

# **ИНТЕРФЕЙС MMPS ПОДКЛЮЧЕНИЯ ПОСТАВЩИКОВ УСЛУГ**

## ОГЛАВЛЕНИЕ

---

### 2 ТРЕБОВАНИЯ К ИНТЕРФЕЙСУ ПОСТАВЩИКА УСЛУГ2

### 3 ОСНОВНЫЕ ПРИНЦИПЫ РАБОТЫ ИНТЕРФЕЙСА2

### 4 ОПИСАНИЕ МЕТОДОВ ПРОТОКОЛА4

1. `check`:4
2. `pay`:6
3. `GetPaymentsSummary`:7
4. `GetPayments`:9
5. `cancel`:10

### 5 Приложение 1. Список кодов завершения11

## 2 ТРЕБОВАНИЯ К ИНТЕРФЕЙСУ ПОСТАВЩИКА УСЛУГ

---

1. Интерфейс должен принимать запросы либо посредством веб сервиса, либо по протоколу HTTPS со следующих IP адресов:  
[185.212.10.190](#); [217.12.118.190](#)
2. Интерфейс должен обрабатывать параметры, передаваемые системой методом [GET](#).
3. Интерфейс должен формировать ответ системе [в формате XML в кодировке UTF-8](#) (если ответ содержит символы национальных алфавитов)
4. Обмен информацией ведется в режиме запрос-ответ, при этом скорость ответа не должна превышать 60 секунд, в противном случае система разрывает соединение по таймауту.
5. Если предполагаемое количество платежей за услуги подключаемого поставщика услуг ожидается достаточно интенсивным (до 10 платежей в минуту и более), желательно, чтобы интерфейс спокойно переносил многопоточную коммуникацию до 10-15 одновременных соединений.

## 3 ОСНОВНЫЕ ПРИНЦИПЫ РАБОТЫ ИНТЕРФЕЙСА

---

1. Каждый платеж в системе MMPS имеет уникальный идентификатор, который передается поставщику услуг в переменной [txn\\_Id](#), представляющей собой строку из шестнадцатеричных цифр (0-F) длиной в 32 знака. По этому идентификатору производится дальнейшая сверка взаиморасчетов и решение спорных вопросов.

2. Сумма платежа принимается от абонента и передается поставщику услуг в леях в переменной **sum** в виде дробного числа с точностью до сотых, в качестве разделителя используется «.» (точка). Если сумма представляет целое число, то оно все равно дополняется точкой и нулями, например – «152.00»
3. В запросе на добавление платежа система передает дату платежа (под датой платежа в системе подразумевается дата получения запроса от клиента) в переменной **txn\_date** в формате **YYYYMMDDHHMMSS**. Эту дату необходимо использовать для проведения сверок и бухгалтерских взаиморасчетов. Так как в системе MMPS учет платежей ведется по дате получения запроса от клиента, то и расчеты с поставщиком услуг необходимо вести по этой дате. Например, ситуация: клиент прислал в систему запрос 31.12.2005 в 23:59:59, учитывая задержку на обработку данных и пересылку информации по каналам связи, система смогла отправить запрос поставщику услуг 01.01.2006 00:00:05, соответственно платеж будет учтен в системе провайдера в другом отчетном периоде, что вызовет некоторые проблемы при проведении сверок. Чтобы избежать такой ситуации система передает поставщику услуг дату, в которой нужно учитывать платеж.
4. Поставщик услуг идентифицирует своего абонента по уникальному идентификатору (номер лицевого счета, телефона, логин и т.д.). Идентификатор абонента передается в переменной **account** в виде строки, содержащей буквы и цифры, длиной до 200 символов.
5. Оплата услуг провайдера производится системой в 2 этапа: проверка состояния абонента и непосредственно проведение платежа. Тип запроса передается системой в переменной **command** в виде строки, принимающей значения «**check**» и «**pay**». При проверке статуса поставщик услуг должен проверить наличие в своей базе абонента с указанным идентификатором и выполнить внутренние проверки идентификатора и суммы платежа в соответствии с принятой логикой пополнения лицевых счетов через платежные системы. При проведении платежа, поставщик услуги должен произвести пополнение баланса абонента.
6. В случае если любой из запросов провайдеру завершается ошибкой, то провайдер возвращает код ошибки в соответствии с таблицей, приведенной в [Приложении 1](#). Все ошибки имеют признак фатальности. Фатальная ошибка означает для системы, что повторная отправка запроса с теми же параметрами приведет к 100% повторению той же ошибки, поэтому система прекращает обработку клиентского запроса и завершает его с ошибкой. Нефатальная ошибка означает для системы, что повторение запроса с теми же параметрами через некоторый промежуток времени, возможно, приведет к успеху. Отсутствие связи с сервером провайдера является нефатальной ошибкой. Отсутствие в ответе элемента `<result>` (некорректный XML и т.д.) - является фатальной ошибкой. Клиентские запросы получают отказ с ошибкой 100 – Другая ошибка провайдера.
7. В базе поставщика услуг не должно содержаться двух успешно проведенных платежей с одним и тем же номером **txn\_id**. Если система присылает повторно запрос с уже существующим в базе провайдера **txn\_id**, то провайдер должен вернуть результат с кодом 333.



