



[www.rapida.ru](http://www.rapida.ru)

## Общее техническое описание по работе с тестовым сервисом

Код документа: EXT.2.03.07.00001.006  
Версия: 012  
Статус: Действующий  
Дата: 26.10.2015

## Оглавление

История изменений документа.....	2
1. О документе.....	3
2. Термины и определения .....	3
3. Тестирование сервиса по продукту «Платежи» .....	4
3.1 Порядок подготовки к тестированию.....	4
3.2 Общие положения.....	4
3.3 Тестирование передачи параметра с кириллическими символами .....	5
3.4 Запрос на получение справочника ТСП и предустановленных комиссий .....	5
3.5 Запрос на получение остатка средств Участника .....	5
3.6 Запрос на получение кода ТСП .....	5
3.7 Запрос на получение расширенных данных.....	6
3.8 Запрос на получение статуса платежа.....	6
3.9 Запрос на получение тестовых параметров.....	6
3.10 Запрос тестовых параметров по коду ТСП .....	6
3.11 Использование полученных параметров в тестовых запросах.....	8
4. Тестирование сервиса по продукту «Переводы по коду требования» .....	8
4.1 Общие положения.....	8
4.2 Функции протокола .....	9
4.2.1 Служебные функции .....	9
4.2.2 Платежные функции.....	10
4.3 Типы платежных шаблонов .....	10
5. Нестандартные сценарии: оплата билетов и путевок.....	11
6. Рекомендации по настройке ПО Участника .....	14

## История изменений документа

Версия	Дата	Внесенные изменения
001	01.08.2010	Создан новый документ.
002	30.10.2010	Добавлены обязательные параметры тестовых платежей по ГИБДД.
003	30.11.2010	Добавлен раздел «Платежи за виртуальную предоплаченную карту».
004	18.01.2011	Добавлен раздел «Порядок получения тестового сертификата».
005	27.01.2011	Добавлен раздел «1099-Единый код оплаты ГИБДД».
006	03.02.2011	Добавлен раздел «Список кодов получателей платежей, настроенных для проведения тестирования».
007	05.04.2011	Изменена структура документа. Добавлено описание теста проверки передачи кириллицы.
008	15.09.2011	Исправлены и добавлены новые тестовые данные.
009	31.10.2011	Изъят раздел, описывающий получение сертификата в связи с дублированием его в другом документе. Добавлены и обновлены пункты в разделе «Тестирование платежей по продукту «Платежи». Добавлено Приложение 2.
010	25.11.2011	Добавлены описания новых получателей, настроенных в тестовой среде.
011	02.03.2012	Добавлено описание одноразового шаблона для протокола «Платежи по коду требования».
012	01.08.2012	Обновлена информация о параметрах тестирования в адрес ГИБДД.
013	08.02.2013	Приложение 2. Исключен пункт «Запрос статуса платежа».
014	29.03.2013	Добавлен раздел «Расширение функции getfee для тестового сервиса».
015	09.09.2013	Исключены разделы: «Тестирование по оплате штрафов ГИБДД» и «Тестирование оплаты платежей в бюджет и налоговых платежей».
016	27.01.2014	Документ полностью переработан.
017	06.05.2014	Добавлено описание проверки номера платежа, превышающего значение два миллиарда.
018	10.06.2014	В разделе 4.10 расширена информация о зарегистрированных в тестовой среде пунктах приема платежа.
019	03.09.2014	Изъято описание функции на отмену платежа (cancelreq). Добавлен раздел «Рекомендации по настройке ПО Участника».
020	11.08.2015	Добавлен раздел «Нестандартные сценарии. Оплата билетов и путевок».
021	15.10.2015	Раздел «Функциональное тестирование» дополнен цифровыми кодами ответов. Добавлены рекомендации по обработке цифрового кода 55 (нарушены правила финансового или fraud - мониторинга). Наименование Пунктов Приема Платежей приведено в соответствие требованиям основного документа «Протокол ПП Платежи».
022	20.10.2015	Исключен раздел «Функциональное тестирование». Сценарии перенесены в Приложение 1 (Методика тестирования сервиса по продукту «Платежи»). Актуализирована Методика тестирования сервиса по продукту «Платежи».
023	26.10.2015	Приложение 2 (Методика тестирование сервиса по продукту «Переводы по коду требования») дополнено цифровыми кодами ответов.

