



## eKassir v3 Protocol

---

Версия: 3.1.0

Дата: 21 февр. 12 г.

## **Изменения**

## Общее описание

Протокол eKassirV3 предназначен для приема платежей с терминалов, точек с кассиром, других точек приема платежей.

Особенности:

1. В качестве транспортного протокола используется HTTP/HTTPS.
2. Запросы и ответы представляют собой xml-сообщения, формируемые на основе xsd-схем.
3. Протокол не использует пакетный режим. Один запрос – одна операция. В ответе приходит результат выполнения операции.
4. В протоколе используется аутентификация точки по паролю или по ЭЦП (см следующий раздел).

## Транспортный уровень

### Общие сведения

Все запросы передаются по протоколу http (версия 1.1, RFC 2616) методом POST. В случае использования другого метода возвращается ошибка 405 (Methodnotallowed). Сообщение протокола eKassirV3 передается в теле запроса. В запросе должен быть заголовок **eKassir-Version**, значение которого всегда 3 (три) – номер версии протокола.

В протоколе eKassir используются следующие способы аутентификации:

1. [ЭЦП и идентификатор точки в заголовке http](#),
2. [По идентификатору точки и паролю](#).

В случае, если не прошла аутентификация, возвращается ошибка 401 (Unauthorized), а в теле ответа текстовое описание ошибки.

Протокол eKassirV3 не поддерживает разделенные (chunked) данные. В случае получения запроса с разделенными данными или с длиной запроса 0 байт возвращается ошибка 415 (Unsupportedmediatype). При обработке запроса может возникнуть внутренняя ошибка сервера. В этом случае возвращается ошибка 500 (Internalservererror), а в теле ответа содержится информация об ошибке сервера. По протоколу eKassirV3 длина запроса на сервер ограничена. По умолчанию ограничение — 10000 байт. Это значение может быть изменено администратором eKassir. В случае превышения ограничения возвращается ошибка 413 (Requestentitytoolarge).

### Кодировка

Сообщения протокола могут быть переданы на сервер в любой кодировке на усмотрение разработчика. Название кодировки, в которой представлено сообщение, передается в заголовке **Content-Type**. Также в этом заголовке передается media-type, который по протоколу eKassir должен быть text/xml.

Например, значение заголовка для кодировки utf-8 будет таким:

**text/xml; charset=utf-8**

В случае отсутствия кодировки в запросе или неверного названия кодировки (например, utg-8 вместо utf-8) будет использоваться кодировка по умолчанию. Необходимо уточнить ее у администратора сервера.

Ответ сервера передается в кодировке запроса.

### Аутентификация по идентификатору точки и ЭЦП

Для аутентификации этим способом на сервер в каждом запросе должны быть переданы идентификатор точки в eKassir и подпись передаваемого сообщения. Также может передаваться тип хеш-функции, которая используется для вычисления подписи сообщения. Идентификатор, подпись и тип хеш-функции передаются в следующих заголовках http:

<b>eKassir-PointID</b>	Идентификатор точки в eKassir. Целое положительное число. Выдается клиенту при подключении к системе.
<b>eKassir-Signature</b>	Подпись сообщения по алгоритму RSA.
<b>eKassir-HashType</b>	Используемая при вычислении подписи хеш-функция:

- MD5

- SHA1

Заголовок необязательный. По умолчанию MD5.

Подпись формируется по следующему алгоритму:

1. Вычисляется хеш MD5 (или SHA1) сообщения
2. Полученный хеш подписывается секретным ключом RSA
3. Результат обрабатывается по алгоритму Base64

Подпись вычисляется перед сжатием сообщения.



**Для разработчиков, использующих php или другой язык, не имеющий строгой типизации данных:**

**Хеш вычисляется после того, как сформированная строка сообщения преобразована в массив байт в выбранной кодировке.**

## Аутентификация по идентификатору точки и паролю

Для аутентификации этим способом на сервер eKassir в каждом запросе должны быть переданы идентификатор точки в eKassir и MD5 хеш пароля.

Идентификатор точки и хеш передаются в следующих заголовках http:

<b>eKassir-PointID</b>	Идентификатор точки в eKassir
<b>eKassir-Password</b>	MD5 хеш пароля

Хеш пароля вычисляется следующим образом:

1. Строка с паролем конвертируется в массив байт в кодировке utf-8
2. Из полученного массива вычисляется хеш MD5
3. Полученный хеш обрабатывается по алгоритму Base64

## Сжатие запросов и ответов

Перед отправкой сообщения по сети клиент может сжать данные по алгоритму gzip или deflate.

Сжатие делается после формирования ЭЦП. Для того чтобы сообщить серверу о методе сжатия запроса, необходимо добавить заголовок **Content-Encoding** в http-запрос со значением **gzip** или **deflate** соответственно. Если серверу не удастся произвести декомпрессию принятых данных или указано неверное значение заголовка **Content-Encoding**, сервер ответит ошибкой 415 (Unsupportedmediatype).

Для уменьшения трафика клиент также может управлять сжатием ответов сервера. Для этого используется заголовок **Accept-Encoding**. В заголовке указывается способ сжатия ответа сервера **gzip** или **deflate** (может использоваться перечисление через ',' (запятую), тогда сервер сам выберет способ сжатия). Если в запросе указан способ, который не поддерживается, сервер отвечает ошибкой 406 (Notacceptable). В заголовке ответа **Content-Encoding** сервер указывает способ, которым сжат ответ.

## Управление временем обработки запроса

Клиентское приложение может сообщать Серверу, какое время отводится на ожидание ответа. Это позволяет Серверу не тратить больше времени на обработку, чем это необходимо, а клиентское приложение может получать информацию о том, что Сервер запрос не обрабатывал вообще. Для этого используется заголовок http-запроса **eKassir-ExpectedTimeout**. Его значение – таймаут в миллисекундах, который дается на обработку запроса. Рекомендуется устанавливать его равным половине времени ожидания ответа сервера в запросе. Например, таймаут запроса по сети 30 секунд, тогда значение заголовка  $30 * 1000 / 2 = 15000$  миллисекунд. В случае, если сервер не начал обрабатывать запрос, то клиент получит ошибку 503 (Service Unavailable).

## Справочник ошибок транспортного уровня

В таблице ниже приведен полный перечень ошибок транспортного уровня.

Код	Название	Описание
401	Unauthorized	Ошибка аутентификации. Текст с описанием ошибки передается в теле ответа. При разработке необходимо убедиться в том, что все заголовки переданы корректно, что использованы правильные ключи (или пароль). После передачи в эксплуатацию этот код ответа означает, что пользователи указали аутентификационные данные в настройках клиентской части.
405	Methodnotallowed	Разработчик использовал неверный метод запроса. По протоколу eKassir должен использовать метод POST.
406	Not acceptable	В заголовке <b>Accept-Encoding</b> запроса передан способ сжатия, который не поддерживается сервером.
413	Request entity too large	Длина запроса превышает максимально разрешенную. Обратитесь к администратору сервера, чтобы увеличить максимальную длину запроса.
415	Unsupportedmediatype	Данная ошибка может возникнуть по следующим причинам: <ol style="list-style-type: none"> <li>1. Запрос нулевой длины,</li> <li>2. Запрос с разделенными данными (chunked). По протоколу такие запросы не поддерживаются.</li> <li>3. В запросе передано неверное значение заголовка <b>Content-Encoding</b>.</li> </ol>
500	Internal server error	Произошла внутренняя ошибка сервера. В теле ответа передаются данные об ошибке, которые следует передать разработчикам серверной части. При получении этой ошибки следует повторить запрос через 5 минут.
503	Service unavailable	Ошибка может возникнуть в случае, если сервер перегружен. Эта ошибка означает, что сервер вообще не обработал запрос.

