

Протокол взаимодействия клиентского ПО и процессинга

Версия 2.0

Содержание

1 ВЗАИМОДЕЙСТВИЕ С ПРОЦЕССИНГОМ	2
1.1 НАЗНАЧЕНИЕ ДОКУМЕНТА	2
1.2 АВТОРИЗАЦИЯ	2
1.3 ПРИНЦИПЫ ВЗАИМОДЕЙСТВИЯ СИСТЕМЫ АГЕНТА И ПРОЦЕССИНГА.....	2
2 СТРУКТУРА ПРОТОКОЛА.....	5
2.1 ЗАПРОС СПИСКА ДОСТУПНЫХ ДЛЯ ПРИЕМА ПЛАТЕЖЕЙ УСЛУГ И ОПИСАНИЯ ПО НИМ	5
2.2 ЗАПРОС ПАРАМЕТРОВ ОПРЕДЕЛЕННОЙ УСЛУГИ.....	6
2.3 ЗАПРОС НА ПРОВЕРКУ БАЛАНСА АГЕНТА	8
2.4 ЗАПРОС НА ПРОВЕРКУ ВОЗМОЖНОСТИ ПРОВЕДЕНИЯ ПЛАТЕЖА	8
2.4.1 <i>Варианты запросов</i>	8
2.4.2 <i>Варианты ответов</i>	10
2.5 ЗАПРОС НА ПРОВЕДЕНИЯ ПЛАТЕЖА	15
2.5.1 <i>Варианты запросов</i>	15
2.5.2 <i>Варианты ответов</i>	17
2.6 ЗАПРОС НА ОПРЕДЕЛЕНИЕ ОПЕРАТОРА СОТОВОЙ СВЯЗИ MNP.....	17

1 ВЗАИМОДЕЙСТВИЕ С ПРОЦЕССИНГОМ

1.1 Назначение документа

Взаимодействие между системой Агента и процессинговым центром осуществляется по протоколу **HTTP** или **HTTPS**. URL-адрес тестового и продуктового процессингового центра предоставляется Агенту отдельным письмом.

Все запросы к процессинговому центру передаются методом **POST**. Для передачи данных в запросе используются формат **JSON** (Content-Type: application/json, Accept: application/json) в кодировке **UTF-8**.

По результатам обработки полученного сообщения процессинг отправляется ответ в том же формате: **JSON** (Content-Type: application/json) в кодировке **UTF-8**.

1.2 Авторизация

Авторизация запросов от системы Агента в процессинге осуществляется с использованием токенов или IP-адресов. Управление параметрами авторизации осуществляется через личный кабинет Агента в разделе «Настройки протокола».

