentification": "053598653254",
      "DeliveryAddress": {
        "addressLine": [
          "Шлюзовая наб., 4, Москва, 115114",
          "Rosso Riva"
        ],
        "streetName": "Шлюзовая наб.",
        "buildingNumber": "4",
        "postCode": "115114",
        "townName": "Moscow",
        "countySubDivision": [
          "Moscow"
        ],
        "country": "RU"
      }
    }
  }
}
```

Таблица 58 - Ответ GET /payment-consents

6.6.3.1.4 Получение платежа

Запрос GET /payments. СППУ выполняет запрос к ППУ на получение списка инициированных платежей вызовом конечной точки GET /payments без параметров.

Ответ на GET /payments. ППУ возвращает СППУ платеж с идентификатором (paymentId) "58923-001", имеющий согласие (consentId): "58923" и статус "AcceptedSettlementInProgress".

```
GET /payments/58923-001 HTTP/1.1
Authorization: Bearer 3YotnDFGEjrlzCsicMwPSS
x-fapi-auth-date: Sun, 29 Sep 2019 12:42:08 GMT
x-fapi-customer-ip-address: 106.32.21.32
x-fapi-interaction-id: 32bae548-f4de-4874-b184-880a4363460d
Accept: application/json
```

Таблица 59 - Запрос GET /payment

```
HTTP/1.1 200 OK
x-jws-signature:
D4hhdCB3Zfsefe234xQgaGVyZQ0K..dWQgZmFpbHVyZSB0bdsRft5erewf40ZQ0K
x-fapi-interaction-id: 32bae548-f4de-4874-b184-880a4363460d
Content-Type: application/json

{
  "Data": {
    "paymentId": "58923-001",
    "consentId": "58923",
    "status": "AcceptedSettlementInProgress",
    "creationDateTime": "2019-06-05T15:15:13+00:00",
    "statusUpdateDateTime": "2019-06-05T15:15:13+00:00",
    "Initiation": {
      "instructionIdentification": "PISP412",
      "endToEndIdentification": "MERCHANT.256702.IDN.12",
      "InstructedAmount": {
        "amount": "23463.00",
        "currency": "RUB"
      },
      "DebtorAccount": {
        "SchemeName": "RU.CBR.AccountNumber",
        "Identification": "40817810621234567232",
        "Name": "Иван Иванов"
      },
      "CreditorAccount": {
        "schemeName": "RU.CBR.AccountNumber",
        "identification": "40817810621234567890",
        "name": "MERCHANT Inc"
      },
      "RemittanceInformation": {
        "Reference": "CBR-130",
        "Unstructured": "Назначение платежа - оплата за товары. Внутренний код операции 1234567"
      }
    }
  }
}
```

Таблица 60 - Ответ GET /payment

6.6.3.2 Инициирование пользователем через СППУ

Для данного случая оплаты:

- Пользователь (Платательщик) выбирает реквизиты своего счета во время инициирования платежа (при создании ресурса согласия на проведение платежа).

Пользователь Петр Петров намерен инициировать через СППУ перевод денежных средств в размере 23463.00 рублей в пользу Петрова Петра, указав его номер счета "40817810621234567754" .

6.6.3.2.1 Создание согласия на проведение платежа

Запрос POST /payment-consents. СППУ присваивает данной платежной инструкции номер (instructionIdentification) PISP412 и создает на стороне ППУ ресурс согласия на

проведения платежа вызовом конечной точки POST /payment-consents, указывая номер инструкции и реквизиты участников перевода.

Ответ POST /payment-consents. ППУ производит обработку запроса и возвращает идентификатор согласия (consentId) "58923" со статусом "AwaitingAuthorisation", указав дату создания (creationDateTime) (дата обновления статуса (statusUpdateDateTime) в данном случае идентична дате создания).

Далее СППУ перенаправляет Пользователя на страницу ППУ и аутентифицирует его. Пользователь производит авторизацию согласия проведения на стороне СППУ инициирования платежа.