## Сценарии взаимодействия

В этом разделе описаны сценарии приема платежа с участием плательщика. При описании взаимодействия используются следующие термины:

<b>Плательщик</b>	лицо, осуществляющее платеж
<b>Клиент</b>	Приложение, обеспечивающее взаимодействие с Плательщиком
<b>Сервер</b>	Серверное приложение, обслуживающее запросы Клиента

### Основной сценарий при приеме платежа

1. Плательщик выбирает услугу.
2. Клиент предлагает плательщику заполнить параметры платежа.
3. Плательщик заполняет параметры платежа.
4. Клиент подтверждает, что параметры платежа корректны, и отправляет заявку проверки параметров платежа на Сервер.
5. Сервер ставит заявку в очередь и отвечает, через какое время заявка будет обработана.
6. Клиент ждет отведенное время и отправляет запрос результатов проверки номера на Сервер.
7. Сервер подтверждает, что параметры платежа корректны.
8. Клиент предлагает Плательщику оплатить услугу.
9. Плательщик оплачивает услугу.
10. Клиент отправляет платеж на сервер.
11. Сервер принимает платеж и ставит его в очередь обработки.

#### Расширения:

4а. Параметры платежа некорректны:

4а1. Клиент предлагает Плательщику исправить некорректные данные.

4б. Услуга не поддерживает проверку параметров платежа:

4б1. Клиент переходит к шагу 8.

4в.6а. Не удалось отправить запрос проверки на Сервер (или запрос получения результатов проверки), а проверка параметров платежа по услуге обязательна:

4в1, 6а1. Клиент сообщает плательщику, что проверка параметров платежа временно недоступна, предлагает попробовать позднее.

4г. 6б. Не удалось отправить запрос проверки на Сервер (или запрос получения результатов проверки), а проверка параметров платежа по услуге возможна, но необязательна:

4г1. 6б1. Клиент сообщает плательщику, что проверка параметров платежа временно невозможна и переходит к шагу 8.

7а. Сервер сообщает, что параметры платежа некорректны:

7а1. Клиент сообщает плательщику, что параметры платежа некорректны и предлагает исправить некорректные данные.

7б. Сервер сообщает, что проверка параметров платежа невозможна по техническим причинам, а проверка параметров платежа по услуге обязательна:

7б1. Клиент сообщает плательщику, что проверка параметров платежа временно недоступна, предлагает попробовать позднее.

7в. Сервер сообщает, что проверка параметров платежа невозможна по техническим причинам, а проверка параметров платежа по услуге возможна, но необязательна:

7в1. Клиент сообщает плательщику, что проверка параметров платежа временно невозможна и переходит к шагу 8.

9а. Плательщик отказался от оплаты:

9а1. Сценарий прекращается.

10а. Не удалось отправить платеж на Сервер:

10а1. Клиент сохраняет платеж, чтобы отправить его позже.



## Команды протокола

### Общее описание

Все типы протокола обмена описаны в схеме eKassirV3Protocol.xsd.

В протоколе все запросы имеют общий базовый тип Request.

Подобно запросам все ответы также имеют общий базовый тип Response. Ответы на все запросы наследуют базовому типу. Базовый тип не содержит никаких параметров.

### Ответ ErrorResponse

В протоколе определен тип ответа ErrorResponse, с помощью которого сервер сообщает об ошибке обработки запроса. Данный тип ответа может приходить на любой ответ. Коды ошибок который приходят

Пример ответа с ошибкой:

```
<?xml version="1.0" encoding="utf-8"?>
<Response xmlns:xsi="http://www.w3.org/2001/XMLSchema-
instance" xmlns:xsd="http://www.w3.org/2001/XMLSchema" xsi:type="ErrorResponse" xmlns="
http://ekassir.com/eKassir/PaySystem/Server/eKassirV3Protocol">
  <Error Number="11" Description="Cancel unavailable. Payment has been rejected already
." IsFatal="false" xmlns="" />
</Response>
```

## Проверка параметров платежа

Проверка параметров платежа происходит асинхронно в два этапа. На первом этапе производится регистрация заявки на возврат платежа (RegisterCheckRequest). Признаком успешной регистрации является ответ (RegisterCheckResponse), в котором указан таймаут, через который рекомендуется проверить статус заявки при помощи запроса результатов проверки параметров платежа (GetCheckResultRequest). В ответе на данный запрос (GetCheckResultResponse) сервер возвращает текущий статус обработки. Если статус заявки не является конечным, клиенту следует повторно запросить получение результатов через рекомендуемый в ответе таймаут.

## Взаимодействие

Схема вызовов:



### Запрос RegisterCheckRequest

Этот запрос используется клиентом для проверки параметров платежа. Сервер в ответе предоставляет информацию о том, когда заявка будет обработана системой. Для того чтобы получить результаты проверки, клиент должен отправить запрос результатов проверки параметров платежа.

Пример запроса, который содержит один параметр платежа (номер телефона):

```
<?xml version="1.0" encoding="utf-8"?>
<Request xmlns:xsi="http://www.w3.org/2001/XMLSchema-
instance" xmlns:xsd="http://www.w3.org/2001/XMLSchema" xsi:type="RegisterCheckRequest"
Id="e8e1ebdb-420e-4ed5-8081-
23902d48c383" Service="4" xmlns="http://ekassir.com/eKassir/PaySystem/Server/eKassirV3P
rotocol">
  <PaymentParameters xmlns="">
    <Parameter Name="account" Value="9999999999" />
  </PaymentParameters>
</Request>
```

### Ответ RegisterCheckResponse

В ответ на запрос RegisterCheckRequest сервер возвращает ответ, в котором сообщает через какое время клиенту рекомендуется запросить статус обработки запроса проверки параметров платежа.

Пример успешного ответа:

```
<?xml version="1.0" encoding="utf-8"?>
<Response xmlns:xsi="http://www.w3.org/2001/XMLSchema-
instance" xmlns:xsd="http://www.w3.org/2001/XMLSchema" xsi:type="RegisterCheckResponse"
GetCheckResultTimeout="PT5.217S" xmlns="http://ekassir.com/eKassir/PaySystem/Server/eK
assirV3Protocol" />
```

### Запрос GetCheckResultRequest

Запрос предназначен для получения результатов проверки параметров платежа. Для выполнения этого запроса предварительно необходимо отправить запрос RegisterCheckRequest.

Пример запроса:

```
<?xml version="1.0" encoding="utf-8"?>
<Request xmlns:xsi="http://www.w3.org/2001/XMLSchema-
instance" xmlns:xsd="http://www.w3.org/2001/XMLSchema" xsi:type="GetCheckResultRequest"
Id="e8e1ebdb-420e-4ed5-8081-
23902d48c383" xmlns="http://ekassir.com/eKassir/PaySystem/Server/eKassirV3Protocol" />
```

### Ответ GetCheckResultResponse

В ответе на запрос GetCheckResultRequest сервер возвращает результат проверки параметров платежа. В случае успешной проверки сервер возвращает параметры, которые были высланы в запросе регистрации заявки на проверку, а также параметры, которые вернул поставщик услуги, чтобы показать их плательщику. Заявка на проверку параметров еще может находиться не в

конечном статусе, в этом случае, клиенту следует повторить запрос GetCheckResultRequest через рекомендуемый в ответе таймаут.

Статусы проверки параметров платежа:

Статус	Категория	Описание
0	Конечный	Параметры платежа корректны
1	Конечный	Параметры платежа не корректны
2	Конечный	Проверка параметров платежа невозможна.
3	Промежуточный	Проверка параметров еще не выполнена. Запросите результаты позже

Пример ответа:

```
<?xml version="1.0" encoding="utf-8"?>
<Response xmlns:xsi="http://www.w3.org/2001/XMLSchema-
instance" xmlns:xsd="http://www.w3.org/2001/XMLSchema" xsi:type="GetCheckResultResponse
" State="0" Description="Аккаунт существует" GetCheckResultTimeout="PT0S" MaxPaymentSum
m="1000.00" xmlns="http://ekassir.com/eKassir/PaySystem/Server/eKassirV3Protocol">
  <PaymentParameters xmlns="">
    <Parameter Name="account" Value="9999999999" />
    <Parameter Name="documentnumber" Value="123456" />
  </PaymentParameters>
</Response>
```

## Проведение платежа

Сервер платежей производить обработку платежа асинхронно. Для проведения платежа клиенту необходимо зарегистрировать платеж на сервере с помощью запроса (SendPaymentRequest). После получения данного запроса сервер производит первичную проверку платежа и в случае успеха отвечает клиенту о том, что запрос был поставлен в очередь обработки (SendPaymentResponse). Если платеж не пройдет первичную проверку сервер возвращает клиенту ответ (ErrorResponse), в котором содержится описание ошибки. После успешной регистрации платежа в системе, в виду его асинхронной отправки поставщику услуги, при необходимости клиент должен воспользоваться запросом статуса платежа через рекомендуемый в ответе на регистрацию платежа (SendPaymentResponse) таймаут.

