

JSON API
личного кабинета PayKeeper
v. 2.6.0

Описание

Личный кабинет PayKeeper предназначен для управления приемом платежей. Функционал личного кабинета реализуется с помощью JSON API. JSON-запросы выполняются на специальный сервер, на котором расположен PayKeeper. Каждый запрос должен иметь заголовок Authorization: Basic для Basic HTTP-авторизации.

Личный кабинет PayKeeper реализует следующий функционал:

1. Отображение статистики принимаемых платежей по различным платежным системам. Распределения приема платежей по дням и платежным системам.
2. Работа с платежами, отображение дополнительной информации по платежам, возврат и частичный возврат платежей.
3. Отображение реестра принятых платежей с возможностью фильтрации по «Статусам», «Датам», «Платежным системам».
4. Работа с электронными счетами, выставление счетов на e-mail, дублирование старых счетов.
5. Работа с реестром электронных счетов, отображение с возможностью фильтрации по датам и статусам.
6. Изменение конфигурации системы

Любые запросы на чтение данных выполняются GET запросами, на изменение данных на сервере или совершение каких-либо действий POST запросами. Для подписи POST запросов используется специально сгенерированный token (Подробнее в разделе 6. **Токен Безопасности**)

Обработка ошибок

В случае логической ошибки или ошибки во время выполнения в ответе возвращается объект ошибки в JSON:

```
{'result': 'fail', 'msg': 'Error description'}
```

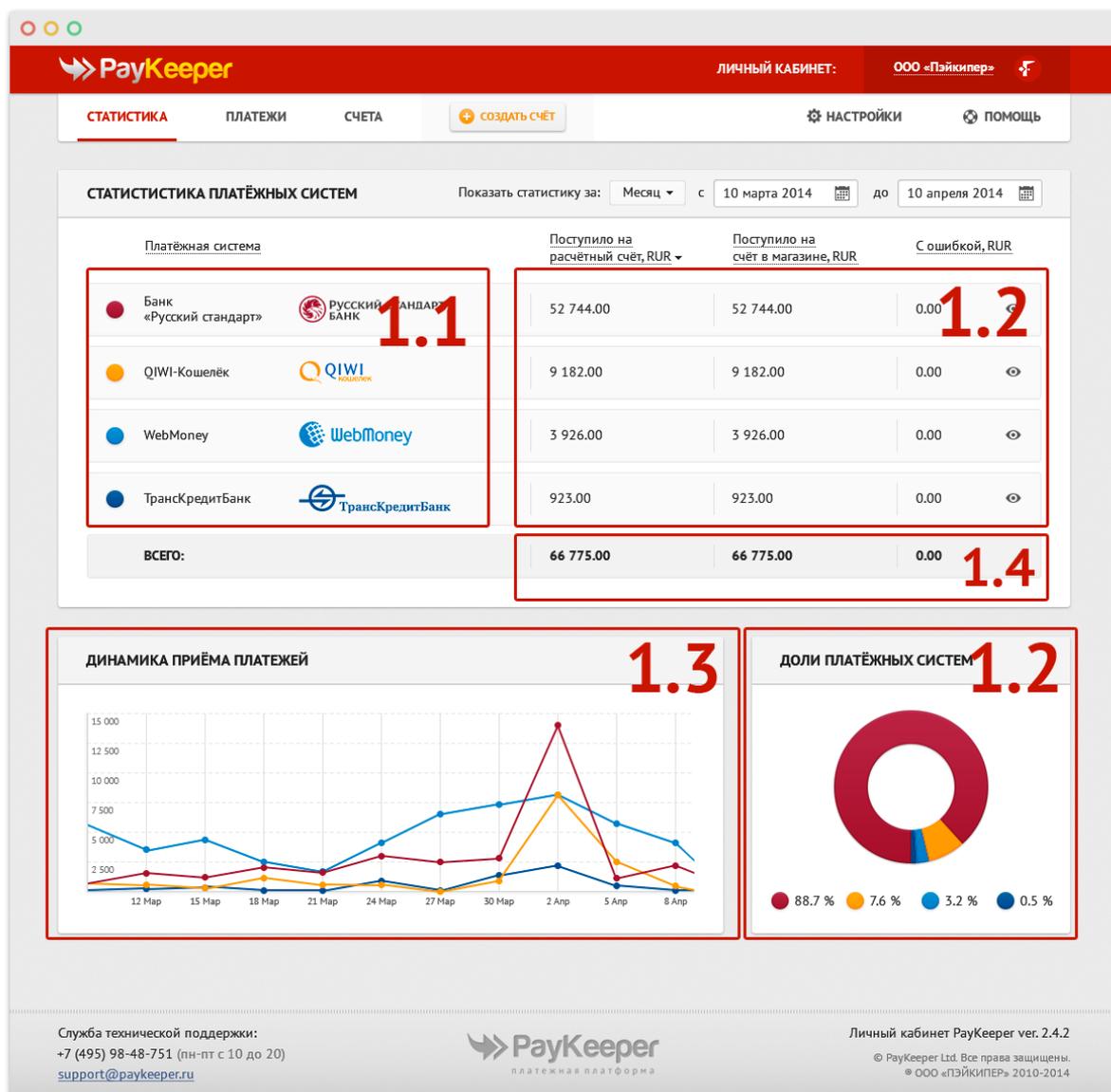
Объект ошибки содержит два поля – **result**, всегда равно **fail**, и поле **msg** со словесным пояснением причины ошибки для отображения пользователю.

1. СТАТИСТИКА

Вкладка «Статистика» предназначена для отображения сводной информации о принятых платежах. Следующие методы используются для отображения информации на странице:

	URI	Назначение
1.1.	/info/systems/list/	Получение списка платежных систем
1.2.	/info/systems/sums/	Получение баланса платежных систем
1.3.	/info/systems/sums/details/	Получение баланса платежных систем, разбитого по временным интервалам
1.4.	/info/systems/allsum/	Получения суммарного баланса всех платежных систем

Схема 1. Вкладка СТАТИСТИКА



1.1. Запрос получения списка платежных систем **/info/systems/list/**

Для получения списка всех платежных систем необходимо выполнить GET-запрос по URL без параметров.

Таблица 1.0.1.1 Параметры запроса списка платежных систем

Тип	Формат запроса
GET	/info/systems/list/

В ответ возвращается объект следующего вида:

Таблица 1.1.2 Параметры ответа на запрос платежных систем

Тип	Формат ответа	
	Параметр	Назначение
1.	id	Идентификатор платежной системы
2.	system_description	Системное название платежной системы
3.	site_description	Название платежной системы для вывода на сайте

Ответ на запрос состоит из массива объектов. Пример ответа на запрос:

```
[
  {
    "id": "2",
    "system_description": "Qivi",
    "site_description": "Visa Qivi Wallet"
  },
  ...
]
```

1.2. Запрос получения баланса платежных систем **/info/systems/sums/**

Запрос возвращает баланс платежных систем за период. Для получения баланса всех платежных систем необходимо выполнить GET-запрос по URL со следующими GET-параметрами.