## 1. О документе

Документ описывает требования и рекомендации использования тестового сервиса компании Рапида, предназначенного для проведения платежей в тестовом режиме, а также определяет методику проведения тестирования при подключении новых Участников.

## 2. Термины и определения

Сокращение	Описание
БИК	Банковский идентификационный код
ИС	Информационная система
Плательщик	Лицо, пользующееся услугами Участника для оплаты услуг ТСП
Рапида	Общество с ограниченной ответственностью Небанковская кредитная организация «Рапида»
СКЗИ	Средство криптографической защиты информации
ТСП	Торгово-сервисное предприятие, получатель платежей в рамках описываемых в документе процессов
Участник	Участник платежной системы Рапиды, осуществляющий прием платежей у Плательщиков в пользу ТСП

### 3. Тестирование сервиса по продукту «Платежи»

#### 3.1 Порядок подготовки к тестированию

Для начала использования тестового сервиса Участнику необходимо:

- Связаться с курирующим менеджером компании Рапида;
- Подтвердить готовность шлюза и наличие соответствующих настроек для проведения платежей;
- Заказать, получить и установить тестовый сертификат. Порядок получения тестового сертификата описан в документе [«Использование сертификатов»](#).

#### 3.2 Общие положения

Основным документом, который описывает правила взаимодействия ИС Участника и ИС компании Рапида по продукту «Платежи», является документ [«Протокол ПП Платежи»](#). Документ доступен на официальном сайте компании Рапида <http://soft.rapida.ru/download/>.

Вызов всех процедур производится по адресам:

- <https://tepay.rapida.ru/test/> для защищённого SSL соединения на основе RSA алгоритмов шифрования;
- <https://tepaygost.rapida.ru/test/> для защищённого SSL соединение на основе ГОСТ алгоритмов шифрования (требуется дополнительная установка СКЗИ; рекомендуется использовать КриптоПро r3, VipNet, КриптоКом).

**Важно!** В тестовой среде зарегистрировано по одному пункту приема платежа на каждый тип устройства. При формировании запросов необходимо передавать в запросе TermID, равный одному из следующих вариантов:

- DOT1 – Касса;
- DOT2 – Банкомат;
- DOT3 - Терминал самообслуживания;
- DOT4 - Банкомат cash-in;
- DOT5 - POS – терминал;
- DOT6 - Мобильный банкинг;
- DOT7 - Интернет банкинг;
- DOT8 - POS – терминал;
- DOT9 - IVR каналы;
- DOT10 – Автоплатежи;
- DOT11 - Интернет-портал.

### 3.3 Тестирование передачи параметра с кириллическими символами

Согласно документу [«Протокол ПП Платежи»](#), все кириллические символы должны передаваться в кодировке win1251. Использование URL-кодирования допускается только в кодировке win1251. Наиболее распространенная ошибка - использование UTF-8 URL-кодирования. Для контроля корректности передачи таких параметров в тестовой среде введен код ТСП «1» – тестирование кириллического параметра с единственным параметром номер один.

```
/?function=check&PaymExtId=3003201&PaymSubjTp=1&Amount=100&TermType=003-09&TermId=DOT3&FeeSum=100&Params=1+%CA%E8%F0%E8%EB%EB%E8%F6%E0
```

Ответ на запрос с правильной кодировкой.

```
<Response>
  <Result>OK</Result>
  <PaymExtId>3003201</PaymExtId>
  <Description>Обработан параметр: Кириллица</Description>
  <Balance>998450,47</Balance>
</Response>
```

Ответ на запрос с неправильной кодировкой.

```
<Response>
  <Result>OK</Result>
  <PaymExtId>3003201</PaymExtId>
  <Description>Обработан параметр: РьРёСЪРёР»Р»РёС+Р°</Description>
  <Balance>998450,47</Balance>
</Response>
```

### 3.4 Запрос на получение справочника ТСП и предустановленных комиссий

Получение справочника ТСП (функция getfee) полностью аналогично соответствующему запросу в продуктивной среде. Перечень ТСП в тестовой среде не зависит от Участника и может отличаться от аналогичного перечня в промышленной среде.