Схема взаимодействия:



### Запрос SendPaymentRequest

Запрос предназначен для отправки платежа на Сервер. Сервер ставит платеж в очередь обработки и формирует ответ в случае успешной первичной проверки.

Пример запроса:

```
<?xml version="1.0" encoding="utf-8"?>
<Request xmlns:xsi="http://www.w3.org/2001/XMLSchema-
instance" xmlns:xsd="http://www.w3.org/2001/XMLSchema" xsi:type="SendPaymentRequest" Id
="593b2bc2-4cd7-40d2-9f0c-
4d610d39e5f9" Service="4" Ticket="212506755" Number="16366833241247828975" Time="2012-
01-
26T16:38:21.1449243+04:00" EncashmentNumber="142" Value="100" Commission="0" Currency="
RUB" PaymentTool="5" xmlns="http://ekassir.com/eKassir/PaySystem/Server/eKassirV3Protoc
ol">
  <PaymentParameters xmlns="">
    <Parameter Name="account" Value="9999999999" />
  </PaymentParameters>
  <Cheques xmlns="" />
  <FiscalData FRNumber="2242" EKLZNumber="105" DocumentNumber="3743" OperationDate="201
2-01-
26T15:49:47.4419243+04:00" ShiftNumber="25" SaleAmountByCash="12" SaleAmountByCard="14"
OperationNumber="39" INN="666666666666" xmlns="">
    <Parameters>
      <Parameter Name="12" Value="12" />
    </Parameters>
    <Extension agent="gwtest" />
  </FiscalData>
</Request>
```

### Ответ SendPaymentResponse

Данный ответ возвращается, только если платеж успешно поставлен в очередь обработки. Если платеж не принят сервером будет возвращен ответ [ErrorResponse](#).

Пример успешного ответа:

```
<?xml version="1.0" encoding="utf-8"?>
<Response xmlns:xsi="http://www.w3.org/2001/XMLSchema-
instance" xmlns:xsd="http://www.w3.org/2001/XMLSchema" xsi:type="SendPaymentResponse" P
aymentId="1810629" PaymentTime="2012-01-
26T16:38:21.2+04:00" OperatorId="1" ContractId="1" CheckStateTimeout="PT0.22S" Availabl
eBalance="-2259447181.50" CurrentBalance="-
2259447281.50" Currency="RUR" NeedBlock="false" xmlns="http://ekassir.com/eKassir/PaySy
stem/Server/eKassirV3Protocol" />
```

## Отмена платежа

Отмена платежа используется для того, чтобы отменить завершенный платеж или платеж, который еще не был отправлен поставщику услуги. Отмена платежа поддерживается не всеми поставщиками услуг, также платеж можно отозвать только в течение тех же суток, когда он был отправлен на сервер. Для регистрации заявки на отмену платежа клиент посылает запрос на отмену платежа (CancelPaymentRequest), сервер производит проверку возможности отзыва платежа и возвращает ответ о том что платеж поставлен в очередь на отмену . Если платеж не пройдет проверку сервер возвращает клиенту ответ(ErrorResponse), в котором содержится описание ошибки. После успешной регистрации отмены платежа в системе, в виду асинхронности отмены платежа у поставщика услуги, при необходимости клиент должен воспользоваться запросом статуса платежа через рекомендуемый в ответе на регистрацию отмены платежа(CancelPaymentResponse) таймаут.

Схема взаимодействия:



### Запрос CancelPaymentRequest

Запрос отмены платежа используется для того, чтобы попытается поставить платёж очередь на отмену у поставщика, если он завершен на сервере или отменить платеж, который еще не был отправлен поставщику. Запрос отмены платежа может посылать только та точка, которая произвела его регистрацию на сервере.

Пример запроса:

```
<?xml version="1.0" encoding="utf-8"?>
<Request xmlns:xsi="http://www.w3.org/2001/XMLSchema-
instance" xmlns:xsd="http://www.w3.org/2001/XMLSchema" xsi:type="CancelPaymentRequest"
Id="593b2bc2-4cd7-40d2-9f0c-
4d610d39e5f9" xmlns="http://ekassir.com/eKassir/PaySystem/Server/eKassirV3Protocol" />
```

### Ответ CancelPaymentResponse

Если в ответе сервера нет ошибки, то запрос принят к обработке. После успешной операции регистрации отмены необходимо проверять статус платежа, используя запрос проверки статуса платежа.

Пример успешного ответа:

```
<?xml version="1.0" encoding="utf-8"?>
<Response xmlns:xsi="http://www.w3.org/2001/XMLSchema-
instance" xmlns:xsd="http://www.w3.org/2001/XMLSchema" xsi:type="CancelPaymentResponse"
CheckStateTimeout="PT0S" xmlns="http://ekassir.com/eKassir/PaySystem/Server/eKassirV3P
rotocol" />
```



## Проверка статуса платежа

Так как проведение или отмена платежа производятся сервером асинхронно, то платеж в результате обработки изменяет свой статус во времени независимо от клиента. Для того чтобы конечному клиенту получить статус платежа в платёжной системе, ему необходимо воспользоваться запросом статуса платежа. После получения такого запроса сервер произведет поиск запрашиваемого платежа, и в случае если запрашиваемый платеж будет найден, и его регистрация производилась данной платёжной точкой, сервер вернет ответ, содержащий текущий статус платежа и дополнительные параметры, которые образовались в платеж в результате его обработки. В противном случае сервер вернет ответ с описанием ошибки о том, что платеж не найден. Статусы платежа делятся на две группы: конечные и промежуточные. При получении клиентом промежуточного статуса, клиент обязан повторять запрос `CheckStatusRequest` через таймаут, рекомендованный в ответе.

Статусы платежа:

Статус	Категория	Описание
0	Промежуточный	Принят сервером к обработке, но еще не обрабатывался
1	Конечный	Отвергнут, так как его параметры некорректны (не существуют у поставщика услуги),
2	Конечный	Отвергнут по другим причинам (см. описание статуса),
3	Промежуточный	Отправлен поставщику услуги, но результат обработки еще не получен
4	Промежуточный	Отложен, так как невозможно отправить поставщику
5	Конечный	Завершен.
6	Промежуточный	Отзывается у поставщика услуги
7	Конечный	Отменен
8	Конечный	Возвращен плательщику

## Запрос CheckStatusRequest

Запрос используется для проверки результата обработки платежа. Данный запрос предназначен для получения текущего статуса платежа, после запросов `SendPaymentRequest` или `CancelPaymentRequest`.

Пример запроса:

```
<?xml version="1.0" encoding="utf-8"?>
<Request xmlns:xsi="http://www.w3.org/2001/XMLSchema-
instance" xmlns:xsd="http://www.w3.org/2001/XMLSchema" xsi:type="CheckStatusRequest" Id
="593b2bc2-4cd7-40d2-9f0c-
4d610d39e5f9" xmlns="http://ekassir.com/eKassir/PaySystem/Server/eKassirV3Protocol" />
```

## Ответ CheckStatusResponse

Данный ответ возвращается на запрос `CheckStatusRequest`. В ответе содержится информация о платеже и статус платежа. Пример ответа, в котором платеж находится в промежуточном статусе:

```
<Response xmlns:xsi="http://www.w3.org/2001/XMLSchema-
instance" xmlns:xsd="http://www.w3.org/2001/XMLSchema" xsi:type="CheckStatusResponse"
  PaymentId="1810629" StateTime="2012-01-26T16:51:42.61+04:00" State="4"
  StateComment="Отложен, так как невозможно отправить поставщику"
  OperatorId="1" ContractId="1" NextCheckStateTimeOut="PT1M30S" xmlns="http://ekassir.com
/eKassir/PaySystem/Server/eKassirV3Protocol">
  <PaymentParameters xmlns="">
    <Parameter Name="account" Value="999999999" />
  </PaymentParameters>
</Response>
```