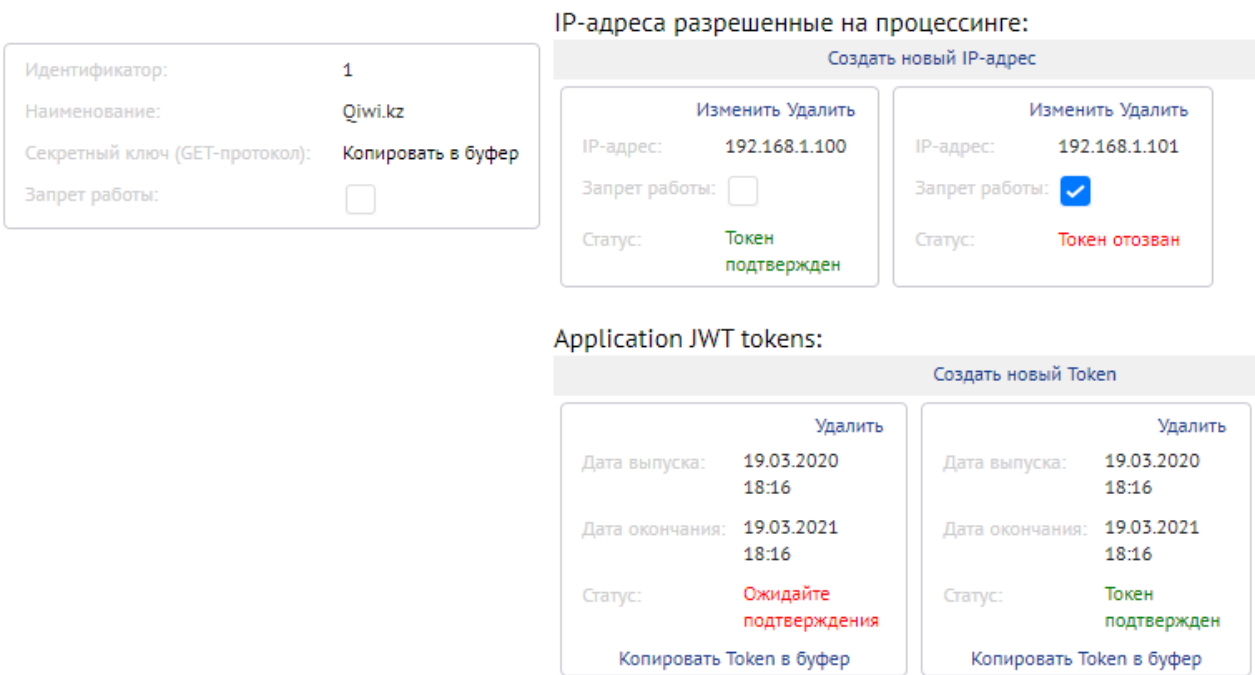


Рисунок 1 – Настройка параметров авторизации

Для создания токена щелкните «Создать новый Token» и подтвердите свое действие. Созданный токен будет автоматически отправлен на сервер процессинга для подтверждения регистрации (для токена будет выставлен статус «Ожидайте подтверждения»). Токены, успешно подтвержденные оператором процессинга, переводятся в статус «Токен подтвержден». Для отзыва действующего токена щелкните «Удалить», токен будет переведен в статус «Токен отозван».

При необходимости, взаимодействие с процессингом может быть ограничено набором IP-адресов, с которых будут приниматься запросы. Для ввода ограничения по IP-адресу необходимо щелкнуть «Создать новый IP-адрес» и указать IP для которого будет введено разрешение на прием запросов процессингом.

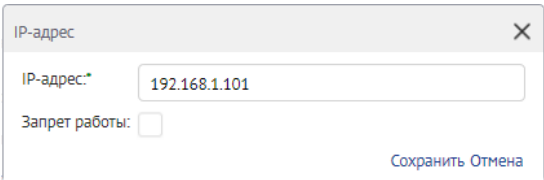


Рисунок 2 – Ввод IP-адреса для ограничения взаимодействия с процессингом

В случае если не указан ни один IP-адрес, то ограничений по ним нет.

1.3 Принципы взаимодействия системы Агента и процессинга

Каждый платеж в системе Агента имеет уникальный идентификатор, который передается в процессинг в переменной **agentTransactionId** – целое число длиной до 18 знаков. По

данному идентификатору производится дальнейшая сверка взаиморасчетов и решение спорных вопросов.

На каждый переданный в процессинг платеж возвращается уникальный номер транзакции проведения платежа в переменной **transactionId** – целое число длиной до 18 знаков. По значению **transactionId** так же можно провести сверку взаиморасчетов и решения спорных вопросов.

При проведении платежа сумма к зачислению передается в переменной **amountTo** – дробное число с точностью до сотых, в качестве разделителя используется «.» (точка). Если сумма представляет целое число, то оно дополняется точкой и нулями, например – «180.00».

Внимание! В запросе check значение в переменной amountTo передается минимальная сумма платежа по сервису, данный параметр в запросе на проверку не обязательный его можно не передавать.

При проведении платежа сумма, принятая от клиента, передается в переменной **amountFrom** – дробное число с точностью до сотых, в качестве разделителя используется «.» (точка). Если сумма представляет целое число, то оно дополняется точкой и нулями, например – «200.00».

Внимание! В запросе check значение в переменной amountFrom передается минимальная сумма платежа по сервису, данный параметр в запросе на проверку не обязательный его можно не передавать.

В запросе на оплату **pay**, Агент передает дату платежа (под датой платежа подразумевается дата получения запроса от клиента) в переменной **agentTransactionDate** – дата в формате YYYY-MM-DDThh:mm:ss. Эту дату необходимо использовать для проведения сверок и бухгалтерских взаиморасчетов. Так как в Платежной организации учет платежей ведется по дате получения запроса от Агента, то и расчеты с Агентом производиться по этой дате.

Идентификатор пользователя (плательщика) (номер лицевого счета, телефона, логин и т.д.) передается в переменной **account** – строка, содержащая буквы, цифры и спецсимволы, длиной до 200 символов. Перед отправкой запроса в процессинг, интерфейс Агента должен произвести проверку корректности в соответствии с регулярным выражением, которое можно получить в ответе на запрос списка доступных сервисов **services**.

При проведении платежей в пользу некоторых сервисов возможна передача дополнительных полей (дополнительные данные, введенные пользователем (плательщиком), номер терминала, тип платежа и т.п.) которые передаются в процессинг как экстра-поля в объекте **extras**, при платеже может быть передано одно или несколько экстра-полей в зависимости от требований провайдера услуг. Например, при оплате заказа пользователь вводит свой логин и номер заказа, в таком случае логин передается в параметре **account**, а номер заказа в экстра-поле **account1** или **order**. Список параметров необходимых при проведении платежа можно получить путем отправки запроса на получение списка доступных сервисов **services**.

Проведение платежа производится в 2 этапа – проверка возможности проведения платежа **check**, а также проверка идентификатора клиента и непосредственно проведение платежа **pay**. Тип запроса передается как часть пути до сервиса.