### 3.5 Запрос на получение остатка средств Участника

Получение остатка средств Участника полностью аналогично соответствующему запросу (функция getbalance) в промышленной среде. В тестовой среде остаток, как правило, всегда отрицательный.

### 3.6 Запрос на получение кода ТСП

В тестовой среде запрос на получение кода ТСП (функция getpaymsubj) полностью аналогично соответствующему запросу в промышленной среде.

### 3.7 Запрос на получение расширенных данных

Получение расширенных параметров (функция `getextinfo`) полностью аналогично соответствующему запросу в промышленной среде.

### 3.8 Запрос на получение статуса платежа

Получение статуса платежа (функция `getstate`) полностью аналогично соответствующему запросу в промышленной среде.

### 3.9 Запрос на получение тестовых параметров

При тестировании запросов на проверку параметров платежей и проведения платежей перед Участником встает вопрос получения тестовых параметров и примеров запроса. Для решения этого вопроса необходимо использовать функцию `getttestparams`.

Типовой сценарий работы с функцией состоит из следующих шагов:

1. Запрос тестовых параметров по нужному коду ТСП.
2. Использование полученных параметров для выполнения тестовых запросов.

### 3.10 Запрос тестовых параметров по коду ТСП

Для получения тестовых параметров Участник выполняет вызов процедуры со следующими параметрами:

Описание параметра	Параметр	Формат
Идентификатор запроса	Function	Константа = « <code>getttestparams</code> ».
Уникальный номер Участника	PaymExtId	От 1 до 20 символов.
Код ТСП	PaymSubjTp	Цифровой код ТСП в ИС компании Рапида.

#### Пример

[/?function=getttestparams&paymextid=123456789&PaymSubjTp=101](http://?function=getttestparams&paymextid=123456789&PaymSubjTp=101)

ИС компании Рапида возвращает ответ Участнику в виде XML-документа, содержащего следующие элементы:

Элемент	Атрибут	Описание
Response		Составной элемент блока ответа.
Response.Result		Результат выполнения запроса. Может принимать следующие значения: ОК – функция отработала успешно;

		ERROR – функция не может быть обработана по причине неверного формата запроса, либо отсутствия тестовых данных.
Response.Description		Описание выполнения запроса.
Response.testparams		Блок-элемент информационного блока о ТСП.
Response.testparams	code	Код ТСП.
Response.testparams.field		Элемент информационного блока о тестовых параметрах платежа.
Response.testparams.field	set	Порядковый номер элемента field.
Response.testparams.field	code	Код тестового параметра
Response.testparams.field	value	Значение тестового параметра
Response.testparams.field	result	Описание предустановленного ответа
Response.testparams.example		Элемент информационного блока, содержащий пример запроса.
Response.testparams.example	request	Пример тестового запроса.
Response.testparams.external		Элемент информационного блока, содержащий идентификатор, определяющий возможность проведения платежа со своими реквизитами.

### Успешный ответ

<Response>

<Result>OK</Result>

<Description>Тестовые параметры для данного кода ТСП определены.</Description>

<testparams code="101">

<field set="0" code="188" value="9054697951" result="Результат обработки:OK для запроса check и payment"/>

<field set="1" code="188" value="9604243781" result="Результат обработки:Error для запроса check"/>

<field set="2" code="188" value="9608569942" result="Результат обработки:Timeout для запроса check"/>

<field set="3" code="188" value="9614482711" result="Результат обработки:OK для запроса check, Error для запроса payment"/>

<field set="4" code="188" value="9631689922" result="Результат обработки: OK для запроса check, недостаток дс для запроса payment"/>

<field set="5" code="188" value="9682683591" result="Результат обработки: OK для запроса check, Timeout для первых 5 запросов payment, OK на 6-й запрос payment"/>

<example

request="function=check&PaymExtId=123456abcd&PaymSubjTr=101&Amount=1000&Params=188%209054697951&TermType=003-07&TermID=DOT3&FeeSum=0"/>