Пример ответа, в котором платеж успешно зачислен получателю:

```
<Response xmlns:xsi="http://www.w3.org/2001/XMLSchema-
instance" xmlns:xsd="http://www.w3.org/2001/XMLSchema" xsi:type="CheckStatusResponse" P
aymentId="1810629" StateTime="2012-01-
26T16:51:42.61+04:00" State="5" StateComment="Завершен" OperatorId="1" ContractId="1" N
extCheckStateTimeOut="PT0S" xmlns="http://ekassir.com/eKassir/PaySystem/Server/eKassirV
3Protocol">
  <PaymentParameters xmlns="">
    <Parameter Name="account" Value="999999999" />
  </PaymentParameters>
</Response>
```

Пример ответа, в котором платеж успешно отменен:

```
<Response xmlns:xsi="http://www.w3.org/2001/XMLSchema-
instance" xmlns:xsd="http://www.w3.org/2001/XMLSchema" xsi:type="CheckStatusResponse" P
aymentId="1810629" StateTime="2012-01-
26T16:51:42.61+04:00" State="7" StateComment="Отвергнут, отозван" OperatorId="1" Contra
ctId="1" NextCheckStateTimeOut="PT0S" xmlns="http://ekassir.com/eKassir/PaySystem/Serve
r/eKassirV3Protocol">
  <PaymentParameters xmlns="">
    <Parameter Name="account" Value="999999999" />
  </PaymentParameters>
</Response>
```

## Справочники

### Запрос GetDirectoryRequest

Запрос справочника используется для получения информации об услугах, оплата которых возможна на Сервере для этого клиента, точке приема платежей, счете и пр.

Клиент может запросить справочник с разной детализацией. От уровня детализации зависит размер справочника. В каждой услуге есть номер ее версии. Номер версии меняется, если изменилась информация об услуге. В протоколе также предусмотрен запрос полной информации об одной услуге или группе услуг. Все это позволяет использовать различные стратегии по работе со справочниками в зависимости от типа приложения. Примеры:

Тип приложения	Стратегия
<b>Сервер</b>	Так как сервер использует каналы с высокой пропускной способностью, размер передаваемых данных для него не имеет большого значения, поэтому можно всегда использовать запрос полного справочника.
<b>Мобильный клиент</b>	Как правило, мобильный клиент использует GPRS, в котором ограничена пропускная способность, кроме того на нем нет необходимости печатать чеки с информацией об участниках. Приложение может загружать короткий или средний справочник, а затем запрашивать информацию об услугах, оплата которых необходима. Таким образом, можно существенно сэкономить трафик.

Пример запроса

```
<?xml version="1.0" encoding="utf-8"?>
<Request xmlns:xsi="http://www.w3.org/2001/XMLSchema-
instance" xmlns:xsd="http://www.w3.org/2001/XMLSchema" xsi:type="GetDirectoryRequest" S
ervicesDetalization="4" IncludeRegions="true" IncludeServiceTypes="true" IncludeContrac
ts="true" IncludeOperators="true" IncludeCurrencies="true" IncludeNominals="true" xmlns
="http://ekassir.com/eKassir/PaySystem/Server/eKassirV3Protocol" />
```

### Ответ GetDirectoryRequest

```
<?xml version="1.0" encoding="utf-8"?>
<Response xmlns:xsi="http://www.w3.org/2001/XMLSchema-
instance" xmlns:xsd="http://www.w3.org/2001/XMLSchema" xsi:type="GetDirectoryResponse"
xmlns="http://ekassir.com/eKassir/PaySystem/Server/eKassirV3Protocol">
  <Client Id="1" Name="1" Blocked="false" xmlns="">
    <Requisites LegalName="1" LegalAddress="1" PostAddress="1" TaxpayerIdNumber="1" RRC
ode="1" ContactPhoneNumber="1" />
  </Client>
  <Point Id="1" PointId="7d198045-cf94-4e0b-8764-
d3c3a408f572" Blocked="false" Address="" Name="1-1" CreationDate="2010-09-
09" xmlns="" />
  <Account Id="7" Name="p 1-1" AllowedBalance="-100.00" CurrentBalance="-
2259447281.50" AvailableBalance="-2259447181.50" Currency="RUR" xmlns="" />
  <Contracts xmlns="">
    <Contract Id="1" Number="11111" Date="2010-09-09" />
  </Contracts>
```

```

<Operators xmlns="">
  <Operator Id="1" Name="1" ContractId="1">
    <Requisites LegalName="Unknown" LegalAddress="Unknown" PostAddress="Unknown" Taxp
ayerIdNumber="000000000000" RRCode="000000" ContactPhoneNumber="" />
  </Operator>
  <Operator Id="2" Name="2" ContractId="3">
    <Requisites LegalName="" LegalAddress="" PostAddress="" TaxpayerIdNumber="" RRCod
e="" ContactPhoneNumber="" />
  </Operator>
</Operators>
<Regions xmlns="">
  <Region Id="1" Name="Test" />
</Regions>
<ServiceTypes xmlns="">
  <ServiceType Id="1" Name="" />
</ServiceTypes>
<FullServices xmlns="">
  <Service Id="6" Version="19" ExternalId="" Name="asda" TypeId="0" RegionId="0" Logo=
"" Alias="asd" DefaultOperatorId="1" MinValue="0.00" MaxValue="0.00" Currency="RUR" Che
ckNecessity="2" CheckTimeout="30">
    <Parameters>
      <Parameter Name="test" Value="&lt;[[D/&gt;null" />
      <Parameter Name="TEST33" Value="null" />
    </Parameters>
    <Requisites LegalName="" LegalAddress="" PostAddress="" TaxpayerIdNumber="" RRCod
e="" ContactPhoneNumber="" />
    <PaymentParameters>
      <Parameter Name="account" DisplayName="account" Description="account" Mask="" R
egexp="" DefaultValue="" SendAt="3" Direction="1" IsRequired="true" AvailableValuesRefe
rence="" />
    </PaymentParameters>
    <CommissionLimitations />
  </Service>
  <Service Id="8" Version="37" ExternalId="" Name="Мерафон" TypeId="1" RegionId="1" L
ogo="" Alias="1" DefaultOperatorId="1" MinValue="0.00" MaxValue="30000.00" Currency="RU
R" CheckNecessity="2" CheckTimeout="30">
    <Parameters>
      <Parameter Name="test" Value="&lt;[[D/&gt;null" />
      <Parameter Name="TEST33" Value="null" />
    </Parameters>
    <Requisites LegalName="" LegalAddress="" PostAddress="" TaxpayerIdNumber="" RRCod
e="" ContactPhoneNumber="" />
    <PaymentParameters>
      <Parameter Name="account" DisplayName="123" Description="123" Mask="" Regexp=""
DefaultValue="" SendAt="3" Direction="3" IsRequired="true" AvailableValuesReference=""
/>
      <Parameter Name="1" DisplayName="test" Description="1" Mask="" Regexp="" Defaul
tValue="" SendAt="3" Direction="3" IsRequired="true" AvailableValuesReference="123" />
    </PaymentParameters>
    <CommissionLimitations>
      <CommissionLimitation StartDate="2011-09-
24" Type="4" MinRate="0.000" MaxRate="100.000" MinAmount="0.00" MaxAmount="12.00" Commi
ssionType="1" />
    </CommissionLimitations>
  </Service>
</FullServices>
<Nominals xmlns="">

```

```

    <Nominal NominalId="1" CurrencyCode="810" Value="10.00" Type="1" />
    <Nominal NominalId="2" CurrencyCode="643" Value="50.00" Type="2" />
  </Nominals>
  <Currencies xmlns="">
    <Currency CurrencyCode="643" AlfabtCode="RUB" Description="RUB" />
    <Currency CurrencyCode="810" AlfabtCode="RUR" Description="RUR" />
    <Currency CurrencyCode="840" AlfabtCode="USD" Description="USD" />
  </Currencies>
  <LastEncashment PointId="7d198045-cf94-4e0b-8764-d3c3a408f572" EncashmentId="142" InputeDate="0001-01-01T00:00:00.0000000+04:00" StartDate="2012-01-26T15:35:07.7400000+04:00" EndDate="0001-01-01T00:00:00.0000000+04:00" State="1" xmlns="" />
</Response>

```

## Параметры платежа

Часто возникает необходимость передать на сервер не только номер лицевого счета (как account), но и дополнительные параметры. Так, например, при приеме коммунальных платежей бывает необходимо вбить показания счетчиков учета воды, электричества и пр. Также бывает необходимость передать в ответе сервера на запрос проверки номера фамилию, имя, отчество или другие данные о плательщике и его лицевом счете.