Если у Агента имеется техническая возможность идентифицировать плательщиков - то идентификационные данные плательщика указываются в параметре **identityCard** и **fullName**.

Результат выполнения запроса возвращается процессингом в виде кодов ответов в параметре **result**. Все коды ответов кроме 0 имеют признак финальности «+/-». Финальный код ответа (тот что отмечен символом «+») означает, что повторная отправка запроса с теми же параметрами, приведет к 100% возврату такого же кода ответа – следовательно, при получении финального кода ответа сервис Агента прекращает отправку запроса и присваивает платежу статус в зависимости от кода ответа. Нефинальный (временный) код ответа означает, что сервис Агента должен повторять запросы, постоянно увеличивая интервал, пока не получит код ответа 0 или финальный код ответа. Отсутствие связи с процессингом является нефинальной ошибкой.

Таблица 1 «Список кодов ответа на запросы»

Код ответа	Описание	Признак финальности
0	успех	да
1	запрос в обработке, повторите запрос позже	нет
4	неверный формат идентификатора абонента	да
5	идентификатор абонента не найден (ошиблись номером)	да
7	прием платежа запрещен провайдером	да
8	прием платежа запрещен по техническим причинам	да
130	работа с данному сервису не реализована	да
153	платеж на проверке в сб	нет
155	прием платежа по данному сервису запрещен	да
202*	ошибка данных запроса	да
215	транзакция с таким номером уже есть в базе	да
220	недостаток средств у агента	нет
250	не был пополнен баланс агента	да
241	сумма слишком мала	да
242	сумма слишком велика	да
255	недопустимая дополнительная комиссия	нет
300	другая ошибка	да

***Код ответа 202** имеет фатальный статус только в случае первичного запроса к процессингу. В случае если Рау запрос находится во временном статусе (status 1, 153, 220,255) и при повторе транзакции было изменено тело запроса на недопустимые параметры – процессинг вернет ошибку 202. Но это не остановит Рау запрос в обработке. Платеж также продолжит обработку в процессинге и примет финальный успешный или ошибочный статус (0 – ОК / 300 – другая ошибка.

В процессинге не должно содержаться двух успешно проведенных платежей от одного Агента с одним и тем же номером транзакции **agentTransactionId**. Если сервис Агента повторно присылает запрос с уже существующим в процессинге **agentTransactionId**, то процессинг возвращает результат обработки предыдущего запроса.

Идентификатор сервиса передается в параметре **serviceId**, список доступных для агента сервисов, их описание, входные параметры и возвращаемые значения можно получить путем отправки запроса **services**

В процессинге все провайдеры условно разделены на три схемы проведения платежа – «простая», «с выбором» и «комплексная».

1.3.1.1 В «простую» схему оплаты входят услуги, при оплате которых пользователь вводит один или несколько параметров. Например, оплата мобильной связи по номеру телефона, покупка игровой валюты по номеру телефона и email адреса и т.п. В случае оплаты сервисов по которым поставщик возвращает данные которые необходимо отобразить пользователю (ФИО, сумма к оплате, задолженность и т.п.) в ответе на запрос проверки такие данные приходят в объекте **displays** который содержит в себе параметры, значение которых можно вывести пользователю.

1.3.1.2 В схему «с выбором» входят услуги по которым на запрос проверки в объекте **contracts** приходят параметры, из которых пользователь выбирает один и он в дальнейшем участвует при оплате. Например, пополнение банковского счета по ИИН т.е. когда пользователь вводит ИИН и на запрос проверки в ответе возвращаются несколько счетов из которых пользователь выбирает один и производит пополнение выбранного счета.

1.3.1.3 В «комплексную» схему входят услуги, при оплате которых за одну транзакцию оплачиваются квитанция с несколькими услугами, в основном такая схема применяется к услугам ЖКХ. Например, пользователь по услуге ЖКХ вводит лицевой счет, в ответе возвращается квитанция с услугами (электроэнергия, водоснабжение, отопление и т.д.), пользователь выбирает необходимые услуги из квитанции вводит по ним суммы и оплачивает одним платежом. Квитанция, которую необходимо вывести в понятном виде для пользователя приходит в объекте **invoice**.

2 СТРУКТУРА ПРОТОКОЛА

2.1 Запрос списка доступных для приема платежей услуг и описания по ним

Структура запроса

Метод	POST
URI	<адрес процессинга>/api/v20/ services
URI Parameters	без параметров
Body Parameters	без параметров
Content-Type	application/json

Структура ответа

параметр	описание	обязательный
services	объект, содержащий информацию по услугам	да
serviceId	код услуги в процессинге	да
type	тип схемы по которой работает услуга (см. Таблица 2)	да
name	наименование услуги	да
group	группа к которой относится услуга	нет
fixedPayment	фиксированная сумма платежа (true – для услуг при оплате которых необходима фиксированная сумма платежа например оплата заказа или брони)	нет
country	страна в которой предоставляется сервис	нет
inputs	объект, содержащий информацию по входящим параметрам которые должен ввести пользователь	нет
name	название параметра	
required	признак обязательности параметра	
title	подсказка для пользователя с краткими указаниями того, что необходимо сделать	
regexp	регулярное выражение для проверки вводимых данных	
displays	объект, содержащий информацию по данным для отображения пользователю	нет
name	название параметра	
required	признак обязательности параметра	