<external>1</external>

</testparams>

</Response>

### Неуспешный ответ

<Response>

```
<Result>Error</Result>
<Description>Тестовые параметры для данного кода ТСП не определены.
</Description>
</Response>
```

### 3.11 Использование полученных параметров в тестовых запросах

Для использования полученных параметров в тестовых запросах необходимо:

1. Ознакомиться с описанием результата каждого варианта в значении атрибута «[Response.testparams.field@result](#)» и выбрать запланированный результат.
2. Подставить в формируемый запрос значение основного параметра из атрибута «[Response.testparams.example@request](#)». Код основного параметра представлен в атрибуте «[Response.testparams.field@code](#)», тестовое значение основного параметра представлено в значении атрибута «[Response.testparams.field@value](#)»
3. Использовать сформированный запрос для тестов.

#### Пример

Цель тестирования запроса - «Результат обработки: Ok для запроса check, Error для запроса payment» ([Response.testparams.field@result](#)). Используем параметры из 4-го набора ([Response.testparams.field@set](#) = 3). Согласно данному набору необходимо в параметре с кодом 188 ([Response.testparams.field@code](#)) передать значение «9614482711» ([Response.testparams.field@value](#)).

Набор параметров для запроса после модификации должен выглядеть так:

```
/?function=check&PaymExtId=123456abcd&PaymSubjTp=101&Amount=1000&Params=188%9614482711&TermType=003-07&TermID=DOT3&FeeSum=0
```

## 4. Тестирование сервиса по продукту «Переводы по коду требования»

### 4.1 Общие положения

Основным документом, который описывает правила взаимодействия ИС Участника и ИС компании Рапида, является документ «[Протокол переводы по коду требования](#)». Документ «[Протокол переводы по коду требования](#)» доступен на официальном сайте компании Рапида <http://soft.rapida.ru/download/>.

Вызов всех процедур производится по адресам:

- <https://tepay.rapida.ru/hypertest/> для защищённого SSL соединения на основе RSA алгоритмов шифрования;
- <https://tepaygost.rapida.ru/hypertest/> для защищённого SSL соединения на основе ГОСТ алгоритмов шифрования (требуется дополнительная установка СКЗИ; рекомендуется использовать КриптоПро r3, VipNet, КриптоКом).

## 4.2 Функции протокола

Документ «Протокол переводы по коду требования» определяет два типа функций - служебные и платежные. К первому типу относятся функции `listtid`, `reg`, `check_params` и `payment`, ко второму – `check` и `payment`.

### 4.2.1 Служебные функции

#### Функция *listtid*

Функция `listtid` выводит данные о шаблонах, зарегистрированных у Плательщика. Идентифицирующим параметром является номер телефона Плательщика. Вызов функции `listtid` выводит следующую информацию:

- Все платежные шаблоны, зарегистрированные у Плательщика;
- Только кредитные платежные шаблоны, зарегистрированные у Плательщика;
- Только платежные шаблоны для разовой оплаты, зарегистрированные у Плательщика.

#### Функция *reg*

Функция `reg` производит регистрацию нового Плательщика в ИС компании «Рапида». Регистрация может быть:

- Частичной (для проведения платежей на сумму до 15000 рублей);
- Полной с идентифицирующими параметрами (для проведения платежей на сумму свыше 15000 рублей).

#### Функция *check\_params*

Функция `check_params` выводит по предоставленному БИК данные о ТСП и его дополнительных реквизитах. Вызов данной функции необходим перед регистрацией нового платежного шаблона.

#### Функция *payment*

Функция `payment` служит для отмены платежа день в день, то есть платежа, дата исполнения которого равна дате попытке его отмены.

## 4.2.2 Платежные функции

### Функция *check*

Функция *check* производит авторизацию платежа и регистрацию нового платежного шаблона. Вызов функции производится с разными параметрами, по которым определяется выполняемый функционал. Варианты вызова функции *check* представлены ниже.