```
POST /payment-consents HTTP/1.1
Authorization: Bearer 3YotnDFGEjrlzCsicMwPSS
x-idempotency-key: MERCHANT.256702.IDN.12
x-jws-signature:
UTOmZSdzIGEgam91ckgfyH343YSBkZXN0aW5hdGlvbiA=.D3ghkjk84rk14fIG1yIHR5bGVyI
GdvaW5nIGRvd24gPw==
x-fapi-auth-date: Sun, 29 Sep 2019 12:42:08 GMT
x-fapi-customer-ip-address: 106.32.21.32
x-fapi-interaction-id: 32bae548-f4de-4874-b184-880a4363460d
Content-Type: application/json
Accept: application/json

{
  "Data": {
    "Initiation": {
      "instructionIdentification": "PISP412",
      "InstructedAmount": {
        "amount": "23463.00",
        "currency": "RUB"
      },
      "DebtorAccount": {
        "SchemeName": "RU.CBR.AccountNumber",
        "Identification": "40817810621234567754",
        "Name": "Петр Петров"
      },
      "CreditorAccount": {
        "schemeName": "RU.CBR.AccountNumber",
        "identification": "40817810621234567890",
        "name": "Иван Иванов"
      },
      "RemittanceInformation": {
        "Reference": "CBR-130",
        "Unstructured": "Назначение платежа - перевод денежных средств.
Внутренний код операции 1234567"
      }
    },
    "Risk": {
      "PaymentContextCode": "PartyToParty"
    }
  }
}
```

Таблица 61 - Запрос POST /payment-consents

```
HTTP/1.1 201 Created
x-jws-signature:
D4hhdCB3Zfsefe234xQgaGVyZQ0K..dWQgZmFpbHVyZSB0bdsRft5erewf40ZQ0K
x-fapi-interaction-id: 32bae548-f4de-4874-b184-880a4363460d
Content-Type: application/json

{
  "Data": {
    "consentId": "58923",
    "status": "AwaitingAuthorisation",
    "creationDateTime": "2019-06-05T15:15:13+00:00",
    "statusUpdateDateTime": "2019-06-05T15:15:13+00:00",
    "Initiation": {
      "instructionIdentification": "PISP412",
      "InstructedAmount": {
        "amount": "23463.00",
        "currency": "RUB"
      },
      "DebtorAccount": {
        "SchemeName": "RU.CBR.AccountNumber",
        "Identification": "40817810621234567754",
        "Name": "Петр Петров"
      },
      "CreditorAccount": {
        "schemeName": "RU.CBR.AccountNumber",
        "identification": "40817810621234567890",
        "name": "Иван Иванов"
      },
      "RemittanceInformation": {
        "Reference": "CBR-130",
        "Unstructured": "Назначение платежа - перевод денежных средств.
Внутренний код операции 1234567"
      }
    },
    "Risk": {
      "PaymentContextCode": "PartyToParty"
    }
  }
}
```

Таблица 62 - Ответ POST /payment-consents

6.6.3.2.2 Создание платежа

Запрос POST /payment. СППУ, получив согласие Пользователя на проведение платежа, дополняет платежную инструкцию идентификатором согласия "consentId": "58923" и создает на стороне ППУ ресурс платежа вызовом конечной точки POST /payments.

Ответ на POST /payment-consents. ППУ производит обработку (и процесс исполнения) платежа и возвращает СППУ его идентификатор (paymentId) 58923-001.

```
POST /payments HTTP/1.1
Authorization: Bearer 3YotnDFGEjrlzCsicMWpSS
x-idempotency-key: MERCHANT.256702.IDN.12
x-jws-signature:
UTOmZSdzIGEgam91ckgfyH343YSBkZXN0aW5hdGlvbiA=.D3ghkjk84rk14fIG1yIHR5bGVyI
GdvaW5nIGRvd24gPw==
x-fapi-auth-date: Sun, 29 Sep 2019 12:42:08 GMT
x-fapi-customer-ip-address: 106.32.21.32
x-fapi-interaction-id: 32bae548-f4de-4874-b184-880a4363460d
Content-Type: application/json
Accept: application/json
{
  "Data": {
    "consentId": "58923",
    "Initiation": {
      "instructionIdentification": "PISP412",
      "InstructedAmount": {
        "amount": "23463.00",
        "currency": "RUB"
      },
      "DebtorAccount": {
        "SchemeName": "RU.CBR.AccountNumber",
        "Identification": "40817810621234567754",
        "Name": "Петр Петров"
      },
      "CreditorAccount": {
        "schemeName": "RU.CBR.AccountNumber",
        "identification": "40817810621234567890",
        "name": "Иван Иванов"
      },
      "RemittanceInformation": {
        "Reference": "CBR-130",
        "Unstructured": "Назначение платежа - перевод денежных средств.
Внутренний код операции 1234567"
      }
    },
    "Risk": {
      "PaymentContextCode": "PartyToParty"
    }
  }
}
```

Таблица 63 - Запрос POST /payment