Для этих целей в справочнике передаются описания дополнительных атрибутов платежа.

Дополнительные атрибуты платежа могут передаваться в запросе на сервер, в ответе сервера и на разных этапах обработки платежа. Справочник позволяет автоматически сформировать меню для пользователя в клиентском приложении.

## Формат маски

Маска параметра платежа предназначена для ввода значения пользователем. В маске могут использоваться следующие символы:

[ ] – внутри квадратных скобок может содержаться любое количество символов «0», «9», «A», «a» и «\_». Символ «0» означает обязательную цифру. Количество указанных в маске символов «0» соответствует количеству обязательных цифр для ввода. Например, маска [000] требует у оператора ввода текста длиной три символа. В поле ввода значения нули будут заменены символом «\*». Например, маска [000] заполнит поле ввода строкой «\*\*\*».

Символ «9» означает необязательную цифру. Например, маска [00099] требует ввода строки длиной от 3-х до 5-и цифр.

Символ «A» означает обязательную букву. Например, маска [AAA] требует ввода строки длиной трех букв.

Символ «a» означает необязательную букву. Например, маска [AAAAaa] требует ввода строки длиной от 3-х до 5-и букв.

Символ «\_» означает обязательный символ (букву или цифру).

Любые символы, не заключенные в квадратные скобки, будут выведены в поле ввода значения атрибута. Например, маска +7([000]) [000]-[0000] в поле ввода значения выведет +7(\*\*\*) \*\*\*\_\*\*\*\*. Символы за пределами квадратных скобок не включаются в значение, введенное оператором.

{}- символ, включенный в фигурные скобки, будет отображаться в поле ввода значения атрибутов так же, как если бы он был за пределами квадратных скобок. Но символы внутри фигурных скобок включаются в введенное значение. Например, маски [00]-[00] и [00]{-}[00] выведут в поле ввода одинаковый текст: \*\*-\*\*. Если оператор введет строку «1234», то в первом случае значением атрибута будет «1234», а во втором «12-34».

{}- символ, включенный в фигурные скобки, будет отображаться в поле ввода значения атрибутов так же, как если бы он был за пределами квадратных скобок. Но символы внутри фигурных скобок включаются в введенное значение. Например, маски \*00+-\*00+ и \*00+,\*00+ выведут в поле ввода одинаковый текст: \*\*-\*. Если оператор введет строку «1234», то в первом случае значением атрибута будет «1234», а во втором «12-34».

### Запрос GetServiceRequest

Запрос предназначен для получения полной информации по одной услуге.

Пример запроса:

```
<?xml version="1.0" encoding="utf-8"?>
<Request xmlns:xsi="http://www.w3.org/2001/XMLSchema-
instance" xmlns:xsd="http://www.w3.org/2001/XMLSchema" xsi:type="GetServiceRequest" Id=
"2" xmlns="http://ekassir.com/eKassir/PaySystem/Server/eKassirV3Protocol" />
```

### Ответ GetServiceResponse

В ответе высылается полное описание услуги, если она разрешена клиенту. Параметры услуги смотрите выше.

Пример ответа:

```
<?xml version="1.0" encoding="utf-8"?>
<Response xmlns:xsi="http://www.w3.org/2001/XMLSchema-
instance" xmlns:xsd="http://www.w3.org/2001/XMLSchema" xsi:type="GetServiceResponse" xm
lns="http://ekassir.com/eKassir/PaySystem/Server/eKassirV3Protocol">
  <Service Id="2" Version="32" ExternalId="" Name="2" TypeId="1" RegionId="1" Logo="" A
lias="2" DefaultOperatorId="1" MinValue="0.00" MaxValue="30000.00" Currency="RUR" Check
Necessity="2" CheckTimeout="31" xmlns="">
    <Parameters>
      <Parameter Name="test" Value="&lt;[[D/&gt;null" />
      <Parameter Name="TEST33" Value="null" />
    </Parameters>
    <Requisites LegalName="" LegalAddress="" PostAddress="" TaxpayerIdNumber="" RRCode=
"" ContactPhoneNumber="" />
    <PaymentParameters>
      <Parameter Name="account" DisplayName="account" Description="" Mask="" Regexp=""
DefaultValue="" SendAt="3" Direction="1" IsRequired="true" AvailableValuesReference=""
/>
    </PaymentParameters>
    <CommissionLimitations>
```

```

    <CommissionLimitation StartDate="2011-09-
24" Type="4" MinRate="0.000" MaxRate="100.000" MinAmount="0.00" MaxAmount="12.00" Commi
ssionType="1" />
  </CommissionLimitations>
</Service>
</Response>

```

### Запрос GetServicesRequest

Запрос предназначен для получения полной информации по нескольким услугам.

Пример запроса:

```

<?xml version="1.0" encoding="utf-8"?>
<Request xmlns:xsi="http://www.w3.org/2001/XMLSchema-
instance" xmlns:xsd="http://www.w3.org/2001/XMLSchema" xsi:type="GetServicesRequest" xm
lns="http://ekassir.com/eKassir/PaySystem/Server/eKassirV3Protocol">
  <IdCollection xmlns="">
    <Id>6</Id>
    <Id>3</Id>
    <Id>1</Id>
  </IdCollection>
</Request>

```

### Ответ GetServicesResponse

В ответе высылается полное описание услуг, если они разрешены клиенту. Параметры услуги смотрите выше.

Пример ответа:

```

<?xml version="1.0" encoding="utf-8"?>
<Response xmlns:xsi="http://www.w3.org/2001/XMLSchema-
instance" xmlns:xsd="http://www.w3.org/2001/XMLSchema" xsi:type="GetServicesResponse" x
mlns="http://ekassir.com/eKassir/PaySystem/Server/eKassirV3Protocol">
  <Services xmlns="">
    <Service Id="6" Version="19" ExternalId="" Name="asda" TypeId="0" RegionId="0" Logo
="" Alias="asd" DefaultOperatorId="1" MinValue="0.00" MaxValue="0.00" Currency="RUR" Ch
eckNecessity="2" CheckTimeout="30">
      <Parameters>
        <Parameter Name="test" Value="&lt;[D/&gt;null" />
        <Parameter Name="TEST33" Value="null" />
      </Parameters>
      <Requisites LegalName="" LegalAddress="" PostAddress="" TaxpayerIdNumber="" RRCod
e="" ContactPhoneNumber="" />
      <PaymentParameters>
        <Parameter Name="account" DisplayName="account" Description="account" Mask="" R
egexp="" DefaultValue="" SendAt="3" Direction="1" IsRequired="true" AvailableValuesRefe
rence="" />
      </PaymentParameters>
      <CommissionLimitations />
    </Service>
    <Service Id="1" Version="45" ExternalId="x76" Name="1" TypeId="1" RegionId="1" Logo=
"" Alias="1-
1" DefaultOperatorId="1" MinValue="10.00" MaxValue="30000.00" Currency="RUR" CheckNeces
sity="2" CheckTimeout="30">