- Для авторизации платежа в функцию передается номер платежного шаблона. Номер платежного шаблона может быть получен вызовом функцией *listtid*, либо его предоставляет Плательщик. По информации из шаблона производится проверка реквизитов получателя. При успешном ответе, ИС Рапиды готова принять запрос на платежную транзакцию.
- Для регистрации нового платежного шаблона. На этом шаге в функцию передаются идентифицирующие данные о Плательщике и данные о ТСП, например, БИК банка, номер счета, ФИО владельца счета, а также иные дополнительные параметры, которые требует конкретное ТСП. ИС Рапиды направляет запрос на проверку предоставленных параметров в ТСП, и если все данные корректны, шаблону присваивается номер кода требования и данный шаблон привязывается к номеру телефона Плательщика. Дальнейшая оплата производится с использованием этого кода требования. После регистрации платежного шаблона можно делать вызов функции на проведение оплаты *payment*.

### Функция *payment*

Вызов функции *payment* отправляет запрос на исполнение платежа.

## 4.3 Типы платежных шаблонов

Документ [«Протокол переводы по коду требования»](#) определяет два типа платежных шаблонов – многоразовые и одноразовые. Многоразовые шаблоны используются для многократной оплаты по одному коду требования, например, оплата кредита в банке, одноразовые шаблоны используются для однократного выполнения платежа, например, для оплаты заказа в интернет-магазине. После успешной оплаты по одноразовому шаблону ИС Рапиды его блокирует и совершение повторных платежей по нему невозможно.

Для проведения тестовых платежей по одноразовым шаблонам, следует использовать одноразовый шаблон 0129992972.

**Важно!** В отличие от промышленной среды, в тестовой среде успешный платеж по одноразовому шаблону не блокирует шаблон и код требования.

## 5. Нестандартные сценарии: оплата билетов и путевок

- 5.1 Сценарий работы, при котором требуется при платеже передать только полную сумму заказа. Полную сумму заказа можно получить путем выполнения запроса на проверку параметров check с произвольной суммой, отличной от полной суммы заказа. Полная сумма заказа составляет 20000 рублей. Для тестирования используется код назначения 777998.

Параметры кода:

Код параметра	Название	Маска проверки	Обязательность
1	Номер заказа	^\d{3,20}\$	да
2	Номер телефона	^\d{10}\$	нет

### Пример запроса и ответа с неполной суммой

```
/?Function=check&PaymExtId=testnumber1&PaymSubjTp=777998&Amount=1000&Params=1+1234567890;2+9091694587&TermType=002-22&TermId=DOT1
```

```
<Response>
  <Result>Error</Result>
  <PaymNumb>300000016332</PaymNumb>
  <ErrCode>14</ErrCode>
  <PaymExtId>testnumber1</PaymExtId>
  <Description> $amount$20000$amount$ Сумма платежа, равная 200р. не соответствует сумме заказа 20000р.</Description>
  <TechInfo> $amount$20000$amount$ Сумма платежа, равная 200р. не соответствует сумме заказа 20000р.</TechInfo>
  <Balance>301127.05</Balance>
</Response>
```

### Пример запроса и ответа с полной суммой

```
/?Function=payment&PaymExtId=testnumber2&PaymSubjTp=777998&Amount=2000000&Params=1+1234567890;2+9091694587&TermType=002-22&TermId=DOT1
```

```
<Response>
  <Result>OK</Result>
  <PaymNumb>300000016781</PaymNumb>
  <ErrCode>0</ErrCode>
  <PaymExtId>testnumber2</PaymExtId>
  <BillRegId></BillRegId>
  <PaymDate>2015-08-11 09:29:25</PaymDate>
```

```

<Description>$amount$20000$amount$ Платеж по заказу 1234567890 на сумму
20000р. успешно исполнен КВИТАНЦИЯ / ЭЛЕКТРОННЫЙ
БИЛЕТ| |*****
**||Код брони: 9ЦСВДС||Маршрут:||Уфа-Москва, Внуково||Рейс: УТ0366, вылет 31.12.2015
22:00||Класс: U||№ билета: 298 6158619916 1||Пассажир: Г-жа Иванов/Иван || Mr
Иванов/Иван ||Статус: Подтвержден||Тариф: 20000RUB||Форма оплаты:
наличные| |*****
*****||Регистрация в аэропорту осуществляется||при предъявлении
документа,||удостоверяющего личность пассажира,||который предъявлялся при
оформлении||брони.||Возврат денежных средств за||неиспользованные проездные
документы||осуществляется после обращения в||компанию по указанным
телефонам$url$https://www.starndartur1.ru/ORD123456789.htm$url$</Description>
<Balance>1250508.95</Balance>
</Response>

```

**Важно!** PaymExtId в каждом запросе должен быть уникальным. В противном случае будет возвращаться ответ первого запроса.