Пример ответа

application/json, text/json
<pre>{ "result": 0, "resultMessage": "ОК", "services": [{ "serviceId": "P1011", "type": 0, "name": "Билайн Казахстан", "group": "Сотовая связь", "fixedPayment": true, "country": "КАЗАХСТАН", "inputs": [{ "name": "account", "required": true, "title": "Номер телефона", "regexp": "^\\d{10}\$" }] }], { "serviceId": "P1012", "type": 0, "name": "Herbalife", "group": "Товары по каталогам", "fixedPayment": true, "country": "КАЗАХСТАН", "inputs": [{ "name": "account",</pre>

```
        "required": true,
        "title": "Номер независимого партнера",
        "regex": "^\\d{10}$"
    }, {
        "name": "ev_account1",
        "required": true,
        "title": "Номер заказа",
        "regex": "^\\d{10}$"
    }
],
"displays": [
    {
        "name": "fio",
        "required": true,
        "title": "ФИО",
        "regex": "^.+$"
    }, {
        "name": "info",
        "required": true,
        "title": "Сумма к оплате",
        "regex": "^.+$"
    }, {
        "name": "fixedAmount",
        "required": true,
        "title": "",
        "regex": "^\\d{1,9}\\.\\d{2}$"
    }
]
}, {
    "serviceId": "P0005",
    "type": 1,
    "name": "Алсеко",
    "group": "ЖКХ",
    "fixedPayment": false,
    "country": "КАЗАХСТАН",
    "inputs": [
        {
            "name": "account",
            "required": true,
            "title": "Номер лицевого счета",
            "regex": "^\\d{10}$"
        }
    ]
}, {
    "serviceId": "P0006",
    "type": 2,
    "name": "XXX банк",
    "group": "Другие услуги",
    "fixedPayment": false,
    "country": "КАЗАХСТАН",
    "inputs": [
        {
            "name": "account",
            "required": true,
            "title": "Введите ИИН",
            "regex": "^\\d{12}$"
        }
    ]
}
]
```

2.2 Запрос параметров определенной услуги

Структура запроса

Описание протокола WEB API version 2.0

Метод	POST
URI	<адрес процессинга>/api/v20/ service
URI Parameters	без параметров
Body Parameters	параметры запроса
Content-Type	application/json

Параметры запроса

параметр	описание	обязательный
serviceId	код услуги в процессинге	да

Пример запроса

application/json, text/json
<pre>{ "serviceId": "P1012" }</pre>

Структура ответа

параметр	описание	обязательный
services	объект, содержащий информацию по услугам	да
serviceId	код услуги в процессинге	да
type	тип схемы по которой работает услуга (см. Таблица 2)	да
name	наименование услуги	да
group	группа к которой относится услуга	нет
fixedPayment	фиксированная сумма платежа (true – для услуг при оплате которых необходима фиксированная сумма платежа например оплата заказа или брони)	нет
country	страна в которой предоставляется сервис	нет
inputs	объект, содержащий информацию по входящим параметрам которые должен ввести пользователь	нет
name	название параметра	
required	признак обязательности параметра	
title	подсказка для пользователя с краткими указаниями того, что необходимо сделать	
regex	регулярное выражение для проверки вводимых данных	
displays	объект, содержащий информацию по данным для отображения пользователю	нет
name	название параметра	
required	признак обязательности параметра	

Пример ответа

application/json, text/json
<pre>{ "result": 0, "resultMessage": "OK", "services": [{ "serviceId": "P1012", "type": 0, "name": "Herbalife", "group": "Товары по каталогам", "fixedPayment": true, "country": "КАЗАХСТАН", "inputs": [{ "name": "account", "required": true, "title": "Номер независимого партнера", "regex": "^\\d{10}\$" }, { "name": "ev_account1", "required": true, "title": "Номер заказа", "regex": "^\\d{10}\$" }] }], }</pre>

Описание протокола WEB API version 2.0

```

    "displays": [
      {
        "name": "fio",
        "required": true,
        "title": "ФИО",
        "regexp": "^.+ $"
      }, {
        "name": "info",
        "required": true,
        "title": "Сумма к оплате",
        "regexp": "^.+ $"
      }, {
        "name": "fixedAmount",
        "required": true,
        "title": "",
        "regexp": "^[\\d{1,9}\\.\\d{2}] $"
      }
    ]
  }
}

