Департамент информационных технологий города Москвы

Руководство пользователя

электронного сервиса Рсмэв

Сервис регистрации платежей системы проход и питание по УЭК

Дата: 18.11.13

Версия: 1.2

Листов:

АННОТАЦИЯ

Данный документ является руководством пользователя электронного сервиса регистрации платежей системы проход и питание по УЭК.

СОДЕРЖАНИЕ

[1 Общие сведения 5](#_Toc338052092)

[1.1 Руководящие документы 5](#_Toc338052093)

[1.2 Описание электронного сервиса 5](#_Toc338052094)

[1.3 Операции (методы) электронного сервиса 5](#_Toc338052095)

[1.4 Сценарии использования и схема взаимодействия 5](#_Toc338052096)

[1.5 Связи с другими электронными сервисами 5](#_Toc338052097)

[2 Руководство пользователя 6](#_Toc338052098)

[2.1 Операция «Обработать уведомление о произведенном платеже» 6](#_Toc338052099)

[2.1.1 Общие сведения 6](#_Toc338052100)

[2.1.2 Описание входных параметров 6](#_Toc338052101)

[2.1.3 Описание выходных параметров 6](#_Toc338052102)

[2.1.4 Ошибки 7](#_Toc338052103)

[2.1.5 Контрольные примеры 7](#_Toc338052104)

[3 Приложения 9](#_Toc338052105)

[3.1 Описание сервиса (WSDL) 9](#_Toc338052106)

[3.2 Описание формата прикладного блока запроса 14](#_Toc338052107)

изменения

|  |  |  |  |
| --- | --- | --- | --- |
| Версия | Дата | Автор | Изменения |
| 1.0 | 30.07.2012 | Скворцов А.В. | Первичная версия документа |
| 1.1 | 12.10.2012 | Скворцов А.В. | Внесено описание прикладного блока внутри документа |
| 1.2 | 18.11.2013 | Кадыров Д.И. | Расширены коды ошибок |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |

термины/СОКРАЩЕНИя

|  |  |
| --- | --- |
| Термин/сокращение | Описание |
| РСМЭВ | Региональная система межведомственного электронного взаимодействия города Москвы |
| ИС ПП | Информационная система проход и питание по УЭК |
| Поставщик | Организация-владелец электронного сервиса |
| WSDL | Web Services Description Language — язык описания [веб-сервисов](http://ru.wikipedia.org/wiki/%D0%92%D0%B5%D0%B1-%D1%81%D0%B5%D1%80%D0%B2%D0%B8%D1%81). |
| XSD | XML Schema definition - язык описания структуры XML документа. |
| SOAP | Simple Object Access Protocol - протокол обмена структурированными сообщениями в распределённой вычислительной среде. |
| XML | eXtensible Markup Language — расширяемый язык разметки. |

1. Общие сведения

Руководящие документы

Основанием для создания и использования электронного сервиса являются перечисленные ниже документы.

Нормативно-правовые документы:

* Постановление Правительства Российской Федерации от 8 сентября 2010 г. № 697 «О единой системе межведомственного электронного взаимодействия»;
* Приказ Министерства связи и массовых коммуникаций Российской Федерации от 27 декабря 2010 г. № 190 «Об утверждении технических требований к взаимодействию информационных систем в единой системе межведомственного электронного взаимодействия».

Описание электронного сервиса

|  |  |
| --- | --- |
| **Наименование:** | Сервис регистрации платежей системы проход и питание по УЭК |
| **Назначение:** | Получение уведомлений о произведенных платежах для пополнения лицевого счета |
| **Поставщик:** | Департамент информационных технологий г. Москвы |
| **Область применения:** | Другое - Получение уведомлений о факте платежей от кредитных организаций и платежных систем |
| **URL (SOAP)** | https://server:port/processor/soap/payment?wsdl |
| **URL (HTTP)** | https://server:port/processor/payment-std |

Операции (методы) электронного сервиса

|  |  |
| --- | --- |
| **Метод** | **Назначение** |
| process  Получение данных о лицевом счете | Обработать уведомление о произведенном платеже |

Сценарии использования и схема взаимодействия

1. Информационная система вызывает сервис в синхронном режиме используя метод process, передавая данные о платеже

Связи с другими электронными сервисами

Данный сервис не имеет связей с другими сервисами.