```
HTTP/1.1 201 Created
x-jws-signature:
D4hhdCB3Zfsefe234xQgaGVyZQ0K..dWQgZmFpbHVyZSB0bdsRft5erewf40ZQ0K
x-fapi-interaction-id: 32bae548-f4de-4874-b184-880a4363460d
Content-Type: application/json
{
  "Data": {
    "paymentId": "58923-001",
    "consentId": "58923",
    "status": "AcceptedSettlementInProgress",
    "creationDateTime": "2019-06-05T15:15:13+00:00",
    "statusUpdateDateTime": "2019-06-05T15:15:13+00:00",
    "Initiation": {
      "instructionIdentification": "PISP412",
      "InstructedAmount": {
        "amount": "23463.00",
        "currency": "RUB"
      },
      "DebtorAccount": {
        "SchemeName": "RU.CBR.AccountNumber",
        "Identification": "40817810621234567754",
        "Name": "Петр Петров"
      },
      "CreditorAccount": {
        "schemeName": "RU.CBR.AccountNumber",
        "identification": "40817810621234567890",
        "name": "Иван Иванов"
      },
      "RemittanceInformation": {
        "Reference": "CBR-130",
        "Unstructured": "Назначение платежа - перевод денежных средств.
Внутренний код операции 1234567"
      }
    }
  }
}
```

Таблица 64 - Ответ POST /payment

6.6.3.2.3 Получение согласия на проведение платежа

Запрос GET /payment-consents. СППУ выполняет запрос к ППУ на получение списка ранее предоставленных Пользователем согласий на проведение платежей вызовом конечной точки GET /payment-consents без параметров.

Ответ на GET /payment-consents. ППУ возвращает СППУ согласие на проведение платежа с идентификатором (consentId) "58923" имеющего статус "Consumed".

```
GET /payment-consents/58923 HTTP/1.1
Authorization: Bearer 3YotnDFGEjrlzCsicMwPSS
x-fapi-auth-date: Sun, 29 Sep 2019 12:42:08 GMT
x-fapi-customer-ip-address: 106.32.21.32
x-fapi-interaction-id: 32bae548-f4de-4874-b184-880a4363460d
Accept: application/json
```

Таблица 65 - Запрос GET /payment-consents

```
HTTP/1.1 200 OK
x-jws-signature:
D4hhdCB3Zfsefe234xQgaGVyZQ0K..dWQgZmFpbHVyZSB0bdsRft5erewf40ZQ0K
x-fapi-interaction-id: 32bae548-f4de-4874-b184-880a4363460d
Content-Type: application/json

{
  "Data": {
    "consentId": "58923",
    "status": "Consumed",
    "creationDateTime": "2019-06-05T15:15:13+00:00",
    "statusUpdateDateTime": "2019-06-05T15:15:13+00:00",
    "Initiation": {
      "instructionIdentification": "PISP412",
      "InstructedAmount": {
        "amount": "23463.00",
        "currency": "RUB"
      },
      "DebtorAccount": {
        "SchemeName": "RU.CBR.AccountNumber",
        "Identification": "40817810621234567754",
        "Name": "Петр Петров"
      },
      "CreditorAccount": {
        "schemeName": "RU.CBR.AccountNumber",
        "identification": "40817810621234567890",
        "name": "Иван Иванов"
      },
      "RemittanceInformation": {
        "Reference": "CBR-130",
        "Unstructured": "Назначение платежа - перевод денежных средств.
Внутренний код операции 1234567"
      }
    },
    "Risk": {
      "PaymentContextCode": "PartyToParty"
    }
  }
}
```

Таблица 66 - Ответ GET /payment-consents

6.6.3.2.4 Получение платежа

Запрос GET /payments. СППУ выполняет запрос к ППУ на получение списка инициированных платежей вызовом конечной точки GET /payments без параметров.

Ответ на GET /payments. ППУ возвращает СППУ платеж с идентификатором (paymentId) "58923-001", имеющий согласие (consentId): "58923" и статус "AcceptedSettlementInProgress".

```
GET /payments/58923-001 HTTP/1.1
Authorization: Bearer 3YotnDFGEjrlzCsicMWpSS
x-fapi-auth-date: Sun, 29 Sep 2019 12:42:08 GMT
x-fapi-customer-ip-address: 106.32.21.32
x-fapi-interaction-id: 32bae548-f4de-4874-b184-880a4363460d
Accept: application/json
```

Таблица 67 - Запрос GET /payment

```
HTTP/1.1 200 OK
x-jws-signature:
D4hhdCB3Zfsefe234xQgaGVyZQ0K..dWQgZmFpbHVyZSB0bdsRft5erewf40ZQ0K
x-fapi-interaction-id: 32bae548-f4de-4874-b184-880a4363460d
Content-Type: application/json