Таблица 1.2.1. Параметры запроса

Тип	Формат запроса
GET	/info/systems/sums/?start=2014-04-12&end=2014-05-12

	Параметр	Назначение
1.	start	Дата начала периода в формате YYYY-MM-DD
2.	end	Дата конца периода в формате YYYY-MM-DD

В ответ возвращается объект следующего вида:

Таблица 1.2.2. Параметры ответа

Тип	Формат ответа	
	Параметр	Назначение
1.	id	Идентификатор платежной системы
2.	system_description	Системное название платежной системы
3.	site_description	Название платежной системы для вывода на сайте
4.	success	Сумма платежей поступивших на расчетный счёт
5.	account	Сумма платежей поступивших на счёт в магазине
6.	stuck	Сумма платежей принятых с ошибкой
7.	currency	Валюта принятых платежей

Ответ на запрос состоит из массива объектов. Пример ответа на запрос:

Таблица 1.2.3. Пример ответа на запрос:

```
[
  {
    "id": "1",
    "system_description": "WebMoney",
    "site_description": " WebMoney Transfer",
    "success": "566.00",
    "account": "566.00",
    "stuck": "0.00",
    "currency": "RUR"
  },
  ...
]
```

1.3. Запрос получения баланса платежных систем для построения графика приема платежей **/info/systems/sums/details/**

Запрос служит для получения данных, необходимых для построения графика приема платежей по времени. В случае если задан интервал меньше 30 дней, возвращает сумму платежей принятых за каждый день. Если указанный интервал больше 30 дней, то производит усреднение и выводит

результат в виде 30 точек с указанным временем и объемом принятых платежей за этот интервал. Для получения данных необходимо передать GET-запрос по URL со следующими GET-параметрами.

Таблица 1.3.1. Параметры запроса

Тип	Формат запроса	
GET	/info/systems/sums/details/?start=2014-04-12&end=2014-05-12	
	Параметр	Назначение
1.	start	Дата начала периода в формате YYYY-MM-DD
2.	end	Дата конца периода в формате YYYY-MM-DD

В ответ возвращается массив объектов следующего вида:

Таблица 1.3.2. Параметры ответа

Тип	Формат ответа	
	Параметр	Назначение
1.	payment_system_id	Идентификатор платежной системы
2.	label	Название платежной системы
3.	data	Содержит массив составных объектов следующего формата: [TIMESTAMP,AMOUNT], где TIMESTAMP – время в секундах в формате Unix Timestamp, AMOUNT – объем принятых платежей за интервал времени.

Пример ответа на запрос:

```
[
  {
    "payment_system_id": "6",
    "label": "Russian Standart Bank",
    "data": [
      [1394380800000,120],
      [1394467200000,101],
      [1394553600000,331],
      [1394640000000,290],
      [1394726400000,305],
      [1394812800000,201]
    ]
  },
  ...
]
```

1.4. Запрос получения суммарного баланса всех платежных систем `/info/systems/allsum/`

Запрос возвращает суммарный баланс всех платежных систем за период. Для получения баланса всех платежных систем необходимо выполнить GET-запрос по URL со следующими GET-параметрами.

Таблица 1.4.1. Параметры запроса

Тип	Формат запроса	
GET	<code>/info/systems/allsum/?start=2014-04-12&end=2014-05-12</code>	
	Параметр	Назначение
1.	start	Дата начала периода в формате YYYY-MM-DD
2.	end	Дата конца периода в формате YYYY-MM-DD

В ответ возвращается объект следующего вида:

Таблица 1.4.2. Параметры ответа

Тип	Формат ответа	
	Параметр	Назначение
1.	success	Сумма платежей поступивших на расчетный счёт
2.	account	Сумма платежей поступивших на счёт в магазине
3.	stuck	Сумма платежей принятых с ошибкой

Пример ответа на запрос:

```
{
  "success": "10932.00",
  "account": "10932.00",
  "stuck": "0.00"
}
```

2. ПЛАТЕЖИ

Вкладка «Платежи» предназначена для работы с платежами. Получать реестр платежей можно с использованием фильтров по датам, статусам, платежным системам. По каждому платежу можно получить дополнительную расширенную информацию. Следующие методы используются для отображения информации на странице:

	URI	Назначение
2.1	/info/payments/bydate/	Получение реестра платежей за период, с фильтром по статусам и платёжным системам.
2.2	/info/payments/bydatecount/	Получение количества платежей за период, сгруппированное по статусам и по платежным системам
2.3	/info/payments/byid/	Получение информации о платеже по идентификатору
2.4	/info/options/byid/	Получение дополнительной информации об опциях платежа
2.5	/info/params/byid/	Получение дополнительных необязательных параметров платежа
2.6	/info/httplog/byid/	Отображает информацию об HTTP запросе по идентификатору платежа
2.7	/info/refunds/bypaymentid/	Отображает информацию по возвратам для данного платежа
2.8	/change/payment/reverse/	Полный или частичный возврат платежа
2.9	/change/payment/repeatcnt/	Сброс счетчик повтора оповещений для платежа
2.10	НЕ ПОДДЕРЖИВАЕТСЯ	НЕ ПОДДЕРЖИВАЕТСЯ
2.11	/info/payments/search/	Поиск платежа по значениям параметров в диапазоне дат
2.12	/change/payment/capture/	Подтверждение списания средств по ранее авторизованной операции
2.13	/change/payment/post-sale-receipt/	Сгенерировать чек окончательного расчёта

Схема 2. Вкладка «ПЛАТЕЖИ»

The screenshot shows the 'ПЛАТЕЖИ' (Payments) section of the PayKeeper personal cabinet. The interface includes a navigation bar, a filter section for payment status and system, a main table of transactions, and detailed information panels for selected payments. Red and blue boxes with numbers 2.1 through 2.7 highlight specific elements:

- 2.1**: Filter section 'По статусу:' (By status) and 'По платёжной системе:' (By payment system).
- 2.2**: Filter section 'По статусу:' (By status).
- 2.3**: Search bar 'Найти платёж по номеру:' (Find payment by number).
- 2.3**: Information panel for payment 89133, including date, amount, and system.
- 2.8**: Button 'ПОЛНЫЙ ИЛИ ЧАСТИЧНЫЙ ВОЗВРАТ ПЛАТЕЖА' (Full or partial refund of payment).
- 2.9**: Button 'СБРОСИТЬ СЧЁТЧИК ПОВТОРОВ' (Reset counter of retries).
- 2.4**: Information panel for payment 89130, including result codes and transaction details.
- 2.7**: Information panel for payment 89128, including a table of refunds.