1. Руководство пользователя

## Операция «Обработать уведомление о произведенном платеже»

Общие сведения

|  |  |
| --- | --- |
| **Код операции:** | Process |
| **Наименование операции:** | Обработать уведомление о произведенном платеже |
| **Назначение операции:** | Запрос о возможности проведения платежа / регистрация информации о произведенном платеже и зачисление средств на лицевой счет |

Описание входных параметров

**Входные данные: process**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| № | Код параметра | Описание параметра | Обязательность | Способ заполнения/Тип | Комментарий |
| 1 | request | Данные платежа | + | xs:string | В соответствии с разделом 3.2 |

Описание выходных параметров

**Выходные данные: paymentResult**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| № | Код параметра | Описание параметра | Обязательность | Способ заполнения/Тип | Комментарий |
| 1 | response | Результат обработки | + | xs:string | В соответствии с разделом 3.2 |
| 2 | clientId | Номер лицевого счета | + | xs:long |  |
| 3 | opId | Идентификатор операции | + | xs:string |  |
|  | res | Код возврата | + | xs:int |  |
|  | desc | Описание кода возврата | + | xs:string |  |
|  | bal | Баланс после операции |  | xs:long |  |
|  | clientFIO | ФИО клиента |  | xs:string |  |
|  | tspContragentId | Идентификатор поставщика по-умолчанию |  | xs:long |  |
|  | addInfo | Дополнительные параметры поставщика |  | xs:string |  |

Ошибки

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| № | Код возврата | Описание кода возврата | Условия возникновения | Комментарий |
| 1 | 0 | Успешно | Операция завершена (запрос обработан) успешно. |  |
| 2 | 100 | Неопределенная ошибка | Неопределенная ошибка, относящаяся к разряду внутренних. |  |
| 3 | 105 | Клиент не зарегистрирован. | Клиент не зарегистрирован. |  |
| 4 | 106 | Не найден субсчет клиента | Субсчет клиента не активирован |  |
| 5 | 120 | У клиента нет активной карты. | У клиента нет активной карты. |  |
| 6 | 130 | Контрагент не зарегистрирован. | Контрагент не зарегистрирован. |  |
| 7 | 140 | Платеж с указанным идентификатором уже зарегистрирован. | Платеж с указанным идентификатором уже зарегистрирован. |  |
| 8 | 150 | Платеж не может быть совершен для указанного поставщика | Платеж не может быть совершен для указанного поставщика |  |

Контрольные примеры

**Запрос проверки возможности проведения платежа**

|  |
| --- |
| *<?xml version="1.0" encoding="UTF-8"?>*  *<s:Envelope xmlns:s="http://schemas.xmlsoap.org/soap/envelope/">*  *<s:Body xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" xmlns:xsd="http://www.w3.org/2001/XMLSchema">*  *<process xmlns="http://soap.integra.partner.web.processor.ecafe.axetta.ru/">*  *<request xmlns="">* *PID=testpid&amp;CLIENTID=200485 &amp;OPID=100&amp;TERMID=0&amp;V=1</request>*  *</process>*  *</s:Body>*  *</s:Envelope>* |

**Запрос регистрация платежа**

|  |
| --- |
| *<?xml version="1.0" encoding="UTF-8"?>*  *<s:Envelope xmlns:s="http://schemas.xmlsoap.org/soap/envelope/">*  *<s:Body xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" xmlns:xsd="http://www.w3.org/2001/XMLSchema">*  *<process xmlns="http://soap.integra.partner.web.processor.ecafe.axetta.ru/">*  *<request xmlns="">PID=testpid&amp;CLIENTID=200485&amp;OPID=100&amp;TERMID=0&amp;TIME=20120730193209&amp;SUM=1000&amp;SIGNATURE=7FD145FC3B81232B9504999B4AA357F21B8FCAF03F2547566E3BC0DC7362443B218157FF05FECF7B045B2F5486454F0799D8CBB995D8AEB6F6C6A1841318433685459264624923C3386C723AE69575E3AB8112DABF4F8416451A8801B8A3A89B3156B6045B7F2E2178E7C2F9FFDB2C4F4DE9A218BE46271D6A6EA50BC21F10EF</request>*  *</process>*  *</s:Body>*  *</s:Envelope>* |