```

```

<Parameters>
  <Parameter Name="test" Value="&lt;[[D/&gt;null" />
  <Parameter Name="TEST33" Value="null" />
</Parameters>
<Requisites LegalName="" LegalAddress="" PostAddress="" TaxpayerIdNumber="" RRCod
e="" ContactPhoneNumber="" />
<PaymentParameters>
  <Parameter Name="account" DisplayName="account" Description="Аккаунт" Mask="" R
egexp="" DefaultValue="" SendAt="3" Direction="1" IsRequired="true" AvailableValuesRefe
rence="" />
  <Parameter Name="Date" DisplayName="Дата создания" Description="Дата создания
реестра" Mask="" Regexp="" DefaultValue="" SendAt="3" Direction="3" IsRequired="true" A
vailableValuesReference="">
    <AllowedValue DisplayName="Первое" Value="1" />
    <AllowedValue DisplayName="Второе" Value="2" />
  </Parameter>
  <Parameter Name="attr" DisplayName="attr" Description="X3" Mask="" Regexp="" De
faultValue="" SendAt="1" Direction="2" IsRequired="false" AvailableValuesReference="" /
>
  <Parameter Name="attr2" DisplayName="attr2" Description="" Mask="" Regexp="" De
faultValue="" SendAt="2" Direction="2" IsRequired="false" AvailableValuesReference="" /
>
  <Parameter Name="Chech1" DisplayName="Chech1" Description="" Mask="" Regexp=""
DefaultValue="" SendAt="1" Direction="1" IsRequired="false" AvailableValuesReference=""
/>
  <Parameter Name="Send1" DisplayName="Send1" Description="" Mask="" Regexp="" De
faultValue="" SendAt="2" Direction="1" IsRequired="false" AvailableValuesReference="" /
>
  <Parameter Name="ChechAndSend1" DisplayName="ChechAndSend1" Description="" Mask
="" Regexp="" DefaultValue="" SendAt="3" Direction="1" IsRequired="false" AvailableValu
esReference="" />
</PaymentParameters>
<CommissionLimitations>
  <CommissionLimitation StartDate="2011-09-
24" Type="4" MinRate="0.000" MaxRate="100.000" MinAmount="0.00" MaxAmount="12.00" Commi
ssionType="1" />
</CommissionLimitations>
</Service>
</Services>
</Response>

```



## Получение сдачи

Запрос сдачи по номеру чека (20изначный штрих-код на чеке терминала). Этим запросом терминал получает сдачу, и сервер ее блокирует на таймаут заданный администратором, чтобы исключить повторное использование.

Пример запроса:

```
<?xml version="1.0" encoding="utf-8"?>
<Request xmlns:xsi="http://www.w3.org/2001/XMLSchema-
instance" xmlns:xsd="http://www.w3.org/2001/XMLSchema" xsi:type="GetBalanceRequest" Num
ber="16366833241796027811" xmlns="http://ekassir.com/eKassir/PaySystem/Server/eKassirV3
Protocol" />
```

## Ответ GetBalanceResponse

Пример ответа:

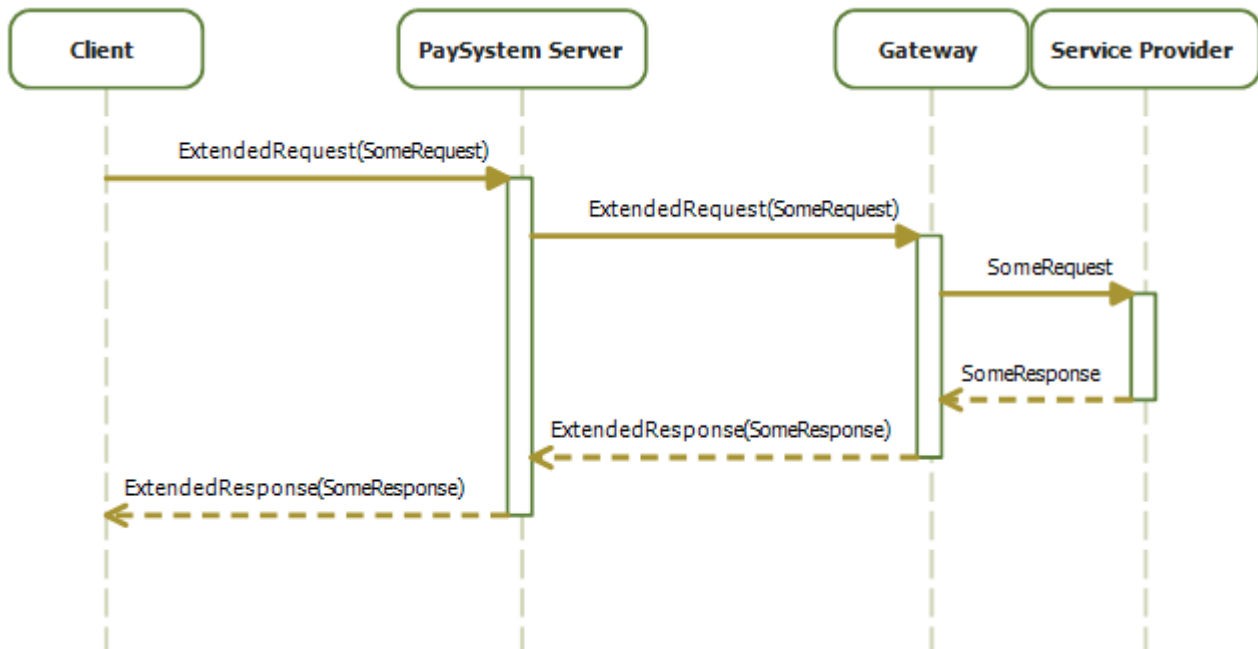
```
<?xml version="1.0" encoding="utf-8"?>
<Response xmlns:xsi="http://www.w3.org/2001/XMLSchema-
instance" xmlns:xsd="http://www.w3.org/2001/XMLSchema" xsi:type="GetBalanceResponse" Ba
lance="10.00" SourceState="0" xmlns="http://ekassir.com/eKassir/PaySystem/Server/eKassi
rV3Protocol" />
```

## Команда передачи внутреннего протокола

eKassir может служить транспортом для другого протокола. Для этой цели служит запрос **ExtendedRequest**. В нем можно передать данные вложенного протокола. Таким образом, возможно, сделать обмен между терминалом и сервером получателя через сервер eKassir по любому протоколу.

## Взаимодействие

Схема вызовов:



Клиент, например терминал, передает запрос PaySystemServer, который далее транслирует вложенные в запрос данные шлюзу. Шлюз формирует запрос поставщику услуги, получает ответ. Ответ передается обратно по цепочке. Если запрос не поддерживается шлюзом, то клиент получит ответ ErrorResponse «Запрос не поддерживается».

## Запрос ExtendedRequest

Пример запроса

```

<?xml version="1.0" encoding="utf-8"?>
<Request xmlns:xsi="http://www.w3.org/2001/XMLSchema-
instance" xmlns:xsd="http://www.w3.org/2001/XMLSchema" xsi:type="ExtendedRequest" Servi
ce="4" xmlns="http://ekassir.com/eKassir/PaySystem/Server/eKassirV3Protocol">RestData</
Request>
  
```

## Ответ ExtendedResponse

Пример ответа

```
<?xml version="1.0" encoding="utf-8"?>
<Response xmlns:xsi="http://www.w3.org/2001/XMLSchema-
instance" xmlns:xsd="http://www.w3.org/2001/XMLSchema" xsi:type="ExtendedResponse" xmln
s="http://ekassir.com/eKassir/PaySystem/Server/eKassirV3Protocol">RestData</Response>
```

## Запрос создания инвентаризации

Данная функциональность используется платежными терминалами, на которых производится работа с фискальными регистраторами. Запрос предназначен для отправки данных инвентаризации на Сервер. Сервер добавляет инвентаризацию в БД и формирует ответ.

### Запрос AddInventoryRequest

Запрос регистрации инвентаризации

Пример запроса:

```
<?xml version="1.0" encoding="utf-8"?>
<Request xmlns:xsi="http://www.w3.org/2001/XMLSchema-
instance" xmlns:xsd="http://www.w3.org/2001/XMLSchema" xsi:type="AddInventoryRequest" E
xternalId="1763516746" Date="2012-01-
26T13:27:25.7986634+04:00" ShiftNumber="1" Amount="3" XReportAmount="3" xmlns="http://e
kassir.com/eKassir/PaySystem/Server/eKassirV3Protocol" />
```

### Ответ AddInventoryResponse

Ответ на запрос регистрации инвентаризации

Пример ответа:

```
<?xml version="1.0" encoding="utf-8"?>
<Response xmlns:xsi="http://www.w3.org/2001/XMLSchema-
instance" xmlns:xsd="http://www.w3.org/2001/XMLSchema" xsi:type="AddInventoryResponse"
Id="114" xmlns="http://ekassir.com/eKassir/PaySystem/Server/eKassirV3Protocol" />
```

## Запрос создания фискальной операции

Данная функциональность используется платежными терминалами, на которых производится работа с фискальными регистраторами. Запрос предназначен для отправки данных фискальной операции произведенной на фискальном регистраторе на Сервер. Сервер добавляет фискальную операцию в БД и формирует ответ.