```

2.3 Запрос на проверку баланса Агента

Структура запроса

Метод	POST
URI	<адрес процессинга>/api/v20/ balance
URI Parameters	без параметров
Body Parameters	без параметров
Content-Type	application/json

Структура ответа

параметр	описание	обязательный
agentId	идентификатор Агента	да
agentName	наименование Агента	да
balance	баланс Агента	да
overdraft	сумма баланса, при достижении которого прием платежей будет отключен автоматически	да
notification	сумма баланса, при достижении которого будет отправлено уведомление сотруднику Агента	нет
currency	валюта договора	нет

Пример ответа

```

application/json, text/json
{
  "result": 0,
  "resultMessage": "Успешно",
  "agent": {
    "agentId": 4,
    "agentName": "Наименоване Агента",
    "balance": 509455.9000,
    "overdraft": 0.0000,
    "notification": 0.0000,
    "currency": "KZT"
  }
}

```

2.4 Запрос на проверку возможности проведения платежа

2.4.1 Варианты запросов

Запрос с проверкой номера

Структура запроса

Метод	POST
URI	<адрес процессинга>/api/v20/ check

Описание протокола WEB API version 2.0

URI Parameters	без параметров
Body Parameters	параметры запроса
Content-Type	application/json

Параметры запроса

параметр	описание	обязательный
serviceld	код услуги в процессинге	да
account	идентификатор пользователя (номер телефона, email и т.п.)	да
agentTransactionId	идентификатор операции на стороне Агента	да
agentTransactionDate	дата операции на стороне Агента	да
amountTo	сумма платежа. Сумма к зачислению	нет
amountFrom	сумма принятая от пользователя	нет
ПРИМЕЧАНИЕ	В запросе check значение в параметрах amountTo и amountFrom передается минимальная сумма платежа по сервису, данные параметр в запросе на проверку не обязательны их можно не передавать.	

Пример запроса

application/json, text/json
<pre>{ "serviceld": "P1011", "account": "7770017711", "agentTransactionId": 1233355, "agentTransactionDate": "2015-02-17T16:48:37", "amountTo": 200.00, "amountFrom": 200.00 }</pre>

Запрос с идентификационными данными пользователя

Структура запроса

Метод	POST
URI	<адрес процессинга>/api/v20/check
URI Parameters	без параметров
Body Parameters	параметры запроса
Content-Type	application/json

Параметры запроса

параметр	описание	обязательный
serviceld	код услуги в процессинге	да
account	идентификатор пользователя (номер телефона, email и т.п.)	да
agentTransactionId	идентификатор операции на стороне Агента	да
agentTransactionDate	дата операции на стороне Агента	да
amountTo	сумма платежа. Сумма к зачислению	нет
amountFrom	сумма принятая от пользователя	Нет
identityCard	номер паспорта или удостоверения личности пользователя	нет
fullName	ФИО пользователя	нет
ПРИМЕЧАНИЕ	Согласно закону о платежах и платежных системах законодательством некоторых стран при оплате услуг пользователь должен передавать свои идентификационные данные такие как номер паспорта и ФИО.	

Пример запроса

application/json, text/json
<pre>{ "serviceld": "P1011", "account": "7770017711", "agentTransactionId": 1233355, "agentTransactionDate": "2015-02-17T16:48:37", "amountTo": 200.00, "amountFrom": 200.00, "identityCard": "123456789", "fullName": "Иванов Иван Иванович" }</pre>

Описание протокола WEB API version 2.0

Запрос с проверкой номера и экстра-полями

Структура запроса

Метод	POST
URI	<адрес процессинга>/api/v20/ check
URI Parameters	без параметров
Body Parameters	параметры запроса
Content-Type	application/json

Параметры запроса

параметр	описание	обязательный
servicId	код услуги в процессинге	да
account	идентификатор пользователя (номер телефона, email и т.п.)	да
agentTransactionId	идентификатор операции на стороне Агента	да
agentTransactionDate	дата операции на стороне Агента	да
amountTo	сумма платежа. Сумма к зачислению	нет
amountFrom	сумма принятая от пользователя	нет
extras	тег-контейнер для экстра-полей	да
account1	экстра поле account1	нет
account2	экстра поле account2	нет
ПРИМЕЧАНИЕ	При check запросе по некоторым услугам необходима передача дополнительных полей это могут быть дополнительные данные введенные пользователем (плательщиком), номер терминала, тип платежа и т.п. которые передаются в процессинг как экстра-поля в объекте extras . Например, при оплате услуги, пользователь помимо основного идентификатора должен ввести дополнительные данные такие как email и номер заказа.	