Table 1: Main Payment Table

Номер	ПС	Клиент	Номер заказа	Сумма, RUR	Статус платежа
89137		yang junfeng	Пополнение счёта	49.00	Превышено число повторов
89136		Мани Дмитрий	136	158 999.00	Совершён
89135		Ivanov Andrey	135	1.00	Ожидает оплаты
89134		Олег Кузьмин	134	4 698.00	Не состоялся
89133		Олег Загидуллин	133	56 300.00	Получен
89132		test027 test027	132	0.99	Совершён
89131		Т.Д.	131	19 800.00	Ожидает оплаты
89130		Мани Дмитрий	130	1 000 999.00	Отменён
89129		Рувль Денис	129	960.00	Ожидает оплаты
89128		Шава Александр	128	100.01 в т.ч. возврат: 35 000 000.00	Частично возвращён клиенту

Table 2: Refunds for Payment 89128

№	Дата возврата	Пользователь	Сумма возврата, RUR	Остаток, RUR
3	31 декабря 2014 в 23:23	demo	34 000 000.00	0.99
2	1 мая 2014 в 08:07	aisatpiae@yahoo.com	135.00	1 002.00
1	28 марта 2014 в 19:48	marina.kibeteva87	22 550.00	445 655.00

2.1. Запрос получения реестра платежей /info/payments/bydate/

Запрос возвращает реестр платежей за период. Для удобства вывода в таблицу можно указать максимальное число объектов в выборке и пропустить первые сколько-то значений. Также можно выбрать платежи только с определёнными статусами. Для получения данных необходимо выполнить GET-запрос со следующими параметрами.

Таблица 2.1.1. Параметры запроса

Тип	Формат запроса	
GET	/info/payments/bydate/?limit=100&from=0&start=2014-04-14&end=2014-06-14&status[]=success&status[]=canceled&status[]=failed&payment_system_id[]=9&payment_system_id[]=1	
	Параметр	Назначение
1.	start	Дата начала периода в формате YYYY-MM-DD
2.	end	Дата конца периода в формате YYYY-MM-DD
3.	payment_system_id[]	Идентификатор платежной системы. Применяется для фильтрации по платежной системе. Обязательный параметр.
4.	status[]	Статус платежей. Применяется для фильтрации платежей по статусам. Может принимать значения: 'pending', 'obtained', 'canceled', 'success', 'failed', 'stuck', 'refunded', 'refunding', 'partially_refunded'. Обязательный параметр. См. Приложение 1.
5.	from	Пропускает первые N значений. Обязательный параметр.
6.	limit	Количество возвращаемых значений. Обязательный параметр.

В случае если не указан параметр status[] или payment_system_id[] выводится пустой массив.

В ответ возвращается массив объектов следующего вида:

Таблица 2.1.2. Параметры ответа на запрос

Тип	Формат ответа	
	Параметр	Назначение
1.	id	Идентификатор платежа
2.	pay_amount	Сумма платежа
3.	refund_amount	Возвращенная часть платежа
4.	clientid	Идентификатор клиента в системе магазина
5.	orderid	Номер заказа в системе магазина
6.	payment_system_id	Идентификатор платежной системы
7.	unique_id	Уникальный идентификатор транзакции
8.	status	Статус платежа
9.	repeat_counter	Количество отправленных запросов при проведении платежа в систему магазина

10.	pending_datetime	Дата/Время создания платежа
11.	obtain_datetime	Дата/Время проведения платежа
12.	success_datetime	Дата/Время информирования магазина о проведенном платеже

Пример ответа на запрос:

```
[
  {
    "id": "97452",
    "pay_amount": "1.00",
    "refund_amount": "0.00",
    "clientid": "Ivan Ivonov",
    "orderid": null,
    "payment_system_id": "3",
    "unique_id": null,
    "status": "success",
    "repeat_counter": "0",
    "pending_datetime": "2014-05-14 15:59:00",
    "obtain_datetime": "2014-05-14 15:59:01",
    "success_datetime": "2014-05-14 15:59:01"
  },
  ...
]
```

2.2. Запрос получения количества платежей за период **/info/payments/bydatecount/**

Запрос возвращает количество платежей за период. Запрос возвращает данные, сгруппированные по статусам и сгруппированные по платежным системам. Также запрос возвращает полную сумму платежей за этот период. Для получения данных необходимо выполнить GET-запрос со следующими параметрами.

Таблица 2.2.1. Параметры запроса

Тип	Формат запроса	
GET	/info/payments/bydatecount/?start=2014-04-14&end=2014-05-14&status[]=success&status[]=canceled&status[]=failed&payment_system_id[]=6&payment_system_id[]=1	
	Параметр	Назначение

1.	start	Дата начала периода в формате YYYY-MM-DD
2.	end	Дата конца периода в формате YYYY-MM-DD
3.	payment_system_id[]	Идентификатор платежной системы. Применяется для фильтрации по платежной системе.
4.	status[]	Статус платежей. Применяется для фильтрации платежей по статусам. Может принимать значения: 'pending', 'obtained', 'canceled', 'success', 'failed', 'stuck', 'refunded', 'refunding', 'partially_refunded'.

В ответ возвращается составной объект следующего вида:

- 1) Массив с платежами, сгруппированный по статусам.
- 2) Массив с платежами, сгруппированный по платежным системам.
- 3) Полный счетчик всех платежей с учетом фильтров.

Таблица 2.2.2. Параметры ответа на запрос

Тип	Формат ответа	
	Параметр	Назначение
1.	statuses	Массив, сгруппированный по различным статусам, состоящий из элементов [status=>"value",count=>"value"]
2.	payment_systems	Массив, сгруппированный по различным платежным системам [payment_system_id=>"value",count=>"value"]
3.	fullcount	Полное количество платежей

Пример ответа на запрос:

```
{
  "statuses":
  [
    {"status":"canceled","count":"14"},
    {"status":"success","count":"71"},
    {"status":"failed","count":"3"}
  ]
},
{
  "payment_systems":
  [
    {"payment_system_id":"1","count":"47"},
    {"payment_system_id":"6","count":"41"}
  ]
},
}
```

```
{
  "fullcount":
    [
      {"count": "88"}
    ]
}
```

2.3. Запрос получения информации о платеже по идентификатору `/info/payments/byid/`

Запрос возвращает информацию о платеже по его идентификатору в PayKeeper. По формату ответа аналогичен предыдущему запросу. Для получения данных необходимо выполнить GET-запрос со следующими параметрами.

Таблица 2.3.1. Параметры запроса

Тип	Формат запроса	
GET	<code>/info/payments/byid/?id=105842</code>	
	Параметр	Назначение
1.	id	Идентификатор платежа в системе PayKeeper

В ответ возвращается объект следующего вида:

Таблица 2.3.2. Параметры ответа на запрос

Тип	Формат ответа	
	Параметр	Назначение
1.	id	Идентификатор платежа
2.	pay_amount	Сумма платежа
3.	refund_amount	Возвращенная часть платежа
4.	clientid	Идентификатор клиента в системе магазина
5.	orderid	Номер заказа в системе магазина
6.	payment_system_id	Идентификатор платежной системы
7.	unique_id	Уникальный идентификатор транзакции
8.	status	Статус платежа
9.	repeat_counter	Количество отправленных запросов при проведении платежа в систему магазина
10.	pending_datetime	Дата/Время создания платежа
11.	obtain_datetime	Дата/Время проведения платежа

12.	success_datetime	Дата/Время информирования магазина о проведенном платеже
-----	------------------	--

Пример ответа на запрос:

```
{
  "id": "97452",
  "pay_amount": "10000.00",
  "refund_amount": "5000.00",
  "clientid": "Konstantin Semenov",
  "orderid": null,
  "payment_system_id": "3",
  "unique_id": null,
  "status": "success",
  "repeat_counter": "0",
  "pending_datetime": "2014-05-14 15:59:00",
  "obtain_datetime": "2014-05-14 15:59:01",
  "success_datetime": "2014-05-14 15:59:01",
  "refund_amount": "0.00"
}
```

2.4. Запрос получение дополнительной информации об опциях платежа `/info/options/byid/`

Запрос возвращает дополнительную информацию об опциях платежа по его идентификатору в PayKeeper. Для получения данных необходимо выполнить GET-запрос со следующими параметрами.