**Ответ на запрос в случае успешного исполнения**

|  |
| --- |
| *<?xml version="1.0" encoding="UTF-8"?>*  *<soap:Envelope xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/">*  *<soap:Body>*  *<ns2:processResponse xmlns:ns2="http://soap.integra.partner.web.processor.ecafe.axetta.ru/">*  *<return>*  *<response>CLIENTID=200485&amp;OPID=100&amp;RES=0&amp;DESC=Ok&amp;BAL=568389&amp;CLIENTFIO=Петров Петр Иванович&amp;SIGNATURE=5D69150E01E057A196A2AEDD7DF924E2A9019677CB4B5FA45732318C77BC192EFA6047688EE82061783AC6C1CF503F27BAB4371C7A08219A3328EE28F0070B8A307BCA3CD1DFA915F01B37D2A51937B59E0A6750FE13394E0D3163568791B75D8AC2A9F8491B7CC932B5560DF2AF8543BD16D9B780B9D8EC2F63668FBC88604E</response>*  *<clientId>200485</clientId>*  *<opId>100</opId>*  *<res>0</res>*  *<desc>Ok</desc>*  *<bal>568389</bal>*  *<clientFIO>Петров Петр Иванович</clientFIO>*  *<tspContragentId>52</tspContragentId>*  *<addInfo>{TSP.SBID=123123}</addInfo>*  *</return>*  *</ns2:processResponse>*  *</soap:Body>*  *</soap:Envelope>* |

1. Приложения

Описание сервиса (WSDL)