Пример запроса

application/json, text/json
<pre>{ "servicId": "P1012", "account": "0", "agentTransactionId": 1233355, "agentTransactionDate": "2015-02-17T16:48:37", "amountTo": 200.00, "amountFrom": 200.00, "extras": { "account1": "test@gmail.com", "account2": "121231" } }</pre>

2.4.2 Варианты ответов

Ответ с результатом проверки платежа и проверки номера

Структура ответа

параметр	описание	обязательный
result	код ответа на запрос	да
resultMessage	описание ошибки если result !=0	да
transactionId	номер транзакции в процессинге	да
agentTransactionId	номер транзакции Агента	да

Пример ответа

application/json, text/json
<pre>{ "result": 0, "resultMessage": "ok", "transactionId": 100000781233355, "agentTransactionId": 1233355 }</pre>

Ответ с результатом проверки платежа и проверки номера по международным провайдерам

Структура ответа

параметр	описание	обязательный
result	код ответа на запрос	да

Описание протокола WEB API version 2.0

resultMessage	описание ошибки если result !=0	да
transactionId	номер транзакции в процессинге	да
agentTransactionId	номер транзакции Агента	да
currencyRate	сообщение о курсе конвертации для пользователя (если result = 0)	нет
rate	курс конвертации (если result = 0)	нет
finalAmount	сумма которая поступит пользователю в валюте провайдера (если result = 0)	нет
currency	валюта провайдера (если result = 0)	нет

Пример ответа

application/json, text/json
<pre>{ "result": 0, "resultMessage": "Успешно", "transactionId": 1402546, "agentTransactionId": 12, "currencyRate": "Курс конвертации KZT/USD 0.0021", "rate": 0.0020833333, "finalAmount": 1.75, "currency": "USD" }</pre>

Ответ с данными для вывода пользователю

Структура ответа

параметр	описание	обязательный
result	код ответа на запрос	да
resultMessage	описание ошибки если result !=0	да
transactionId	номер транзакции в процессинге	да
agentTransactionId	номер транзакции Агента	да
displays	объект/тег-контейнер для перечня выводимых пользователю данных	нет
ПРИМЕЧАНИЕ	В объекте displays может вернуться от 1 до 30 параметров значение которых необходимо вывести пользователю или использовать при платеже, посмотреть какие параметры возвращаются по услуге можно с помощью запроса на получение списка доступных услуг (services)	

Пример ответа

application/json, text/json
<pre>{ "result": 0, "resultMessage": "ok", "transactionId": 270, "agentTransactionId": 205, "displays": { "fio": "Test fio", "address": "Набережная 134 кв 5", "info": "Сумма к оплате 20000.00 тенге" } }</pre>

Ответ для схемы с «выбором»

Структура ответа

параметр	описание	обязательный
result	код ответа на запрос	да
resultMessage	описание ошибки если result !=0	да
transactionId	номер транзакции в процессинге	да
agentTransactionId	номер транзакции Агента	да
contracts	тег-контейнер для перечня выбираемых для оплаты или пополнения счетов, заказов, договоров и т.п.	да
contractId	идентификатор счета, заказа, договора и т.п	да
name	наименование счета, заказа, договора и т.п или фио	нет
contractNumber	Номер контракта	нет

Описание протокола WEB API version 2.0

contractSum	сумма к оплате по счету, заказу, договора и т.п	нет
contractDate	даты выставления счета, заказа, договора и т.п	нет

ПРИМЕЧАНИЕ	Схема «с выбором» применяется в основном по провайдерам, при оплате которых пользователь дополнительно выбирает счет, заказ или номер договора возвращаемые на запрос проверки. Например, пользователь на сайте провайдера совершил 2 заказа, основным идентификатором пользователя на стороне провайдера выступает номер телефона, пользователь в интерфейсе Агента вводит номер телефона и в ответе от провайдера возвращается 2 заказа из которых пользователь выбирает необходимый и производит оплату, такая же схема работы при пополнении банковского счета, идентификатором пользователя на стороне банка является ИИН, пользователь в интерфейсе Агента вводит ИИН и в ответе от банка возвращаются счета (депозитный, кредитный, текущий, карточный) из которых пользователь выбирает необходимый и производит пополнение.
------------	--

Пример ответа

application/json, text/json
<pre>{ "result": 0, "resultMessage": "ok", "transactionId": 100000781233355, "agentTransactionId": 1233355, "contracts": [{ "name": "Петров", "contractId": "000000587", "contractNumber": "1/00-Ф", "contractSum": "1000.00", "contractDate": "20200215000000" }, { "name": "Петров", "contractId": "000000588", "contractNumber": "2/00-Ф", "contractSum": "2000.00", "contractDate": "20200216000000" }, { "name": "Петров", "contractId": "000000589", "contractNumber": "3/00-Ф", "contractSum": "3000.00", "contractDate": "20200217000000" }] }</pre>

Ответ для «комплексной схемы»

Структура ответа

параметр	описание	обязательный
result	код ответа на запрос	да
resultMessage	описание ошибки если result !=0	да
transactionId	номер транзакции в процессинге	да
agentTransactionId	номер транзакции Агента	да
invoice	тег-контейнер для перечня инвойсов	да
invoices	тег-контейнер для перечня инвойсов	да
invoiceId	идентификатор инвойса	нет
formedDate	дата формирования квитанции	нет
expireDate	дата до которой необходимо оплатить квитанцию	нет
clientName	фио плательщика	нет
clientAddress	адрес плательщика	нет
services	тег-контейнер для оплачиваемых позиций по квитанции	да
service	тег-контейнер для перечня данных услуги	да