5.2 Сценарий работы, при котором допускается оплата заказа частями. Полная сумму заказа, а также сумма, которую необходимо оплатить для получения маршрут-квитанции, отдается на каждый запрос. Для тестирования используется код назначения 777999.

Параметры кода:

Код параметра	Название	Маска проверки	Обязательность
1	Номер заказа	^\d{3,20}\$	да
2	Номер телефона	^\d{10}\$	нет

**Пример запроса и ответа с неполной суммой**

```

/?Function=payment&PaymExtId=testnumber3&PaymSubjTp=777999&Amount=10000&Params=1+1234
5678901;2+9091694587&TermType=002-22&TermId=DOT1

```

```

<Response>
<Result>OK</Result>
<PaymNumb>300000017173</PaymNumb>
<ErrCode>0</ErrCode>
<PaymExtId>testnumber3</PaymExtId>
<BillRegId/>
<PaymDate>2015-08-12 10:29:27</PaymDate>
<Description>
$amount$20000$amount$ Платеж на сумму 100р. успешно исполнен. Оставшаяся сумма к
оплате по заказу 12345678901 равна 19900.00р. ||Пассажиры:|| Г-н Иванов/Иван -
Взрослый||Маршрут:|| Эконом|| Уфа-Москва U, Рейс: УТ-0366, Вылет: 31.12.2015 в 22:00
</Description>
<Balance>1250193.95</Balance>
</Response>

```

**Пример запроса и ответа с полной суммой**

```
/?Function=payment&PaymExtId=testnumber4&PaymSubjTp=777999&Amount=2000000&Params=1+12345678901;2+9091694587&TermType=002-22&TermId=DOT1
```

```
<Response>
<Result>OK</Result>
<PaymNumb>300000017178</PaymNumb>
<ErrCode>0</ErrCode>
<PaymExtId>testnumber4</PaymExtId>
<BillRegId/>
<PaymDate>2015-08-12 10:31:54</PaymDate>
<Description>
$amount$20000$amount$ Платеж по заказу 12345678901 на сумму 20000р. успешно исполнен.
Заказ полностью оплачен. КВИТАНЦИЯ / ЭЛЕКТРОННЫЙ
БИЛЕТ|*****
**||Код брони: 9ЦСВДС||Маршрут:||Уфа-Москва, Внуково||Рейс: UT0366, вылет 31.12.2015
22:00||Класс: U||№ билета: 298 6158619916 1||Пассажир: Г-жа Иванов/Иван || Mr
Иванов/Иван ||Статус: Подтвержден||Тариф: 20000RUB||Форма оплаты:
наличные|*****
*****||Регистрация в аэропорту осуществляется||при предъявлении
документа,||удостоверяющего личность пассажира,||который предъявлялся при
оформлении||брони.||Возврат денежных средств за||неиспользованные проездные
документы||осуществляется после обращения в||компанию по указанным
телефонам$url$https://www.starndartur1.ru/ORD123456789.htm$url$
</Description>
<Balance>1230493.95</Balance>
</Response>
```

**Пример запроса и ответа с суммой, превышающей сумму заказа**

```
/?Function=payment&PaymExtId=testnumber5&PaymSubjTp=777999&Amount=4000000&Params=1+12345678901;2+9091694587&TermType=002-22&TermId=DOT1
```

```
<Response>
<Result>Error</Result>
<PaymNumb>300000017193</PaymNumb>
<ErrCode>14</ErrCode>
<PaymExtId>testnumber5</PaymExtId>
<Description>
$amount$20000$amount$ Сумма оплаты 40000р. превышает сумму заказа. Вам необходимо
оплатить 20000р.
</Description>
<TechInfo>
$amount$20000$amount$ Сумма оплаты 40000р. превышает сумму заказа. Вам необходимо
оплатить 20000р.
</TechInfo>
<Balance>1230493.95</Balance>
</Response>
```

**Важно!** Минимальная сумма платежа составляет 1 рубль. Если сумма меньше указанного значения, то в ответе `ter description` не будет содержать информации о заказе.