{
  "Data": {
    "paymentId": "58923-001",
    "consentId": "58923",
    "status": "AcceptedSettlementInProgress",
    "creationDateTime": "2019-06-05T15:15:13+00:00",
    "statusUpdateDateTime": "2019-06-05T15:15:13+00:00",
    "Initiation": {
      "instructionIdentification": "PISP412",
      "InstructedAmount": {
        "amount": "23463.00",
        "currency": "RUB"
      },
      "DebtorAccount": {
        "SchemeName": "RU.CBR.AccountNumber",
        "Identification": "40817810621234567754",
        "Name": "Петр Петров"
      },
      "CreditorAccount": {
        "schemeName": "RU.CBR.AccountNumber",
        "identification": "40817810621234567890",
        "name": "Иван Иванов"
      },
      "RemittanceInformation": {
        "Reference": "CBR-130",
        "Unstructured": "Назначение платежа - перевод денежных средств.
Внутренний код операции 1234567"
      }
    }
  }
}
```

Таблица 68 - Ответ GET /payment

7 Динамические справочники в пространствах имен - v1.0.0

7.1 Введение

Спецификация определяет поля только с фиксированным набором возможных значений, а дальнейшее добавление значений требует изменение спецификации.

В рамках текущей версии определены новые типы данных, представляющие собой расширяемый список значений. Любые расширения этого стандартного списка значений могут быть сделаны ППУ с соответствующей документацией на их порталах для разработчиков.

Значения расширяемых типов данных располагаются в соответствующих пространствах имен, чтобы помочь идентифицировать источник значения и соответствующее значение.

7.2 Основы

Элементы справочников, определенные участниками Ассоциации развития финансовых технологий, задокументированы здесь и будут иметь префикс RU.CBR.

7.2.1 Принципы проектирования

При добавлении собственных значений в справочники участники среды Открытых банковских интерфейсов с ролью ППУ помещают такие значения в пространство имен, состоящее из двухбуквенного кода страны (код ISO 3166-1 Alpha-2), после которого ставится точка, после которой следует наименование организации. Например:

- RU.SBRF.Int-payments
- RU.VTB.Int-payments
- KE.Safaricom.M-Pesa

7.2.2 Управление версиями

В данном разделе будут описываться требования к обратной совместимости при разработке последующих версий Стандарта.

7.3 Общие справочники

AccountIdentificationDynamicType

Значение	Описание
RU.CBR.PAN	Primary Account Number - схема идентификатора, используемая для идентификации карточного счета.
RU.CBR.CellphoneNumber	Схема для осуществления платежа по номеру телефона.
RU.CBR.BBAN	Схема для осуществления платежа по номеру счета.

Таблица 69 - Идентификатор счета

FinancialInstitutionIdentificationDynamicType

Значение	Описание
RU.CBR.BICFI	BIC для финансовых учреждений согласно ISO 9362.
RU.CBR.BIK	Уникальный идентификатор банка используемый в платежных документах на территории Российской Федерации.

Таблица 70 - Идентификатор финансового учреждения

OBRUErrorResponseErrorCode

Тип данных, который дает низкоуровневый текстовый код ошибки для ее классификации. Применяется также код ответа HTTP.

Значение	HTTP статус	Описание
RU.CBR.Field.Expected	400	<p>Если поля передаются парой (ключ-значение) и значение не было передано.</p> <p>В поле path должен передаваться путь к ожидаемому полю (например, <code>ErrorResponse.Errors.path == "AccountResponse.Data.Account.AccountDetails.identification"</code>).</p> <p>Например, для допустимого значения поля «schemeName» должно передаваться соответствующее значение идентификатора в поле «identification».</p>
RU.CBR.Field.Invalid	400	<p>В поле указано недопустимое значение или длина предоставленного значения превышает соответствующую максимальную длину поля в домене ППУ. Ссылка на недопустимое поле должна быть указана в поле path (например, <code>ErrorResponse.Errors.path == "AccountResponse.Data.Account.AccountDetails.schemeName"</code>). В поле URL может быть ссылка на веб-страницу, объясняющую правильное поведение. Проблема должна быть подробно описана в сообщении об ошибке (поле <code>ErrorResponse.Errors.message</code>).</p>
RU.CBR.Field.InvalidDate	400	<p>Указана неверная дата. Например, когда ожидается будущая дата, а указана дата в прошлом или текущая дата. В сообщении можно указать актуальную проблему с датой. Ссылка на недопустимое поле должна быть указана в поле path, а в поле URL может быть ссылка на веб-страницу, объясняющую правильное поведение.</p>