### Запрос AddFiscalOperationRequest

Запрос регистрации фискальной операции

Пример запроса:

```
<?xml version="1.0" encoding="utf-8"?>
<Request xmlns:xsi="http://www.w3.org/2001/XMLSchema-
instance" xmlns:xsd="http://www.w3.org/2001/XMLSchema" xsi:type="AddFiscalOperationRequ
est" FRNumber="3982" EKLZNumber="4070" DocumentNumber="80" OperationDate="2012-01-
26T13:27:44.8885722+04:00" ShiftNumber="48" OperationType="1" StartAmount="36" Amount="
66" Number="77" Reason="2" InventoryId="21" xmlns="http://ekassir.com/eKassir/PaySystem
/Server/eKassirV3Protocol">
  <Extension param1="value1" xmlns="" />
</Request>
```

### Ответ AddFiscalOperationResponse

Ответ на запрос регистрации фискальной операции

Пример ответа:

```
<?xml version="1.0" encoding="utf-8"?>
<Response xmlns:xsi="http://www.w3.org/2001/XMLSchema-
instance" xmlns:xsd="http://www.w3.org/2001/XMLSchema" xsi:type="AddFiscalOperationResp
onse" Id="59" xmlns="http://ekassir.com/eKassir/PaySystem/Server/eKassirV3Protocol" />
```

## Запрос создания Z отчёта

Данная функциональность используется платежными терминалами, на которых производится работа с фискальными регистраторами. Запрос предназначен для отправки данных Z отчёта на Сервер. Сервер добавляет Z отчёт в БД и формирует ответ.

### Запрос AddZReportRequest

Запрос регистрации Z-отчета

Пример запроса:

```
<?xml version="1.0" encoding="utf-8"?>
<Request xmlns:xsi="http://www.w3.org/2001/XMLSchema-
instance" xmlns:xsd="http://www.w3.org/2001/XMLSchema" xsi:type="AddZReportRequest" FRN
umber="9747" EKLZNumber="3367" OperationNumber="0" KPKNNumber="3955" OperationDate="2012
-01-
26T13:28:11.5772408+04:00" ShiftNumber="70" SaleAmount="10" SaleCount="44" SaleAmountBy
Cash="18" SaleAmountByCard="73" PayOutAmount="25" PayOutCount="91" PayInAmount="94" Pay
InCount="86" CashDeskAmount="23" TotalSaleAmount="59" ShiftStartDate="2012-01-
26T13:28:09.6750506+04:00" xmlns="http://ekassir.com/eKassir/PaySystem/Server/eKassirV3
Protocol">
  <Extension att1="value1" xmlns="" />
</Request>
```

### Ответ AddZReportResponse

Ответ на запрос регистрации Z-отчета

Параметры ответа:

```
<?xml version="1.0" encoding="utf-8"?>
<Response xmlns:xsi="http://www.w3.org/2001/XMLSchema-
instance" xmlns:xsd="http://www.w3.org/2001/XMLSchema" xsi:type="AddZReportResponse" Id
="38" xmlns="http://ekassir.com/eKassir/PaySystem/Server/eKassirV3Protocol" />
```

Пример ответа:

## Система работы через заявки

Некоторая функциональность системы работает через механизм заявок. Данный механизм предназначен для возможности выполнения транзакционных операций. При работе клиент посылает заявку на определённую операцию, сервер при этом проверяет возможность регистрации заявки и при успехе возвращает клиенту идентификатор зарегистрированной заявки с секретным кодом активации (в некоторых случаях сервер блокирует ресурсы для возможности выполнения запрашиваемой операции). После чего клиент производит определённые действия на своей стороне, и в зависимости результатов выполнения операции на своей стороне, посылает на сервер запрос подтверждения или отмены заявки. При получении запроса подтверждения заявки сервер выполняет операцию, которая связана с текущей заявкой и освобождает заблокированные ресурсы. В противном случае при получении запроса отмены заявки, сервер, не выполняя операцию, которая связана с данной заявкой, производит возврат данных в исходное состояние (снимает блокировку с ресурсов). Для возможности работы в распределенной среде, механизм заявок предоставляет функциональность автоматического выполнения действия по заявке по истечению заданного клиентом таймаута, таких как авто исполнение, авто отмена. Данная функциональность управляется клиентом путем задания параметров в запросе на регистрацию заявки любого типа.

## Возврат платежа

Для возможности расчета с клиентами по отвергнутым платежам в платежной системе предусмотрена процедура возврата платежей. Для возврата платежа необходимо чтобы соблюдались следующие условия:

- 1) платеж клиента должен быть отменен
- 2) возврат производился на точке того же агента который проводил возвращаемый платеж (любая точка агента).
- 3) Если платеж имеет цепочку переотправки на сервер, все платежи в этой цепочке должны быть отменены
- 4) На платеже не присутствует глобальная блокировка, связанная с операциями редактирования или переотправки.

## Основной сценарий возврата платежа

1. Плательщик приходит в кассу агента, через которого был произведён платеж и предоставляет чек.
2. Кассир на основании чека производит регистрацию заявки на возврат платежа и посылает его на сервер.
3. Сервер производит поиск платежа по критериям поиска заданным в заявке на возврат.
4. В случае если платеж найден, проверяются условия возврата.
5. В случае успешности проверки условий возврата, сервер накладывает на платеж глобальную блокировку
6. В случае успешности получения глобальной блокировки на платеж сервер регистрирует заявку в системе и посылает ответ на регистрацию заявки, в котором содержится информация по возвращаемому платежу и сумма к возврату.
7. При получении успешного ответа на регистрацию заявки касса выполняет действия связанные с возвратом определённые регламентом агента (операции в кассовом по, возврат денег клиенту и т.д.).
8. В случае успеха оформление возврата на кассе, касса посылает запрос на подтверждения заявки на возврат на сервер.
9. Сервер при получении подтверждения заявки производит возврат платежа на сервере (отмечает платеж как возвращённый ,если у платежа была цепочка переотправки, то все платежи в этой цепочки принимают статус «возвращен первичный платеж», и снимает блокировку платежу)
10. В случае успеха возврата платежа на сервере сервер переводит статус заявки в исполнено и возвращает статус заявки клиенту.
11. Касса получает ответ на подтверждение заявки, проверяет статус заявки из ответа
12. В случае получения статуса заявки как исполненной. Касса считает что отзыв завершён успешно. Сценарий прекращается

## Расширения:



- 8.a В случае не успеха оформление возврата на кассе (клиент отказался, нет денег, ошибка в кассовом ПО и. т. д.), касса посылает на сервер запрос отмены заявки на возврат платежа.
- 9.a Сервер при получении отмены заявки производит снятие глобальной блокировки с платежа.
- 10.a В случае успеха возврата платежа на сервере сервер переводит статус заявки в отменена и возвращает статус заявки клиенту.
- 11.a Касса получает ответ на отмену заявки, проверяет статус заявки из ответа
- 12.a В случае получения статуса заявки как отмененной касса считает, что платеж вернулся в исходное состояние на сервере. Сценарий прекращается

### Запрос RegisterRefundPaymentQueryRequest

Запрос регистрации заявки на возврат платежа

Пример запроса:

```
<?xml version="1.0" encoding="utf-8"?>
<Request xmlns:xsi="http://www.w3.org/2001/XMLSchema-
instance" xmlns:xsd="http://www.w3.org/2001/XMLSchema" xsi:type="RegisterRefundPaymentQ
ueryRequest" OnlyCheck="false" ActivityTime="0001-01-
01T00:00:00" ActivityType="0" PaymentIdOnPoint="56d8bdd0-c0c0-4ffd-a136-
f4975a3999c7" PaymentId="1810631" PointId="0" PointGuidId="00000000-0000-0000-0000-
000000000000" PaymentDate="0001-01-
01T00:00:00" Ticket="0" xmlns="http://ekassir.com/eKassir/PaySystem/Server/eKassirV3Pro
tocol" />
```