Таблица 2.4.1. Параметры запроса

Тип	Формат запроса	
GET	<code>/info/options/byid/?id=105842</code>	
	Параметр	Назначение
1.	id	Идентификатор платежа в системе PayKeeper

Ответ на запрос может содержать различные параметры, в зависимости от платежной системы и статуса платежа. Пример возвращаемого объекта:

```
{
  client_ip: "81.5.119.7",
  orderid: "140514-74-9",
}
```

```

service_name: "Услуги по переводу",
client_email: "info@example.com",
client_phone: "+79112511000",
qiwi_phone: "",
bill_name: "",
msgtype: "Visa",
RRN: "413415326998",
APPROVAL_CODE: "356884",
CARD_NUMBER: "5*****8624",
RESULT: "OK",
RESULT_CODE: "000",
3DSECURE: "AUTHENTICATED"
}

```

2.5. Запрос получения дополнительных необязательных параметров платежа /info/params/byid/

Запрос возвращает дополнительные параметры платежа, передаваемые платежными системами. Для получения данных необходимо выполнить GET-запрос со следующими параметрами.

Таблица 2.5.1. Параметры запроса

Тип	Формат запроса	
GET	/info/params/byid/?id=105842	
	Параметр	Назначение
1.	id	Идентификатор платежа в системе PayKeeper

Ответ на запрос содержит параметры платежа в следующем формате.

Таблица 2.5.2. Параметры ответа на запрос

Тип	Формат ответа	
	Параметр	Назначение
1.	id	Идентификатор
2.	payment_id	Идентификатор платежа
3.	field	Ключ
4.	value	Значение

Пример возвращаемого объекта:

```
[
  {
    id: "189445"
    payment_id: "97445"
    field: "RESULT"
    value: "OK"
  },
  {
    id: "189446"
    payment_id: "97445"
    field: "RESULT_CODE"
    value: "000"
  },
  ...
]
```

2.6. Запрос получения HTTP параметров платежа /info/httplog/byid/

Запрос возвращает HTTP параметры запросов платежа по его идентификатору в системе PayKeeper. Для получения данных необходимо выполнить GET-запрос со следующими параметрами.

Таблица 2.6.1. Параметры запроса

Тип	Формат запроса	
GET	/info/httplog/byid/?id=105842	
	Параметр	Назначение
1.	id	Идентификатор платежа в системе PayKeeper

Ответ на запрос содержит массив объектов следующего вида.

Таблица 2.6.2. Параметры ответа на запрос

Тип	Формат ответа	
	Параметр	Назначение
1.	id	Идентификатор
2.	payment_id	Идентификатор платежа
3.	request_method	Тип HTTP запроса
4.	request_uri	URI Запроса

5.	ip	IP Адрес запроса
6.	body	Тело запроса
7.	datetime	Дата/Время запроса

Пример возвращаемого объекта:

```
[
{
  id: "32"
  payment_id: "5"
  request_method: "GET"
  request_uri: "/external/"
  ip: "62.76.178.170"
  body: ""
  datetime: "2012-08-30 15:42:58"
}
]
```

2.7. Запрос информации по возвратам для платежа /info/refunds/bypaymentid/

Запрос возвращает информацию о возвратах, выполненных по данному платежу. Для получения данных необходимо выполнить GET-запрос со следующими параметрами.

Таблица 2.7.1. Параметры запроса

Тип	Формат запроса	
GET	/info/refunds/bypaymentid/?id=436	
	Параметр	Назначение
1.	id	Идентификатор платежа в системе PayKeeper

Ответ на запрос содержит массив объектов следующего вида.

Таблица 2.7.2. Параметры ответа на запрос

Тип	Формат ответа	
	Параметр	Назначение
1.	id	Идентификатор возврата
2.	payment_id	Идентификатор платежа
3.	refund_number	Порядковый номер возврата по данному платежу
4.	user	Пользователь, инициировавший возврат

5.	amount	Сумма возврата
6.	reminder	Остаток после возврата
7.	datetime	Дата/Время проведения возврата
8.	status	Статус возврата. Принимает значения «started», «done», «failed»

Пример возвращаемого объекта:

```
[  
  {  
    "id": "1",  
    "payment_id": "436",  
    "refund_number": "1",  
    "user": "admin",  
    "amount": "22570.00",  
    "reminder": "0.00",  
    "datetime": "2014-06-11 16:09:13",  
    "status": "done"  
  }  
]
```

2.8. Запрос на возврат платежа /change/payment/reverse/

Запрос возвращает платеж полностью или частично. Возврат могут делать только пользователи с включенной функцией возврата. Для возврата платежа необходимо выполнить POST-запрос со следующими параметрами.

Данный метод автоматически определяет, будет выполнен полный возврат или отмена авторизации, частичный возврат или частичное списание в зависимости от состояния платежа на момент выполнения возврата и возможностей банка-эквайера.

Если система работает в режиме автоматической генерации фискальных чеков 54-ФЗ, то в этом запросе может дополнительно передаваться информация о товарах к возврату. В случае, если у исходного платежа была указана корзина с товарами, то при полном возврате корзину можно не указывать, а при частичном корректная корзина необходима для формирования корректного чека 54-ФЗ. Если корзина товара указана неверно, то на проведение возврата это не повлияет, но чек сгенерирован не будет. Если платёж на момент выполнения запроса был авторизован, то чек будет сгенерирован только на сумму частичного списания.

Таблица 2.8.1. Параметры запроса

Тип	Формат запроса	
POST	/change/payment/reverse/	
	Параметр	Назначение
1.	id	Идентификатор платежа в системе PayKeeper
2.	amount	Возвращаемая сумма, разделитель точка
3.	partial	Признак частичного возврата, может принимать значения true/false. В случае полного возврата, сумма возврата не должна быть указана.
4.	token	Токен безопасности. (Подробнее в главе 6 данного руководства)
5.	refund_cart	Массив товаров к возврату. JSON-строка в таком же формате, что и при инициализации платежа.

Результатом данного запроса будет объект

```
{
  'result' : success
}
```

в случае если возврат инициирован успешно, и объект ошибки, если начать возврат не удалось.