|  |
| --- |
| <?xml version="1.0" encoding="UTF-8"?>  <wsdl:definitions name="PaymentControllerWSService" targetNamespace="http://soap.integra.partner.web.processor.ecafe.axetta.ru/">  <wsdl:types>  <xs:schema attributeFormDefault="unqualified" elementFormDefault="unqualified" targetNamespace="http://soap.integra.partner.web.processor.ecafe.axetta.ru/">  <xs:element name="Data">  <xs:complexType>  <xs:sequence>  <xs:element minOccurs="0" name="ClientSummary" type="tns:ClientSummary"/>  <xs:element minOccurs="0" name="ClientSummaryExt" type="tns:ClientSummaryExt"/>  <xs:element minOccurs="0" name="PurchaseList" type="tns:PurchaseList"/>  <xs:element minOccurs="0" name="PurchaseListExt" type="tns:PurchaseListExt"/>  <xs:element minOccurs="0" name="PaymentList" type="tns:PaymentList"/>  <xs:element minOccurs="0" name="MenuList" type="tns:MenuList"/>  <xs:element minOccurs="0" name="MenuListExt" type="tns:MenuListExt"/>  <xs:element minOccurs="0" name="CardList" type="tns:CardList"/>  <xs:element minOccurs="0" name="EnterEventList" type="tns:EnterEventList"/>  <xs:element minOccurs="0" name="CirculationItemList" type="tns:CirculationItemList"/>  </xs:sequence>  <xs:attribute name="IdOfContract" type="xs:long"/>  <xs:attribute name="ResultCode" type="xs:long"/>  <xs:attribute name="Description" type="xs:string"/>  </xs:complexType>  </xs:element>  <xs:element name="process" type="tns:process"/>  <xs:element name="processResponse" type="tns:processResponse"/>  <xs:complexType name="process">  <xs:sequence>  <xs:element minOccurs="0" name="request" type="xs:string"/>  </xs:sequence>  </xs:complexType>  <xs:complexType name="processResponse">  <xs:sequence>  <xs:element minOccurs="0" name="return" type="tns:paymentResult"/>  </xs:sequence>  </xs:complexType>  <xs:complexType name="paymentResult">  <xs:sequence>  <xs:element minOccurs="0" name="response" type="xs:string"/>  <xs:element name="clientId" type="xs:long"/>  <xs:element minOccurs="0" name="opId" type="xs:string"/>  <xs:element name="res" type="xs:int"/>  <xs:element minOccurs="0" name="desc" type="xs:string"/>  <xs:element minOccurs="0" name="bal" type="xs:long"/>  <xs:element minOccurs="0" name="clientFIO" type="xs:string"/>  <xs:element minOccurs="0" name="tspContragentId" type="xs:long"/>  <xs:element minOccurs="0" name="addInfo" type="xs:string"/>  </xs:sequence>  </xs:complexType>  <xs:complexType name="CardList">  <xs:sequence>  <xs:element maxOccurs="unbounded" minOccurs="0" name="C" type="tns:CardItem"/>  </xs:sequence>  </xs:complexType>  <xs:complexType name="CardItem">  <xs:sequence/>  <xs:attribute name="IdOfCard" type="xs:long"/>  <xs:attribute name="CrystalId" type="xs:long"/>  <xs:attribute name="Type" type="xs:int"/>  <xs:attribute name="State" type="xs:int"/>  <xs:attribute name="LifeState" type="xs:int"/>  <xs:attribute name="ExpiryDate" type="xs:date"/>  <xs:attribute name="ChangeDate" type="xs:dateTime"/>  </xs:complexType>  <xs:complexType name="PurchaseListExt">  <xs:sequence>  <xs:element maxOccurs="unbounded" minOccurs="0" name="P" type="tns:PurchaseExt"/>  </xs:sequence>  </xs:complexType>  <xs:complexType name="PurchaseExt">  <xs:sequence>  <xs:element maxOccurs="unbounded" minOccurs="0" name="E" type="tns:PurchaseElementExt"/>  </xs:sequence>  <xs:attribute name="Time" type="xs:dateTime"/>  <xs:attribute name="Sum" type="xs:long"/>  <xs:attribute name="SocDiscount" type="xs:long"/>  <xs:attribute name="TrdDiscount" type="xs:long"/>  <xs:attribute name="Donation" type="xs:long"/>  <xs:attribute name="ByCash" type="xs:long"/>  <xs:attribute name="ByCard" type="xs:long"/>  <xs:attribute name="IdOfCard" type="xs:long"/>  </xs:complexType>  <xs:complexType name="PurchaseElementExt">  <xs:sequence/>  <xs:attribute name="Name" type="xs:string"/>  <xs:attribute name="Sum" type="xs:long"/>  <xs:attribute name="Amount" type="xs:long"/>  <xs:attribute name="Type" type="xs:int"/>  </xs:complexType>  <xs:complexType name="MenuListExt">  <xs:sequence>  <xs:element maxOccurs="unbounded" minOccurs="0" name="M" type="tns:MenuDateItemExt"/>  </xs:sequence>  </xs:complexType>  <xs:complexType name="MenuDateItemExt">  <xs:sequence>  <xs:element maxOccurs="unbounded" minOccurs="0" name="E" type="tns:MenuItemExt"/>  </xs:sequence>  <xs:attribute name="Date" type="xs:date"/>  </xs:complexType>  <xs:complexType name="MenuItemExt">  <xs:sequence/>  <xs:attribute name="Group" type="xs:string"/>  <xs:attribute name="Name" type="xs:string"/>  <xs:attribute name="Price" type="xs:long"/>  <xs:attribute name="Calories" type="xs:double"/>  <xs:attribute name="VitB1" type="xs:double"/>  <xs:attribute name="VitC" type="xs:double"/>  <xs:attribute name="VitA" type="xs:double"/>  <xs:attribute name="VitE" type="xs:double"/>  <xs:attribute name="MinCa" type="xs:double"/>  <xs:attribute name="MinP" type="xs:double"/>  <xs:attribute name="MinMg" type="xs:double"/>  <xs:attribute name="MinFe" type="xs:double"/>  </xs:complexType>  <xs:complexType name="PurchaseElement">  <xs:sequence/>  <xs:attribute name="Name" type="xs:string"/>  <xs:attribute name="Sum" type="xs:long"/>  <xs:attribute name="Amount" type="xs:long"/>  </xs:complexType>  <xs:complexType name="ClientSummary">  <xs:sequence/>  <xs:attribute name="DateOfContract" type="xs:date"/>  <xs:attribute name="StateOfContract" type="xs:string"/>  <xs:attribute name="Balance" type="xs:long"/>  <xs:attribute name="OverdraftLimit" type="xs:long"/>  </xs:complexType>  <xs:complexType name="ClientSummaryExt">  <xs:sequence/>  <xs:attribute name="ContractId" type="xs:long"/>  <xs:attribute name="DateOfContract" type="xs:date"/>  <xs:attribute