## 6. Рекомендации по настройке ПО Участника

### Исходные данные:

- ПО Участника использует двухшаговая схема проведения платежа с предварительной авторизацией;
- произведено успешное тестирование базовых возможностей ПО Участника в тестовой среде согласно методики тестирования.

### Основные рекомендации по настройке ПО Участника:

- Настроить проверку значений параметров платежа, а также минимальной и максимальной суммы платежа, вводимых Плательщиком на этапе формирования платежа. Проверка настраивается на основании данных справочника получателей. Описание формата справочника приведено в документе «[Формат справочника Получателей](#)».
- Повторные запросы на этапе авторизации параметров платежа (`function=check`) или этапе исполнения платежа (`function=payment`) должны быть идентичны первичному запросу этого этапа.
- Запрос на исполнение платежа (`function=payment`) должен быть полностью идентичен запросу на авторизацию (`function=check`), за исключением соответствующего изменения значения параметра `function` и добавления в запрос параметра `TermTime`.

Соблюдение данных рекомендаций позволяет избежать ошибок с цифровыми кодами ответов 4, 8, 10, 41, 42 на этапе авторизации параметров платежа или ошибок с цифровыми кодами ответов 2, 4, 5, 8, 10, 11, 41, 42, 55 (и 1, если не будут устанавливаться ограничения безопасности) на этапе исполнения платежа.

Кроме того, рекомендуется принять решение о том, как ПО Участника будет обрабатывать ответы Системы на этапе авторизации параметров платежа, которые означают промежуточный статус запроса (коды ответа 9, 12, 15). Гарантий успешного результата обработки платежа в случае перехода на этап исполнения нет, хотя вероятность получения такого результата достаточно велика.

Принятие решения о продолжении обработки приведет к потенциальному увеличению количества проведенных платежей, но одновременно может привести к увеличению претензионной работы на стороне Участника.

Принятие решения о прекращении обработки приведет к противоположному эффекту.

Также необходимо принять решение о том, как ПО Участника будет обрабатывать ответы Системы на этапе исполнения платежа с кодом ответа 30, которые означают, что остатка денежных средств на балансе Участника недостаточно для исполнения платежа.

### **Настройки ПО для этапа «Проверка правильности параметров платежа»**

Получен ответ с кодом 0. Производится переход на этап исполнения платежа.

Получен ответ с кодом 1, 2, 5, 11, 14, 25, 26, 55. Переход на этап исполнения платежа не производится, обработка платежа останавливается, плательщику сообщается, что проведение платежа в данный момент невозможно.

Получен ответ с кодом 9, 12 или 15. Если ранее было принято решение о продолжении обработки в подобных случаях, то производится переход на этап исполнения платежа.

Если же ранее было принято решение о прекращении обработки в подобных случаях, то переход на этап исполнения платежа не производится, обработка платежа останавливается, плательщику сообщается, что проведение платежа в данный момент невозможно.

### **Настройки ПО для этапа «Запрос на исполнение платежа»**

Получен ответ с кодом 0. Обработка платежа завершается. Плательщику сообщается об успешном исполнении платежа.

Получен ответ с кодом 1. Обработка платежа приостанавливается или останавливается, если Участником были установлены ограничения безопасности при проведении платежей.

Получен ответ с кодом 14, 25, 26, 55. Обработка платежа завершается. Плательщику сообщается об отказе в исполнении платежа.

Получен ответ с кодом 9, 12, 15. Обработка платежа продолжается. Плательщику сообщается, что платеж будет исполнен позднее.

Получен ответ с кодом 30. В зависимости от принятого ранее решения:

- обработка платежа завершается. Плательщику сообщается об отказе в исполнении платежа. ПО Участника переводит платеж в финальный статус, исключающий повторную отправку запроса.
- обработка платежа продолжается. Плательщику сообщается, что платеж будет исполнен позднее.