Возврат происходит в асинхронном режиме, поэтому сама по себе инициализация возврата не является гарантией его успешного возврата. Для автоматического контроля возвратов рекомендуется использовать поисковый метод 2.1 (с выбором платежей со статусом **refunding**), чтобы отслеживать появление незавершённых возвратов, либо контролировать проведение возврата по конкретному платежу запросом 2.7.

На выполнение возврата обычно требуется до нескольких минут, поэтому рекомендуется проверять результат выполнения возврата через 3 минуты. Также PayKeeper может отправлять callback-запрос с уведомлением о состоявшемся возврате. Эта возможность включается в конфигурации сервера с PayKeeper.

2.9. Запрос на сброс счетчика повторов для платежа /change/payment/repeatcnt/

Запрос сбрасывает счетчик оповещений для платежа и инициирует работу информера PayKeeper. Для сброса счетчика необходимо выполнить POST-запрос со следующими параметрами.

Таблица 2.9.1. Параметры запроса

Тип	Формат запроса
POST	/change/payment/repeatcnt/

	Параметр	Назначение
1.	id	Идентификатор платежа в системе PayKeeper
2.	token	Токен безопасности. (Подробнее в главе 6 данного руководства)

Результатом данного запроса будет объект

```
{
  'result' : success
}
```

в случае, если счетчик сброшен успешно, и объект ошибки, если сбросить счетчик не удалось.

2.10. Не поддерживается

2.11. Поиск платежей по значениям параметров в диапазоне дат `/info/payments/search/`

Запрос возвращает платежи, чьи номера, параметры оплаты, номера карт или коды авторизаций содержат указанную в запросе подстроку. Поиск осуществляется в пределах диапазона дат, который задаётся параметрами запроса.

Таблица 2.11.1. Параметры запроса

Тип	Формат запроса	
GET	<code>/info/payments/search/?query=4038*9682&beg_date=2016-01-01&end_date=2016-04-04</code>	
	Параметр	Назначение
1.	query	Подстрока для поиска в параметрах платежа id и clientid, а также в опциях orderid, service_name, client_email, client_phone, CARD_NUMBER, APPROVAL_CODE. Поиск поддерживает синтаксис MySQL like. Символ * заменяется на %.
2.	beg_date	Дата начала периода в формате YYYY-MM-DD
3.	end_date	Дата конца периода в формате YYYY-MM-DD

Ответ на запрос содержит массив объектов следующего вида.

Таблица 2.11.2. Параметры ответа на запрос

Тип	Формат ответа	
	Параметр	Назначение
1.	id	Номер платежа
2.	clientid	Идентификатор плательщика
3.	orderid	Номер заказа

4.	unique_id	Идентификатор транзакции, назначенный Банком
5.	pay_amount	Сумма платежа
6.	status	Статус платежа
7.	payment_system_id	Идентификатор платёжной системы
8.	repeat_counter	Количество попыток оповещения ТСП об успешности платежа
9.	pending_datetime	Дата/время начала операции
10.	success_datetime	Дата/время выполнения операции

Пример возвращаемого массива:

```
[
  {
    "clientid" : "колосов александр",
    "id" : "35534",
    "obtain_datetime" : "2016-01-05 11:26:04",
    "orderid" : null,
    "pay_amount":"22570.00",
    ...
  }, ...
]
```

2.12. Списание средств по ранее проведённой авторизации /change/payment/capture/

В случае, если необходимо списать средства по ранее выполненной авторизации, не дожидаясь автосписания, следует воспользоваться данным запросом. Для выполнения частичного списания нужно использовать запрос 2.8

Таблица 2.12.1. Параметры запроса

Тип	Формат запроса	
POST	/change/payment/capture/	
	Параметр	Назначение
1.	id	Номер платежа к списанию

Ответ на запрос содержит объект с результатом операции, либо объект ошибки. Успешный результат операции в случае **status = queued** (при работе с Промсвязьбанком) означает лишь принятие запроса Банком, но не его исполнение. Проверку успешности списания необходимо выполнить спустя одну минуту при помощи запроса 2.4

Таблица 2.12.2. Параметры ответа на запрос

Тип	Формат ответа	
	Параметр	Назначение
1.	result	Успешность выполнения запроса: 'success' / 'fail'
2.	status	В случае, если запрос выполнен без ошибки, captured означает успешное списание, queued – отправку запроса
3.	msg	Сообщение с ошибкой в случае, если result = fail

Пример возвращаемого объекта:

```
{
  "result" : "success",
  "status" : "captured"
}
```

2.13. Генерация чека окончательного расчёта /change/payment/post-sale-receipt/

Согласно закону 54-ФЗ о применении контрольно-кассовой техники при передаче товара, предварительно оплаченного через Интернет, необходимо выдать кассовый чек. Также, согласно закону, если при передаче товара не происходит дополнительного расчёта, чек может быть выдан в электронном виде. Данный метод предназначен для простого оформления чека при передаче товара.

Если у всех товарных позиций в чеке по платежу код признака расчёта был **полной предоплатой**, с помощью этого метода можно выдать соответствующий чек окончательного расчёта. Товарные позиции будут соответствовать товарным позициям в исходном чеке.

Таблица 2.133.1. Параметры запроса

Тип	Формат запроса	
	Параметр	Назначение
POST	/change/payment/post-sale-receipt/	
1.	id	Номер платежа, по которому выдаётся чек окончательного расчёта.

Все товарные позиции чека окончательного расчёта будут такими же, как и в первом чеке по данному платежу. Для чеков окончательного расчёта по авансовому платежу эту функцию использовать нельзя, т.к. при окончательном расчёте должен быть указан точный перечень товаров или услуг. Сумма чека «зачёт аванса» окончательного расчёта будет равна сумме чека «электронными» первого чека по платежу. Если платёж был частично возвращён, в качестве корзины товаров будут использоваться оставшиеся товары.

Чек формируется в асинхронном режиме. Если запрос был сформирован корректно и был принят, в ответ будет возвращён объект с полем **result = success** и номером сгенерированного запроса на чек **receipt_id**. Результат формирования чека нужно проверять запросом 8.4.

3. СЧЕТА

Вкладка «Счета» предназначена для работы со счетами. На этой странице можно создавать счета, отправлять счета, просматривать историю счетов.

Следующие методы необходимы для работы со счетами:

Таблица 3.1. JSON-методы работы со счетами

	URI	Назначение
3.1	/info/invoice/byid/	Получить данные счёта по его номеру
3.2	/info/invoice/list/	Получение истории счетов за период
3.3	/info/invoice/listcount/	Получение количества счетов за период
3.4	/change/invoice/preview/	Создание счёта и генерация предпросмотра письма счёта
3.5	/change/invoice/send/	Отправка созданного счёта

Работа с ними происходит посредством GET- либо POST-запросов из браузера пользователя. В ответ возвращается JSON. В зависимости от запроса, вид объекта в ответе может различаться.