name="StateOfContract" type="xs:string"/>  <xs:attribute name="Balance" type="xs:long"/>  <xs:attribute name="OverdraftLimit" type="xs:long"/>  <xs:attribute name="ExpenditureLimit" type="xs:long"/>  <xs:attribute name="FirstName" type="xs:string"/>  <xs:attribute name="LastName" type="xs:string"/>  <xs:attribute name="MiddleName" type="xs:string"/>  <xs:attribute name="Grade" type="xs:string"/>  <xs:attribute name="OfficialName" type="xs:string"/>  <xs:attribute name="LastEnterEventCode" type="xs:int"/>  <xs:attribute name="LastEnterEventTime" type="xs:dateTime"/>  <xs:attribute name="NotifyViaSMS" type="xs:boolean"/>  <xs:attribute name="NotifyViaEmail" type="xs:boolean"/>  <xs:attribute name="MobilePhone" type="xs:string"/>  <xs:attribute name="Email" type="xs:string"/>  <xs:attribute name="DefaultMerchantId" type="xs:long"/>  <xs:attribute name="DefaultMerchantInfo" type="xs:string"/>  </xs:complexType>  <xs:complexType name="PurchaseList">  <xs:sequence>  <xs:element maxOccurs="unbounded" minOccurs="0" name="P" type="tns:Purchase"/>  </xs:sequence>  </xs:complexType>  <xs:complexType name="Purchase">  <xs:sequence>  <xs:element maxOccurs="unbounded" minOccurs="0" name="E" type="tns:PurchaseElement"/>  </xs:sequence>  <xs:attribute name="Time" type="xs:dateTime"/>  <xs:attribute name="Sum" type="xs:long"/>  <xs:attribute name="Discount" type="xs:long"/>  <xs:attribute name="Donation" type="xs:long"/>  <xs:attribute name="ByCash" type="xs:long"/>  <xs:attribute name="ByCard" type="xs:long"/>  <xs:attribute name="IdOfCard" type="xs:long"/>  </xs:complexType>  <xs:complexType name="PaymentList">  <xs:sequence>  <xs:element maxOccurs="unbounded" minOccurs="0" name="P" type="tns:Payment"/>  </xs:sequence>  </xs:complexType>  <xs:complexType name="Payment">  <xs:sequence/>  <xs:attribute name="Time" type="xs:dateTime"/>  <xs:attribute name="Sum" type="xs:long"/>  <xs:attribute name="Origin" type="xs:string"/>  </xs:complexType>  <xs:complexType name="MenuList">  <xs:sequence>  <xs:element maxOccurs="unbounded" minOccurs="0" name="M" type="tns:MenuDateItem"/>  </xs:sequence>  </xs:complexType>  <xs:complexType name="MenuDateItem">  <xs:sequence>  <xs:element maxOccurs="unbounded" minOccurs="0" name="E" type="tns:MenuItem"/>  </xs:sequence>  <xs:attribute name="Date" type="xs:date"/>  </xs:complexType>  <xs:complexType name="MenuItem">  <xs:sequence/>  <xs:attribute name="Group" type="xs:string"/>  <xs:attribute name="Name" type="xs:string"/>  <xs:attribute name="Price" type="xs:long"/>  </xs:complexType>  <xs:complexType name="EnterEventList">  <xs:sequence>  <xs:element maxOccurs="unbounded" minOccurs="0" name="E" type="tns:EnterEventItem"/>  </xs:sequence>  </xs:complexType>  <xs:complexType name="EnterEventItem">  <xs:sequence/>  <xs:attribute name="DateTime" type="xs:dateTime"/>  <xs:attribute name="Day" type="xs:int"/>  <xs:attribute name="EnterName" type="xs:string"/>  <xs:attribute name="Direction" type="xs:int"/>  <xs:attribute name="TemporaryCard" type="xs:int"/>  </xs:complexType>  <xs:complexType name="CirculationItemList">  <xs:sequence>  <xs:element maxOccurs="unbounded" minOccurs="0" name="C" type="tns:CirculationItem"/>  </xs:sequence>  </xs:complexType>  <xs:complexType name="CirculationItem">  <xs:sequence>  <xs:element name="Publication" type="tns:PublicationItem"/>  </xs:sequence>  <xs:attribute name="IssuanceDate" type="xs:date"/>  <xs:attribute name="RefundDate" type="xs:date"/>  <xs:attribute name="RealRefundDate" type="xs:date"/>  <xs:attribute name="Status" type="xs:int"/>  </xs:complexType>  <xs:complexType name="PublicationItem">  <xs:sequence/>  <xs:attribute name="Author" type="xs:string"/>  <xs:attribute name="Title" type="xs:string"/>  <xs:attribute name="Title2" type="xs:string"/>  <xs:attribute name="PublicationDate" type="xs:string"/>  <xs:attribute name="Publisher" type="xs:string"/>  </xs:complexType>  <xs:element name="Exception" type="tns:Exception"/>  <xs:complexType name="Exception">  <xs:sequence>  <xs:element minOccurs="0" name="message" type="xs:string"/>  </xs:sequence>  </xs:complexType>  </xs:schema>  </wsdl:types>  <wsdl:message name="processResponse">  <wsdl:part element="tns:processResponse" name="parameters">  </wsdl:part>  </wsdl:message>  <wsdl:message name="process">  <wsdl:part element="tns:process" name="parameters">  </wsdl:part>  </wsdl:message>  <wsdl:message name="Exception">  <wsdl:part element="tns:Exception" name="Exception">  </wsdl:part>  </wsdl:message>  <wsdl:portType name="PaymentController">  <wsdl:operation name="process">  <wsdl:input message="tns:process" name="process">  </wsdl:input>  <wsdl:output message="tns:processResponse" name="processResponse">  </wsdl:output>  <wsdl:fault message="tns:Exception" name="Exception">  </wsdl:fault>  </wsdl:operation>  </wsdl:portType>  <wsdl:binding name="PaymentControllerWSServiceSoapBinding" type="tns:PaymentController">  <soap:binding style="document" transport="http://schemas.xmlsoap.org/soap/http"/>  <wsdl:operation name="process">  <soap:operation soapAction="" style="document"/>  <wsdl:input name="process">  <soap:body use="literal"/>  </wsdl:input>  <wsdl:output name="processResponse">  <soap:body use="literal"/>  </wsdl:output>  <wsdl:fault name="Exception">  <soap:fault name="Exception" use="literal"/>  </wsdl:fault>  </wsdl:operation>  </wsdl:binding>  <wsdl:service name="PaymentControllerWSService">  <wsdl:port binding="tns:PaymentControllerWSServiceSoapBinding" name="PaymentControllerWSPort">  <soap:address location="http://localhost:8080/processor/soap/payment"/>  </wsdl:port>  </wsdl:service>  </wsdl:definitions> |