RU.CBR.Field.Missing	400	<p>Обязательное поле, необходимое для API, отсутствует в полезной нагрузке. Данный код ошибки можно использовать, если ошибка еще не определена при проверке <code>RU.CBR.Resource.InvalidFormat</code>.</p> <p>Ссылка на отсутствующее поле должна быть указана в поле path, а в поле URL может быть</p>

Значение	HTTP статус	Описание
		ссылка на веб-страницу, объясняющую правильное поведение.
RU.CBR.Header.Invalid	400	В элементе заголовка HTTP указано неверное значение. Элемент заголовка HTTP должен быть указан в элементе пути.
RU.CBR.Header.Missing	400	Обязательный элемент HTTP-заголовок не был предоставлен. Элемент заголовка HTTP должен быть указан в элементе path.
RU.CBR.Resource.ConsentMismatch	400	Несоответствие ресурсов «payment-consent» и «payment». Например, если элемент в разделе «Initiation» или «Risk» ресурса платежа не совпадает с одноименным элементом в соответствующем разделе ресурса согласия. Элемент пути должен быть заполнен элементом ресурса платежа, который не соответствует согласию.
RU.CBR.Resource.InvalidConsentStatus	400	Согласие, соответствующее ресурсу, находится в некорректном статусе, который бы позволил создать ресурс или выполнить запрос. Например, если ресурс согласия имеет статус AwaitingAuthorisation или Rejected, то ресурс не может быть создан с таким статусом соответствующего ему согласия. Элемент пути должен быть заполнен элементом ресурса согласия, который является недопустимым.
RU.CBR.Resource.InvalidFormat	400	Когда json-схема полезной нагрузки не соответствует конечной точке. Например, конечная точка POST /payments вызывается с полезной нагрузкой JSON, которая не может быть проанализирована в классе PaymentRequest.
RU.CBR.Resource.NotFound	400	Возвращается, когда ресурс с указанным идентификатором не существует (ресурс не может быть обработан).
RU.CBR.Resource.NotCreated	400	Возвращается, когда ресурс с указанным идентификатором еще не создан и не может быть передан в ответном сообщении. Для асинхронных вызовов. Например, получение выписки по счету, где сначала создается ресурс выписки (метод POST /statements/{accountId}) и в ответном сообщении приходит идентификатор созданного ресурса выписки, но для наполнения выписки данными ППУ требуется некоторое время. Соответственно, будет приходить данное сообщение об ошибке.

Значение	HTTP статус	Описание
RU.CBR.Rules.AfterCutOffDate Time	400	Ресурс согласия или ресурс платежа запрашиваются после даты CutOffDateTime.
RU.CBR.Signature.Invalid	400	Заголовок подписи x-jws-signature был проанализирован и имеет действительный заголовок JOSE, соответствующий спецификации. Но сама подпись не может быть проверена.
RU.CBR.Signature.InvalidClaim	400	Заголовок JOSE в элементе x-jws-signature имеет одно или несколько утверждений (claim) с недопустимым значением. (например, утверждение kid, которое не принимает сертификат). Наименование отсутствующего утверждения должно передаваться в поле path ответа об ошибке.
RU.CBR.Signature.MissingClaim	400	Заголовок JOSE в элементе x-jws-signature имеет одно или несколько обязательных утверждений, которые не указаны. Имя пропущенного утверждения должно быть указано в поле path ответа об ошибке.
RU.CBR.Signature.Malformed	400	x-jws-signature в заголовке запроса была искажена и не могла быть проанализирована как допустимый JWS.
RU.CBR.Signature.Missing	400	Запрос API предполагает x-jws-signature в заголовке, но элемент отсутствовал.
RU.CBR.Unsupported.AccountId entifier	400	Идентификатор счета не поддерживается для данной схемы. Элемент path должен быть заполнен путем к элементу accountId.
RU.CBR.Unsupported.LocalInstr ument	400	Указанный localInstrument не поддерживается ППУ. Элемент path должен быть заполнен путем к элементу localInstrument. Элемент URL должен быть заполнен ссылкой на документацию ППУ со списком поддерживаемых localInstrument.
RU.CBR.Reauthenticate	403	Данный код ошибки указывает, что для обработки запроса требуется повторная аутентификация Пользователя.
RU.CBR.Rules.ResourceAlready Exists	409	Данный код ошибки указывает, что ресурс с такими же параметрами уже существует.
RU.CBR.UnexpectedError	5xx	Данный код ошибки можно использовать при возникновении непредвиденной ошибки. ППУ должен заполнить сообщение детальным описанием ошибки, не раскрывая конфиденциальную информацию.

Таблица 71 – Низкоуровневая классификация ошибок