Счёт имеет следующие атрибуты:

Таблица 3.2. Атрибуты счёта

Атрибут	Назначение	Допустимые значения
id	Уникальный номер счёта	Строка из символов 0-9
status	Статус заказа	Строковые значения sent, paid, expired
pay_amount	Сумма счёта	Число, разделитель дробной части – точка
clientid	Идентификатор клиента	Строка, любые буквы и цифры
ordered	Номер заказа	Строка, любые буквы и цифры
service_name	Наименование услуги	Строка, любые буквы и цифры
client_email	Е-mail клиента	Строка, корректно записанный e-mail
client_phone	Телефон клиента	Строка, любые буквы и цифры
expiry_datetime	Счёт действует до	Дата в формате YYYY-MM-DD HH:MM:SS
created_datetime	Дата создания счёта	Дата в формате YYYY-MM-DD HH:MM:SS
paid_datetime	Дата оплаты счёта	Дата в формате YYYY-MM-DD HH:MM:SS

Схема 3. Вкладка «СЧЕТА»

PayKeeper ЛИЧНЫЙ КАБИНЕТ: ООО «Пэйкипер»

СТАТИСТИКА ПЛАТЕЖИ **СЧЕТА** СОЗДАТЬ СЧЁТ НАСТРОЙКИ ПОМОЩЬ

СЧЕТА Показать счета за: Месяц с 10 марта 2014 до 10 апреля 2014

3.3 Фильтр счетов
 Оплачен: 150 Отправлен: 10 Просрочен: 25

Показать по: 10 счетов в таблице **3.2**

Дата выставления	Клиент	Сумма, RUR	Статус платежа	Срок действия	
31 марта 2014 в 15:48:54	null	1 000.00	Отправлен	Не ограничен	+
27 марта 2014 в 15:48:54	123	100.00	Отправлен	30 марта 2014 не включительно	+
27 марта 2014 в 15:48:54	Иванов Иван Иванович	18 360.00	Отправлен	30 марта 2014 не включительно	+
26 марта 2014 в 15:48:54	Вячеслав Степаненко	6 999.00	Оплачен	30 марта 2014 не включительно	+
21 марта 2014 в 15:48:54	Латышев Сергей Игоревич	780.00	Отправлен	Не ограничен	+
21 марта 2014 в 15:48:54	Зуев Олег	34.00	Отправлен	30 марта 2014 не включительно	+
21 марта 2014 в 15:48:54	Фёдор Павлович Карамазов	74 220.00	Просрочен	30 марта 2014 не включительно	+
21 марта 2014 в 15:48:54	Олег Загидуллин	16 000.00	Оплачен	30 марта 2014 не включительно	-

Информация о счёте **4.5**

Номер заказа:	Наименование услуги:	Телефон:	Электронная почта:
111	Аренда	+7 (901) 123-45-67	tech@paykeeper.ru

20 марта 2014 в 15:48:54	Yakovlev Sergei	64 000 000.00	Отправлен	30 марта 2014 не включительно	3.1
20 марта 2014 в 15:48:54	Горх Полина	118 999.00	Отправлен	30 марта 2014 не включительно	+

1 2 3 4 5 6 7 8 9 10 **3.3**

В таблице 13 048 счетов на 1 305 страницах

Служба технической поддержки:
+7 (495) 98-48-751 (пн-пт с 10 до 20)
support@paykeeper.ru

PayKeeper
платежная платформа

Личный кабинет PayKeeper ver. 2.4.2
© PayKeeper Ltd. Все права защищены.
© ООО «ПЭЙКИПЕР» 2010-2014

3.1. Запрос получения данных счёта `/info/invoice/byid/`

Для получения данных счёта необходимо выполнить GET-запрос по URL с GET-параметром `id`, равным номеру нужного счёта.

Таблица 3.1.1 Параметры запроса истории счетов

Тип	Пример запроса	
GET	<code>/info/invoice/byid/?id=20140402085214805</code>	
	Параметр	Назначение
1.	<code>id</code>	Id счёта, по которому нужно получить данные.

В ответ возвращается объект счёта с атрибутами, как в таблице.

```
{
  'id'           : '20110115085347329',
  'clientid'    : 'Ivanov Ivan Ivanovich',
  'orderid'     : '1337',
  'service_name' : 'Document delivery',
  'client_email' : 'ivanov@ivan.ivanovich.com',
  'client_phone' : '+79254973590',
  'created_datetime' : '2011-01-15 08:53:47',
  'paid_datetime'  : '2011-01-16 14:12:00',
  'expiry_datetime' : '2011-01-19 00:00:00',
  'pay_amount'   : '1205.98',
  'status'      : 'paid'
}
```

3.2. История счетов `/info/invoice/list/`

Данный запрос позволяет получить список счетов за указанный период. Для удобства вывода в таблицу можно указать максимальное число объектов в выборке и пропустить первые сколько-то значений. Также, указав статусы, можно выбрать счета только с определёнными статусами. Счета выводятся в порядке от самого нового к самому старому.

Таблица 3.2.1. Параметры запроса истории счетов

Тип	Пример запроса	
GET	/info/invoice/list/?status[]=paid&start_date=2014-04-15&end_date=2014-04-25&from=30&limit=10	
	Параметр	Назначение
1.	status[]	Выбрать счета только с указанными статусами, может принимать значения sent, paid, expired.
2.	start_date	Выбрать счета, выставленные после даты в формате YYYY-MM-DD
3.	end_date	Выбрать счета, выставленные до даты в YYYY-MM-DD
4.	from	Пропустить N результатов в начале выборки
5.	limit	Выдавать не более указанного числа результатов

В ответ на данный запрос возвращается массив объектов вида 3.2. Пример возвращаемого объекта:

```
[
  {
    id: "20140415042846859",
    status: "expired",
    pay_amount: "5999.00",
    clientid: "Кристина Бушуева",
    orderid: "3946",
    service_name: null,
    client_email: "hidden@paykeeper.ru",
    client_phone: null,
    expiry_datetime: "2014-04-19 00:00:00",
    created_datetime: "2014-04-15 04:28:46",
    paid_datetime: null
  },
  ...
]
```

Параметры запроса status, limit и from являются обязательными. Допустимые значения для поля status - **sent**, **paid**, **expired**

Приведенный в примере запрос найдет только оплаченные счета, выставленные в период от 15 апреля 2014 года до 25 апреля 2014 года, отбросит первые 30 результатов и выведет следующие 10. Если оплаченных счетов в этот период окажется менее 30, запрос вернет пустой массив.

3.3. Количество счетов /info/invoice/listcount/

Данный запрос возвращает количество счетов за указанный период. Запрос возвращает данные разбитые по статусам счетов **sent**, **paid**, **expired**. Указав статусы, можно подсчитать счета только с определенными статусами. Для получения данных необходимо выполнить GET-запрос со следующими параметрами.

Таблица 3.3.1. Параметры запроса истории счетов

Тип	Пример запроса	
GET	/info/invoice/listcount/?status[]=expired&status[]=sent&status[]=paid&start_date=2014-04-15&end_date=2014-04-25	
	Параметр	Назначение
1.	status[]	Подсчитать счета только с указанными статусами, может принимать значения sent, paid, expired.
2.	start_date	Выбрать счета, выставленные после даты в формате YYYY-MM-DD
3.	end_date	Выбрать счета, выставленные до даты в YYYY-MM-DD

В ответ возвращается составной объект следующего вида:

- 1) Массив счетов, сгруппированный по статусам.
- 2) Полный счетчик всех счетов с учетом фильтров.