Описание протокола WEB API version 2.0

subServiceId	идентификатор услуги в квитанции	да
subServiceName	наименование услуги в квитанции	да
data	тег-контейнер для перечня параметров услуги	нет
paySum	сумма к оплате	нет
tariff	тариф	нет
minSum	минимальная сумма платежа по услуге в квитанции (0- ограничений нет)	нет
maxSum	максимальная сумма платежа по услуге в квитанции (0- ограничений нет)	нет
debtInfo	информация о задолженности	нет
isMeter	услуга оплачиваемая по счетчику (true – да, false – нет)	нет
prevCount	предыдущие показания счетчика	нет
lastCount	текущие показания счетчика	нет
si	ед. изменения	нет
comment	комментарий к услуге	нет
ПРИМЕЧАНИЕ	Электронная квитанция может не содержать некоторых параметров, наличие параметров в квитанции зависит от ответа провайдера на запрос проверки лицевого счета и получения квитанции.	

Пример ответа

application/json, text/json
<pre>{ "result": 0, "resultMessage": "ok", "transactionId": 284, "agentTransactionId": 205, "invoice": { "invoices": [{ "invoiceId": "2020020003535444", "formedDate": "20200201000000", "expireDate": "20200229000000", "clientName": "Иванов Иван Иванович", "clientAddress": "микрорайон 2, дом 42, кв.4", "services": [{ "subServiceId": "36", "subServiceName": "Электроэнергия", "data": { "paySum": 359.8, "tariff": 0.0, "minSum": 0.0, "maxSum": 0.0, "debtInfo": "0.00", "isMeter": true, "prevCount": "4870.00", "lastCount": "0.00", "si": "тг/м2", "comment": "" } }], }, { "subServiceId": "13", "subServiceName": "ТО газа ", "data": { "paySum": 80.0, "tariff": 0.0, "minSum": 0.0, "maxSum": 0.0, "debtInfo": "0.00", "isMeter": false, "prevCount": "0.00", "lastCount": "0.00", "si": "тг/плита", "comment": "" } }], }, }</pre>

```
"subServiceId": "2",
"subServiceName": "Отопление",
"data": {
  "paySum": 925.2,
  "tariff": 500.0,
  "minSum": 0.0,
  "maxSum": 0.0,
  "debtInfo": "0.00",
  "isMeter": false,
  "prevCount": "0.00",
  "lastCount": "0.00",
  "si": "ТТ/М2",
  "comment": ""
}
}
]
}
]
}
}
```

2.5 Запрос на проведения платежа

2.5.1 Варианты запросов

Запрос на проведения платежа

Структура запроса

Метод	POST
URI	<адрес процессинга>/api/v20/ pay
URI Parameters	без параметров
Body Parameters	параметры запроса
Content-Type	application/json

Параметры запроса

параметр	описание	обязательный
serviceld	код услуги в процессинге	да
account	идентификатор пользователя (номер телефона, email и т.п.)	да
agentTransactionId	идентификатор операции на стороне Агента	да
agentTransactionDate	дата операции на стороне Агента	да
amountTo	сумма платежа. Сумма к зачислению	да
amountFrom	сумма принятая от пользователя	нет
ПРИМЕЧАНИЕ	В запросе pay сумма платежа amountTo и сумма принятая от пользователя amountFrom - дробное число с точностью до сотых, в качестве разделителя используется «.» (точка). Если сумма представляет целое число, то оно дополняется точкой и нулями, например – «200.00».	

Пример запроса

```
{
  "servicId": "P1011",
  "account": "7770017711",
  "agentTransactionId": 1233355,
  "agentTransactionDate": "2015-02-17T16:48:37",
  "amountTo": 200.00,
  "amountFrom": 200.00
}
```

Запрос на проведения платежа с экстра-полями

Структура запроса

Метод	POST
URI	<адрес процессинга>/api/v20/ pay
URI Parameters	без параметров
Body Parameters	параметры запроса
Content-Type	application/json

Параметры запроса

параметр	описание	обязательный
serviceld	код услуги в процессинге	да
account	идентификатор пользователя (номер телефона, email и т.п.)	да
agentTransactionId	идентификатор операции на стороне Агента	да
agentTransactionDate	дата операции на стороне Агента	да
amountTo	сумма платежа. Сумма к зачислению	да
amountFrom	сумма принятая от пользователя	да
extras	тег-контейнер для экстра-полей	нет
account1	экстра поле account1	нет
account2	экстра поле account2	нет