### Ответ RegisterRefundPaymentQueryResponse

Ответ на запрос регистрации заявки на возврат платежа

Пример ответа:

```
<?xml version="1.0" encoding="utf-8"?>
<Response xmlns:xsi="http://www.w3.org/2001/XMLSchema-
instance" xmlns:xsd="http://www.w3.org/2001/XMLSchema" xsi:type="RegisterRefundPaymentQ
ueryResponse" QueryId="113" ActivationCode="8df60b79-0e88-430d-98be-
74790a932eb2" Amount="10.00" State="0" xmlns="http://ekassir.com/eKassir/PaySystem/Serv
er/eKassirV3Protocol">
  <Payment PaymentState="7" Id="56d8bdd0-c0c0-4ffd-a136-
f4975a3999c7" Ticket="479467456" Number="1636683324760442998" ClientTime="2012-02-
02T16:11:29.36" ServerTime="2012-02-
02T16:11:29.41" EncashmentNumber="143" Value="10.00" Commission="0.00" Amount="10.00" F
romBalance="0.00">
    <PayParameters xmlns="">
      <Parameter Name="account" Value="9993333333" />
    </PayParameters>
    <Service Id="4" Version="49" ExternalId="" Name="Test" TypeId="1" RegionId="0" Logo
="" Alias="mtsall" DefaultOperatorId="1">
      <Parameters xmlns="">
        <Parameter Name="test" Value="&lt;[D/&gt;null" />
        <Parameter Name="TEST33" Value="null" />
      </Parameters>
    </Service>
```

```
<Point Id="1" PointId="7d198045-cf94-4e0b-8764-
d3c3a408f572" Blocked="false" Address="" Name="1-1" CreationDate="2010-09-09" />
<Client Id="1" Name="1" Blocked="false">
  <Requisites LegalName="1" LegalAddress="1" PostAddress="1" TaxpayerIdNumber="1" R
RCode="1" ContactPhoneNumber="1" xmlns="" />
</Client>
<Contract Id="1" Number="11111" Date="2010-09-09" />
<Operator Id="1" Name="1" ContractId="1">
  <Requisites LegalName="Unknown" LegalAddress="Unknown" PostAddress="Unknown" Taxp
ayerIdNumber="000000000000" RRCode="000000" ContactPhoneNumber="" xmlns="" />
</Operator>
<Serial>1810631</Serial>
</Payment>
</Response>
```

## Запрос отмены заявки.

Запрос предназначен для отмены любой заявки.

### Запрос CancelQueryRequest

Запрос отмены заявки

Пример запроса:

```
<?xml version="1.0" encoding="utf-8"?>
<Request xmlns:xsi="http://www.w3.org/2001/XMLSchema-
instance" xmlns:xsd="http://www.w3.org/2001/XMLSchema" xsi:type="CancelQueryRequest" QueryId="114" ActivationCode="2a446ba2-5be4-49d8-814e-bb979235d081" xmlns="http://ekassir.com/eKassir/PaySystem/Server/eKassirV3Protocol" />
```

### Ответ CancelQueryResponse

Ответ на запрос отмены заявки

Пример ответа:

```
<?xml version="1.0" encoding="utf-8"?>
<Response xmlns:xsi="http://www.w3.org/2001/XMLSchema-
instance" xmlns:xsd="http://www.w3.org/2001/XMLSchema" xsi:type="CancelQueryResponse" QueryId="114" Result="true" TextResult="Заявка 114 на возврат платежа успешно отменена." xmlns="http://ekassir.com/eKassir/PaySystem/Server/eKassirV3Protocol" />
```

## Запрос подтверждения заявки.

Запрос предназначен для подтверждения любой заявки.

### Запрос CommitQueryRequest

Запрос подтверждения заявки

Пример запроса:

```
<?xml version="1.0" encoding="utf-8"?>
<Request xmlns:xsi="http://www.w3.org/2001/XMLSchema-
instance" xmlns:xsd="http://www.w3.org/2001/XMLSchema" xsi:type="CommitQueryRequest" QueryId="113" ActivationCode="8df60b79-0e88-430d-98be-74790a932eb2" xmlns="http://ekassir.com/eKassir/PaySystem/Server/eKassirV3Protocol" />
```

### Ответ CommitQueryResponse

Ответ на запрос подтверждения заявки

Пример ответа:

```
<?xml version="1.0" encoding="utf-8"?>
<Response xmlns:xsi="http://www.w3.org/2001/XMLSchema-
instance" xmlns:xsd="http://www.w3.org/2001/XMLSchema" xsi:type="CommitQueryResponse" QueryId="113" Result="true" TextResult="Заявка 113 на возврат платежа успешно обработана. Платеж Serial = 1810631 возвращен." xmlns="http://ekassir.com/eKassir/PaySystem/Server/eKassirV3Protocol" />
```

## Инкассация.

Для возможности учета денежных средств на платежных точках в системе предусмотрена возможность передачи информации по инкассации. В системе используется понятие инкассационного периода. Инкассационный период – временной интервал от одной инкассаций платежной точки до другой, имеющий порядковый номер в рамках платежной точки. При проведении инкассации платежной точки, платежная точка имеет возможность сообщить данные о загруженных и извлеченных наличных средствах при инкассации, а также оборотах денежных средств в рамках инкассируемого периода. Эти данные могут, используются системой для проведения сверок инкассаций и платежей проведенной точкой в данном периоде. Для этого ПО платежной точки должно отправлять запрос регистрации инкассации, указывая номер текущего периода и данные инкассации. После успешного получения ответа, точка может считать, что инкассация зарегистрирована на сервере и открывать новый Инкассационный период. Для возможности сверки платежей в ПО платежной точки любой запрос на регистрацию платежа должен содержать номер текущего инкассационного периода

### Запрос AddEncashmentRequest

Запрос регистрации инкассации

Пример запроса:

```
<?xml version="1.0" encoding="utf-8"?>
<Request xmlns:xsi="http://www.w3.org/2001/XMLSchema-
instance" xmlns:xsd="http://www.w3.org/2001/XMLSchema" xsi:type="AddEncashmentRequest"
xmlns="http://ekassir.com/eKassir/PaySystem/Server/eKassirV3Protocol">
  <Encashment PointId="7d198045-cf94-4e0b-8764-
d3c3a408f572" EncashmentId="141" Collector="collector" InputeDate="2012-01-
26T13:30:44.9355751+04:00" StartDate="2012-01-26T12:27:02.7133911+04:00" EndDate="2012-
01-26T13:30:44.9355751+04:00" State="0" xmlns="">
    <NominalInfoInput>
      <NominalInfo Count="100" NominalValue="10.00" CurrencyId="810" Type="1" />
      <NominalInfo Count="100" NominalValue="50.00" CurrencyId="643" Type="2" />
    </NominalInfoInput>
    <NominalInfoOutput>
      <NominalInfo Count="100" NominalValue="10.00" CurrencyId="810" Type="1" />
      <NominalInfo Count="100" NominalValue="50.00" CurrencyId="643" Type="2" />
    </NominalInfoOutput>
    <CurrencyEncashment>
      <CurrencyEncashmentTotal CurrencyCode="810" TotalCashOutForAllSystem="0" TotalCas
hInForAllSystem="0" TotalCashOutForMainSystem="0" TotalCashInForMainSystem="0" StartTot
alCashOutput="1678800.00" EndTotalCashOutput="1678800.00" StartTotalCashInput="1678800.
00" />
      <CurrencyEncashmentTotal CurrencyCode="643" TotalCashOutForAllSystem="0" TotalCas
hInForAllSystem="0" TotalCashOutForMainSystem="0" TotalCashInForMainSystem="0" StartTot
alCashOutput="3339800.00" EndTotalCashOutput="3339800.00" StartTotalCashInput="3339800.
00" />
    </CurrencyEncashment>
  </Encashment>
</Request>
```

## Ответ AddEncashmentResponse

Ответ на запрос регистрации инкассации

Пример ответа:

```
<?xml version="1.0" encoding="utf-8"?>
<Response xmlns:xsi="http://www.w3.org/2001/XMLSchema-
instance" xmlns:xsd="http://www.w3.org/2001/XMLSchema" xsi:type="AddEncashmentResponse"
  Serial="53286" xmlns="http://ekassir.com/eKassir/PaySystem/Server/eKassirV3Protocol" /
>
```