Таблица 3.3.2. Параметры ответа на запрос

Тип	Формат ответа	
	Параметр	Назначение
1.	statuses	Массив, сгруппированный по различным статусам, состоящий из элементов [status=>"value",count=>"value"]
2.	fullcount	Полное количество платежей

Пример ответа на запрос:

```
{
  "statuses":
  [
    {"status":"paid","count":"7"},
    {"status":"expired","count":"9"},
    {"status":"sent","count":"1"}
  ],
  "fullcount":
  [
    {"count":"17"}
  ]
}
```

3.4. Предварительный просмотр письма со счётом `/change/invoice/preview/`

Этот запрос создаёт по переданным параметрам оплаты счёт в базе данных и генерирует для отображения в виде предварительного просмотра html-код письма счёта.

Таблица 3.4.1. Параметры запроса генерации предпросмотра

Тип	Пример запроса	
POST	<code>/change/invoice/preview/</code>	
	Параметр	Назначение
1.	<code>pay_amount</code>	Сумма заказа к оплате
2.	<code>clientid</code>	Идентификатор клиента
3.	<code>orderid</code>	Номер заказа
4.	<code>service_name</code>	Наименование услуги
5.	<code>client_email</code>	e-mail адрес клиента
6.	<code>client_phone</code>	Телефон клиента
7.	<code>expiry</code>	Срок действия счёта в формате YYYY-MM-DD (не включительно)
8.	<code>token</code>	Токен безопасности. (Подробнее в главе 6 данного руководства)

В ответ на данный запрос возвращается объект с полями `invoice_id`, содержащим номер созданного счёта в базе данных, `invoice_url`, полный URL данного счёта, и `invoice`, где находится HTML-код предпросмотра e-mail сообщения, которое будет выслано клиенту:

```
{
  'invoice_id' : 20120229133742255,
  'invoice_url' : 'https://pay.example.com/bill/20120229133742255',
  'invoice': '<HTML><HEAD><META http-equiv=Content-Type ...'
}
```

Созданному счёту назначается служебный статус `created`. Счета с этим статусом не возвращаются в запросе истории счетов. Плательщику можно передать адрес счёта из `invoice_url`, по любому каналу (почта, сервисы сообщений и т.п.). При переходе по этому адресу плательщик попадёт на страницу оплаты данного счёта для ввода реквизитов карты.

3.5. Отправка счёта клиенту `/change/invoice/send/`

Данный запрос отправляет клиенту письмо, созданное предыдущим запросом.

Таблица 3.5.1. Параметры запроса отправки счёта

Тип	Пример запроса	
POST	<code>/change/invoice/send/</code>	
	Параметр	Назначение
1.	id	Идентификатор созданного счёта, письмо по которому необходимо отправить.
2.	token	Токен безопасности. (Подробнее в главе 6 данного руководства)

Результатом данного запроса будет объект

```
{  
  'result' : success  
}
```

в случае, если отправка произошла успешно, и объект ошибки, если письмо отправить не удалось.

4. Настройки системы

В этом разделе представлены дополнительные запросы, которые служат для получения информацией о пользователе и юридическом лице. Также, в этом разделе представлены запросы для получения и изменения настроек системы.

Запросы для работы с пользователем:

Таблица 4.1. JSON-методы работы со счетами

	URI	Назначение
4.1	/info/user/settings/	Получить настройки текущего пользователя (имя, возможность совершать возвраты)
4.2	/info/user/lastvisit/	Получить информацию о последнем посещении личного кабинета

Запрос получения настроек системы:

Таблица 4.2. JSON-методы работы со счетами

	URI	Назначение
4.3	/info/organization/settings/	Получить настройки системы.
4.4	/info/organization/notifications/	Получить сообщение информационного блока.
4.5	/info/organization/fields/	Получение списка полей, использующихся в форме оплаты, в платежах и в выставляемых счетах.
4.6	/info/organization/reportemails/	Получить список рассылки для уведомления о платежах.
4.7	/info/organization/users/	Получить список пользователей и их настройки
4.8	/info/organization/fiscal/	Получить настройки работы системы фискализации

4.1. Запрос настроек текущего пользователя /info/user/settings/

Запрос возвращает информацию об имени текущего пользователя и возможности совершать возвраты от его лица. Для получения данных необходимо выполнить GET-запрос со следующими параметрами.

Таблица 4.1.1. Параметры запроса

Тип	Формат запроса
GET	/info/user/settings/

Таблица 4.1.2. Параметры ответа на запрос

Тип	Формат ответа	
	Параметр	Назначение

1.	user	Имя пользователя
2.	refund_allowed	Возможность осуществлять возвраты. Принимает значения: true/false.
3.	refund	Количество возвратов, которое пользователь может осуществить за сутки.
4.	admin	Признак того, что пользователь является администратором. Принимает значения true/false.

Пример возвращаемого объекта:

```
{
  "user": "admin",
  "refund_allow": "true",
  "refund": "0",
  "admin": "true"
}
```

4.2. Запрос информации о последнем посещении пользователя **/info/user/lastvisit/**

Запрос возвращает информацию о последнем посещении пользователем личного кабинета PayKeeper. В случае если это первый визит пользователя, возвращает текущее время. Для получения данных необходимо выполнить GET-запрос со следующими параметрами.

Таблица 4.2.1. Параметры запроса

Тип	Формат запроса
GET	/info/user/lastvisit/

Таблица 4.2.2. Параметры ответа на запрос

Тип	Формат ответа	
	Параметр	Назначение
1.	ip	IP адрес последнего визита
2.	datetime	Дата и время последнего визита

Пример возвращаемого объекта:

```
{
  ip: "81.5.119.7",
  datetime: "2014-06-20 13:30:52"
}
```

4.3. Запрос настроек PayKeeper /info/organization/settings/

Запрос возвращает информацию о настройках PayKeeper. Пользователь, не являющийся администратором, получит только те свойства, которые отмечены в таблице флагом «все пользователи». Для получения данных необходимо выполнить следующий GET-запрос.