8. Для сверки платежей предусмотрено два метода. Метод **GetPaymentsSummary** требует вернуть количество платежей и их сумму за период. Если эти значения расходятся со значениями MMPS COM, то вызывается **GetPayments**, который должен вернуть список всех завершенных платежей за период. Опционально с помощью метода **SendPayments** MMPS COM может отправлять список платежей провайдеру.
9. Предусматривается установка собственных тегов для дополнительной информации внутри тегов `<response></response>`. Следует сообщить о наличии собственных тегов программисту mmps или администратору mmps.

## 4 ОПИСАНИЕ МЕТОДОВ ПРОТОКОЛА

### 1. check:

[https://service.someprovider.ru:8443/payment\\_app.cgi?command=check&txn\\_id=1234567&account=0957835959&sum=10.45&prv\\_id=&txn\\_date=20160101080231](https://service.someprovider.ru:8443/payment_app.cgi?command=check&txn_id=1234567&account=0957835959&sum=10.45&prv_id=&txn_date=20160101080231)

Запрос содержит следующие переменные:

Наименование переменной	Описание переменной
<b>command=check</b>	запрос на проверку состояния абонента
<b>txn_id = 1234567</b>	внутренний номер платежа в системе MMPS
<b>account=0957835959</b>	идентификатор абонента в информационной системе поставщика услуг
<b>sum=10.45</b>	сумма к зачислению на лицевой счет абонента
<b>prv_id =</b>	зарезервировано под идентификатор услуги
<b>txn_date</b>	дата отправки запроса

Ответ провайдера должен выглядеть так:

```
<?xml version="1.0" encoding="UTF-8"?>
<response>
  <osmp_txn_id>1234567</osmp_txn_id>
  <result>0</result>
</response>
```

Или так:

```
<?xml version="1.0" encoding="UTF-8"?>
<response>
  <osmp_txn_id>1234567</osmp_txn_id>
  <result>7</result>
  <comment>The subscriber has gone to Bobruisk...</comment>
</response>
```

result=0 означает, что указанный в запросе с номером osmp\_txn\_id, лицевой счет абонента, может быть пополнен на сумму, указанную в запросе. После успешной проверки состояния, система переходит к отправке запроса на пополнение баланса.

result=7 означает, что прием платежа для данного абонента запрещен по техническим причинам. Данная ошибка является фатальной. Клиентский запрос получает отказ с соответствующим кодом ошибки. [Приложении 1](#).

## 2. pay:

[https://service.someprovider.ru:8443/payment\\_app.cgi?command=pay&txn\\_id=1234567&txn\\_date=20050815120133&account=0957835959&sum=10.45&prv\\_id=](https://service.someprovider.ru:8443/payment_app.cgi?command=pay&txn_id=1234567&txn_date=20050815120133&account=0957835959&sum=10.45&prv_id=)

Запрос содержит следующие переменные:

Наименование переменной	Описание переменной
<b>command=pay</b>	запрос на пополнение баланса абонента
<b>txn_id = <u>1234567</u></b>	внутренний номер платежа в системе ММРС
<b>account=0957835959</b>	идентификатор абонента в информационной системе поставщика услуг
<b>sum=10.45</b>	сумма к зачислению на лицевой счет абонента
<b><u>prv_id</u>=</b>	зарезервировано под идентификатор услуги
<b><u>txn_date</u>=20160101080231</b>	дата отправки запроса

Пример ответа:

```
<?xml version="1.0" encoding="UTF-8"?>
<response>
  <osmp_txn_id>1234567</osmp_txn_id>
  <prv_txn>2016</prv_txn>
  <sum>123.45</sum>
```

```
<result>0</result>
</response>
```

Или так:

```
<?xml version="1.0" encoding="UTF-8"?>
<response>
  <osmp_txn_id>1234567</osmp_txn_id>
  <prv_txn>2016</prv_txn>
  <sum>123.45</sum>
  <result>1</result>
  <comment>Temporarily database error</comment>
</response>
```

result=0, провайдер сообщает об успешном завершении операции пополнения. Система полностью завершает обработку транзакции, сообщая абоненту об успехе.

result=1 – временная ошибка обработки запроса с дополнительным комментарием в поле <comment>. Система будет повторять посылку запроса, пока он не завершится успехом, отказом или пока не истечет время жизни платежа.