Описание формата прикладного блока запроса

Данный раздел описывает правила формирования параметров request и response при работе с сервисом регистрации платежей ИС ПП.

Параметры запроса передаются в стандартной схеме URL-кодирования параметров.

По согласованию используется дополнительная проверка ЭЦП внутри прикладного блока.

Ответ состоит из пар «ПАРАМЕТР=ЗНАЧЕНИЕ», разделенных символом «&». ЭЦП рассчитывается для всего текста запроса (ответа) (блока параметров запроса P1=XXX&P2=XXX…) и добавляется в конце &SIGNATURE=… . Значение подписи кодируется в HEX без префикса (например, &SIGNATURE=A7AED6C8A…).

Используется 2 типа запросов:

1. **Запрос о возможности проведения платежа**

Должен содержать:

* PID: идентификатор платежной системы (текстовый);
* CLIENTID: идентификатор клиента (лица, производящего оплату) он же «номер лицевого счета плательщика» (целое число, 64-битовое знаковое);
* OPID: уникальный номер операции (целое число, 64-битовое знаковое);
* TERMID: код точки приема (целое число, 64-битовое знаковое);
* V: номер версии протокола (0/1);
* SIGNATURE: ЭЦП (SHA1withRSA) для строки запроса до текста &SIGNATURE (в HEX).

Ответ будет содержать:

* CLIENTID: идентификатор клиента (целое число, 64-битовое знаковое);
* OPID: уникальный номер операции (целое число, 64-битовое знаковое);
* RES: поле с целочисленным кодом, указывающим на результат обработки запроса, код 0 – успешно, остальные – ошибка (целое число, 32-битовое знаковое);
* DESC: текстовое описание ошибки, например: «допускается совершение платежа», «не допускается совершение платежа по причине некорректного номера счета», «не допускается совершение платежа по причине занесения номера счета в стоп-лист», «не допускается совершение платежа по причине временной приостановки в обслуживании» и т.п. (текст, максимальная длина 1024).
* BAL: в случае успешного выполнения запроса, состояние баланса – сумма в копейках (целое число, 64-битовое знаковое, опциональное, гарантировано наличие лишь при нулевом коде результата обработки);
* CLIENTFIO: в случае успешного выполнения запроса, Фамилия Имя Отчество клиента, опциональное, гарантировано наличие лишь при нулевом коде результата обработки (формат: «Фамилия Имя Отчество», максимальная длина 322 символа);
* TSPID [версия протокола > 0]: идентификатор поставщика по умолчанию для клиента (целое число, 64-битовое знаковое);
* SIGNATURE: ЭЦП (SHA1withRSA) для ответа (до текста SIGNATURE).