Пример запроса

```
application/json, text/json

{
  "servicId": "P1011",
  "account": "7770017711",
  "agentTransactionId": 1233355,
  "agentTransactionDate": "2015-02-17T16:48:37",
  "amountTo": 200.00,
  "amountFrom": 200.00,
```

Описание протокола WEB API version 2.0

```
"extras": {
  "account1": "test@gmail.com",
  "account2": "121231"
}
```

Запрос на проведение платежа по услуге со схемой работы «с выбором»
Структура запроса

Метод	POST
URI	<адрес процессинга>/api/v20/ pay
URI Parameters	без параметров
Body Parameters	параметры запроса
Content-Type	application/json

Параметры запроса

параметр	описание	обязательный
serviceId	код услуги в процессинге	да
account	идентификатор пользователя (номер телефона, email и т.п.)	да
agentTransactionId	идентификатор операции на стороне Агента	да
agentTransactionDate	дата операции на стороне Агента	да
amountTo	сумма платежа. Сумма к зачислению	да
amountFrom	сумма принятая от пользователя	нет
extras	тег-контейнер для экстра-полей	да
contractId	идентификатор договора который пользователь выбрал из ответа на запрос проверки (check)	да
ПРИМЕЧАНИЕ	Проведение платежа по услуге которая работает по «простой» схеме и схеме «с выбором» не отличается так как дополнительные данные введенные или выбранные пользователем передаются в объекте extras .	

Пример запроса

```
application/json, text/json

{
  "serviceId": "P1011",
  "account": "7770017711",
  "agentTransactionId": 1233355,
  "agentTransactionDate": "2015-02-17T16:48:37",
  "amountTo": 200.00,
  "amountFrom": 200.00,
  "extras": {
    "contractId": "000000588"
  }
}
```

Запрос на проведения платежа по услуге со схемой работы «комплексная»
Структура запроса

Метод	POST
URI	<адрес процессинга>/api/v20/ pay
URI Parameters	без параметров
Body Parameters	параметры запроса
Content-Type	application/json

Параметры запроса

параметр	описание	обязательный
serviceId	код услуги в процессинге	да
account	идентификатор пользователя (номер телефона, email и т.п.)	да
agentTransactionId	идентификатор операции на стороне Агента	да
agentTransactionDate	дата операции на стороне Агента	да
amountTo	сумма платежа. Сумма к зачислению	да
amountFrom	сумма принятая от пользователя	нет
extraServices	тег-контейнер для перечисления оплачиваемых из квитанции услуг	да
subServiceId	идентификатор услуги из квитанции	да
amount	сумма платежа по услуге	да
lastCount	последний показания счетчика передается если услуга счетчиковая	нет

ПРИМЕЧАНИЕ	При проведении платежей по услугам которые работают по «комплексной» схеме оплачиваемые услуги из квитанции передаются в объекте extraServices , при этом общая сумма платежа по всем оплачиваемым услугам из квитанции должна быть равна сумме платежа т.е. amountTo
------------	---

Пример запроса

application/json, text/json
<pre>{ "serviceId": "P1011", "account": "7770017711", "agentTransactionId": 1233355, "agentTransactionDate": "2015-02-17T16:48:37", "amountTo": 450.00, "amountFrom": 450.00, "extraServices": [{ "subServiceId": "114", "lastCount": "11236", "amount": 100.00 }, { "subServiceId": "37", "amount": 150.00 }, { "subServiceId": "45", "amount": 200.00 }] }</pre>

2.5.2 Варианты ответов

Варианты ответов аналогичны ответам на проверку возможности проведения платежа с учетом схемы работы по услуге (см. п. 2.4.2).

Примечание! В случае успешного проведения платежа в ответе возвращаются параметры transactionDate – дата и время платежа в процессинге и statusDate – дата и время, когда платеж принял окончательный статус.

Пример ответа

application/json, text/json
<pre>{ "result": 0, "resultMessage": "Успешно", "transactionId": 1402779, "transactionDate": "2020-10-19T20:19:23.737", "statusDate": "2020-10-19T20:19:24.940", "agentTransactionId": 26801048196008 }</pre>

2.6 Запрос на определение оператора сотовой связи MNP

Структура запроса

Метод	POST
URI	<адрес процессинга>/api/v20/operatorByPhone
URI Parameters	без параметров
Body Parameters	параметры запроса
Content-Type	application/json

Параметры запроса

параметр	описание	обязательный
phoneNumber	номер телефона для проверки	да

Пример запроса

application/json, text/json
<pre>{</pre>

Описание протокола WEB API version 2.0

```
{
  "phoneNumber": "7777731414"
}
```

Структура ответа

параметр	описание	обязательный
result	код ответа на запрос	да
resultMessage	описание ошибки если result !=0	да
serviceId	код услуги в процессинге (возвращается только если оператор был определен)	нет

Пример ответа

```
application/json, text/json
{
  "result": 0,
  "resultMessage": "ok",
  "serviceId": "P1011"
}
```