Коды состояний представлены в [Приложении 1](#).

### 3. GetPaymentsSummary:

[https://someprovider.md/payment\\_service.php?command=GetPaymentsSummary&from=20150815120133&to=20150816120133](https://someprovider.md/payment_service.php?command=GetPaymentsSummary&from=20150815120133&to=20150816120133)

Запрос содержит следующие переменные:

Наименование переменной	Описание переменной
-------------------------	---------------------

<b>command</b> =GetPaymentsSummary	запрос на получение агрегированных данных по платежам
<b>from</b> =20150815120133	начало отчетного периода
<b>to</b> =20150816120133	конец отчетного периода

Пример ответа:

```
<?xml version="1.0" encoding="UTF-8"?>
<response>
  <serviceID>000-001</serviceID>
  <count>100</count>
  <sum>200.00</sum>
  <result>000</result>
</response>
```

Или так:

```
<?xml version="1.0" encoding="UTF-8"?>
<response>
  <result>001</result>
  <comment>Temporarily database error</comment>
</response>
```

Коды состояний представлены в [Приложении 1](#).



#### 4. GetPayments:

[https://someprovider.md/payment\\_service.php?command=GetPayments&from=20150815120133&to=20150816120133](https://someprovider.md/payment_service.php?command=GetPayments&from=20150815120133&to=20150816120133)

Запрос содержит следующие переменные:

Наименование переменной	Описание переменной
<b>command=GetPayments</b>	запрос на получение платежей
<b>from=20150815120133</b>	начало отчетного периода
<b>to=20150816120133</b>	конец отчетного периода

Пример ответа:

```
<?xml version="1.0" encoding="UTF-8"?>
<response>
<count>2</count>
<payments>
  <payment>
    <serviceID>000-001</serviceID>
    <txnId>123dddddddddddddddddddddddddddd</txnId>
    <providerId>123d67</providerId>
    <account>5454668</account>
    <amount>100.00</amount>
    <date>20150815120133</date>
  </payment>
  <payment>
    <serviceID>000-002</serviceID>
    <txnId>123adddddddddddddddddddddddddd</txnId>
    <providerId>123d67</providerId>
    <account>54004668</account>
    <amount>30.45</amount>
    <date>20150815120133</date>
  </payment>
</payments>
<result>000</result>
</response>
```

Или так:

```
<?xml version="1.0" encoding="UTF-8"?>
<response>
  <result>001</result>
  <comment>Temporarily database error</comment>
</response>
```

Коды состояний представлены в [Приложении 1](#).

## 5. Cancel:

[https://someprovider.md/payment\\_service.php?command=cancel&txn\\_id=1234567](https://someprovider.md/payment_service.php?command=cancel&txn_id=1234567)

Запрос содержит следующие переменные:

Наименование переменной	Описание переменной
<b>command=cancel</b>	запрос на отмену платежа
<b>txn_id =<u>1234567</u></b>	внутренний номер платежа в системе MMPS

Ответ:

```
<?xml version="1.0" encoding="UTF-8"?>
<response>
<osmp_txn_id>1234567</osmp_txn_id>
  <result>0</result>
</response>
```

Коды состояний представлены в [Приложении 1](#).

## 5 ПОРЯДОК ТЕСТИРОВАНИЯ ПОДКЛЮЧЕНИЯ

После реализации шлюза, сообщить об этом платежному администратору, и предоставить данные для доступа и аккаунт для теста. Администратор настроит услугу и начнет тестировать. Сначала вызовет метод check, и в случае успеха метод pay. При возникновении проблем вышлет её описание для проверки и устранения.

## 6 ПРИЛОЖЕНИЕ 1. СПИСОК КОДОВ ЗАВЕРШЕНИЯ

При обработке запросов от системы, поставщик услуг должен сопоставить все возникающие в его приложении ошибки с приведенным ниже списком и возвращать соответствующие коды в элементе <result>. Знак «+» в столбце фатальность – показывает то, как система будет интерпретировать данную ошибку. (см. ОСНОВНЫЕ ПРИНЦИПЫ РАБОТЫ ИНТЕРФЕЙСА п. 6)

Код	Комментарий	Фатальность
000	ОК	
001	Сервис временно не работает. Повторите запрос позже	
003	Неверный формат параметров в запросе	+
004	Идентификатор абонента не найден (Ошиблись номером)	+
005	Счет абонента не активен	
006	Сумма не равна инвойсу	+
007	Сумма слишком мала	+
008	Сумма слишком велика	+
300	Другая ошибка провайдера	+
333	Дубликат Платежа (*1)	

\*1 в ответе на дублированный платеж дополнительно сообщить информацию в теге comment

```

<response>
  <osmp_txn_id>1234567</osmp_txn_id>
  <result>333</result>
  <comment>duplicated payment</comment>
</response>

```