***Пример запроса (поле request):***

PID=paysys&CLIENTID=101519&OPID=99999990&TERMID=114&V=1&SIGNATURE=...

***Примеры ответов (поле response):***

Пример успешного запроса:

CLIENTID=101519&OPID=99999990&RES=0&DESC=Ok&BAL=-3700&CLIENTFIO=Иванов Иван Иванович&TSPID=100&SIGNATURE=…

Пример запроса с ошибкой:

CLIENTID=101519&OPID=99999990RES=140&DESC=Payment is already registered. IdOfContragent == 32, IdOfPayment == 99999990&SIGNATURE=…

1. **Уведомление об успешном завершении платежа**

Уведомление должно содержать:

* PID: идентификатор платежной системы (текстовый);
* CLIENTID: номер лицевого счета плательщика (целое число, 64-битовое знаковое);
* SUM: сумму к зачислению с вычетом комиссии (сумма в копейках, целое число, 64-битовое знаковое);
* TIME: дату и время совершения платежа по часам терминала (дата и время, YYYYMMDDHHMMSS);
* OPID: уникальный номер операции (целое число, 64-битовое знаковое, должен совпадать с аналогичным полем в «запросе о возможности проведения платежа»);
* TERMID: код точки приема (целое число);
* TSPID [версия протокола > 0]: идентификатор поставщика, в адрес которого производится платеж (целое число, 64-битовое знаковое) [ОПЦИОНАЛЬНО, если указано проверяется возможность оплаты для данного лицевого счета на данного поставщика];
* SIGNATURE: ЭЦП (SHA1withRSA) для ответа (до текста SIGNATURE).

Если на запрос не получен однозначный ответ (например, ошибка связи), то необходимо повторять запрос до получения однозначного ответа.

Ответ будет содержать:

* CLIENTID: номер лицевого счета плательщика (целое число, 64-битовое знаковое);
* OPID: уникальный номер операции, указанный в запросе (целое число, 64-битовое знаковое);
* RES: поле с целочисленным кодом, указывающим на результат обработки запроса, код 0 – успешно, остальные – ошибка (целое число, 32-битовое знаковое). Значения приводятся ниже в таблице «Коды ошибок»;
* DESC: сообщение об ошибке (текст, максимальная длина 1024);
* BAL: в случае успешного выполнения запроса, состояние баланса – сумма в копейках (целое число, 64-битовое знаковое, опциональное, гарантировано наличие лишь при нулевом коде результата обработки);
* CLIENTFIO: в случае успешного выполнения запроса, Фамилия Имя Отчество клиента, опциональное, гарантировано наличие лишь при нулевом коде результата обработки (формат: «Фамилия Имя Отчество», максимальная длина 322 символа);
* TSPID [версия протокола > 0]: идентификатор поставщика (целое число, 64-битовое знаковое), в адрес которого зачтен платеж (целое число, 64-битовое знаковое);
* SIGNATURE: ЭЦП (SHA1withRSA) для ответа (до текста SIGNATURE) в HEX.

***Пример запроса:***

PID=paysys&CLIENTID=101519&TSPID=100&OPID=99999990&TERMID=114&SUM=1000&TIME=20101217224500&V=1

***Пример ответа:***

CLIENTID=101519&OPID=99999990&RES=0&DESC=Ok&BAL=-1700&CLIENTFIO=Иванов Иван Иванович&TSPID=100&SIGNATURE=…

**Пример запроса для проверки корректности формирования подписи:**

PID=testpid&amp;CLIENTID=200485&amp;OPID=0&amp;TERMID=0&amp;TIME=20120727175501&amp;SUM=1000&amp;SIGNATURE=5F2E84F9ACE0E46098700CDBADC23994A7F59635D989EAB48554564746CFD2F1262E4F71199CE6824EEB7105CF3CA5631BC44066F7E92B8309F3CD46CFA2B2E84A4335AE6F5B88409A8D606B9C847B6BA35C2C108765F5A7095DF167866B19DA13BA8E1C392ABA530E8D9E0D11E4AD992B172886B77482B2E62A8D319D3E91AC

Закрытый ключ: 

Пароль: 1