Таблица 4.3а. Параметры запроса

Тип	Формат запроса
GET	/info/organization/settings/

Таблица 4.3б. Параметры ответа на запрос

Тип	Формат ответа		
	Параметр	Назначение	Пользователи
1.	shopname	Название торговой точки	все
2.	shopdomain	URL-сайта торговой точки	все
3.	legal_name	Юридическое наименование организации	все
4.	paykeeper_id	Номер договора	все
5.	support_email	Почта для обращения в организацию	администратор
6.	informer_type	Режим работы информера, принимает значение post или email	администратор
7.	informer_url	URL уведомления для информера по которому отправляется информация о принятом платеже	администратор
8.	informer_seed	Секретное слово для подписи сообщений информера	администратор
9.	smtp_use_custom	Флаг использования стандартных настроек SMTP сервера PayKeeper. Если указано значение 'true', то для отправки писем используются настройки указанные в параметрах smtp_host, smtp_port, smtp_user, smtp_password. Если указано 'false', то используются настройки почтового сервера PayKeeper по умолчанию и параметры smtp_host, smtp_port, smtp_user, smtp_password не возвращаются в запросе.	администратор
10.	smtp_host	Адрес SMTP сервера.	администратор
11.	smtp_port	Порт SMTP сервера.	администратор
12.	smtp_user	Имя пользователя SMTP в формате адреса электронной почты.	администратор
13.	smtp_password	Пароль пользователя SMTP.	администратор
14.	skip_default_pages	Флаг указывает на то, что нужно пропускать стандартные страницы возврата paykeeper и сразу отправлять пользователя на сайт торговой точки.	администратор

		Принимает значения on или off	
15.	success_url	URL для возврата в случае успешной оплаты	администратор
16.	fail_url	URL для возврата в случае ошибки при оплате	администратор

Пример возвращаемого объекта:

```
{  
  shopname: "Магазин example.com",  
  shopdomain: "http://example.com",  
  legal_name: "ООО МАГАЗИН",  
  paykeeper_id: "140221-031-1",  
  support_email: "support@example.com"  
  informer_type: "email"  
  informer_url: "http://mysite.ru/postinform.php"  
  informer_seed: "secretpassword"  
  smtp_use_custom: "true"  
  smtp_user: "mymail@ya.ru"  
  smtp_password: "mypasswd"  
  smtp_host: "ssl://smtp.yandex.ru"  
  smtp_port: "465"  
  skip_default_pages: "true"  
  success_url: "http://example.com/success/"  
  fail_url: "http://example.com/fail/"  
}
```

Схема 4. Общие параметры

НАСТРОЙКИ

- Общие параметры
- Настройка формы оплаты
- Получение информации о платежах
- Доступ к панели администратора
- SMTP-сервер**
- Адреса перенаправления клиента

ОБЩИЕ ПАРАМЕТРЫ

Эти параметры задают информацию о Вашем предприятии, которая отображается Вашим клиентам на разных этапах оплаты.

Название компании	ООО «Пэйкипер»	legal_name readonly
Название торговой точки	Платформа PayKeeper	shopname readonly
URL Сайта	http://paykeeper.ru	shopdomain readonly
Номер договора	140221-031-1	paykeeper_id readonly
Электронная почта	support@paykeeper.ru	support_email

Отображается в сообщениях о результатах операции

5.1 или [Отменить изменения](#)

Схема 5. SMTP-сервер

НАСТРОЙКИ

- Общие параметры
- Настройка формы оплаты
- Получение информации о платежах
- Доступ к панели администратора
- SMTP-сервер**
- Адреса перенаправления клиента

SMTP-СЕРВЕР

Необходимую информацию Вам могут предоставить сетевые администраторы Вашего предприятия. Если Вы не уверены в необходимости использования отдельного SMTP-сервера, пожалуйста, пропустите этот пункт.

Использовать SMTP-сервер собственной организации smtp_use_custom

Имя SMTP-сервера	<input type="text" value="ssl://smtp.yandex.ru"/>	smtp_host
Порт SMTP-сервера	<input type="text" value="465"/>	smtp_port
Название почтового ящика	<input type="text" value="info@example.com"/>	smtp_user
Пароль к почтовому ящику	<input type="text" value="13gafGG"/>	smtp_password

4.3

5.1 или [Отменить изменения](#)

Схема 6. Адреса перенаправления клиента

НАСТРОЙКИ

Общие параметры

Настройка формы оплаты

Получение информации о платежах

Доступ к панели администратора

SMTP-сервер

Адреса перенаправления клиента

АДРЕСА ПЕРЕНАПРАВЛЕНИЯ КЛИЕНТА

После завершения операции оплаты пользователь возвращается на служебные страницы PayKeeper (<http://<ваш сервер PayKeeper>/fail/> и <http://<ваш сервер PayKeeper>/success/>). Они отображают сообщение о результате операции, и, если оплата не удалась, описание возможных причин неудачи.

С этой страницы пользователь переходит по ссылке обратно на сайт организации. По умолчанию, возврат происходит на главную страницу, но при помощи следующей таблицы Вы можете задать, по каким URL-адресам пользователь сможет перейти в случае успеха и в случае неудачи при оплате.

Обратите внимание, что эти URL нельзя изменять в зависимости от номера заказа, а переход по ним ни в коем случае нельзя считать завершением оплаты. Для того чтобы автоматически отмечать заказы как оплаченные, необходимо использовать POST-интеграцию и обрабатывать POST-сообщения об успешных платежах. Настроить POST-интеграцию можно в пункте [Получение информации о платежах](#).

Пропускать служебные страницы PayKeeper skip_default_pages 4.3

URL страницы, на которую клиент переходит при успешном завершении оплаты success_url [↗](#)

URL страницы, на которую клиент переходит при неудаче в процессе оплаты fail_url [↗](#)

5.1

или [Отменить изменения](#)

4.4. Запрос информационных сообщений для организации `/info/organization/notifications/`

Запрос возвращает информационные сообщения для отображения на странице. Для получения данных необходимо выполнить следующий GET-запрос.

Таблица 4.4.1 Параметры запроса

Тип	Формат запроса
GET	<code>/info/organization/notifications/</code>

В ответ возвращается массив объектов следующего вида:

Таблица 4.4.2 Параметры ответа на запрос

Тип	Формат ответа	
	Параметр	Назначение
1.	type	Тип сообщения. Может принимать значения important , bottom , bottom_important .
2.	message	HTML-код сообщения.
3.	link	Ссылка для отображения в области ссылок, если тип сообщения предусматривает ссылку (значение атрибута href).
4.	link_caption	Текст ссылки link.

Пример ответа на запрос:

```
[
  {
    type: "important",
    message: "<strong>ВНИМАНИЕ:</strong> ДО КОНЦА ОПЛАЧЕННОГО ПЕРИОДА ПОДДЕРЖКИ
И ОБНОВЛЕНИЙ <strong>ОСТАЛОСЬ 10 ДНЕЙ</strong>",
    link: "https://paykeeper.ru/paykeeper/pay/tech/",
    link_caption: "ПРОДЛИТЬ"
  },
  {
    type: "bottom",
    message: "до конца оплаченного периода обновлений <br /> и поддержки
<em>осталось 255 дней</em>",
    link: "https://paykeeper.ru/paykeeper/pay/tech/",
    link_caption: "Обратиться в службу поддержки"
  }
]
```

4.5. Запрос на получение списка используемых полей **/info/organization/fields/**

Запрос возвращает поля, которые используются в форме оплаты, в платежах и в выставляемых счетах. В запросе возвращается название поля в системе PayKeeper, название поля для отображения, значение поля по умолчанию, информация является ли поле активным и является ли поле обязательным для заполнения, а также тип поля и порядок сортировки в форме. Для получения данных необходимо выполнить следующий GET-запрос.

Таблица 4.5.1. Параметры запроса

Тип	Формат запроса
GET	/info/organization/fields/

В ответ возвращается массив объектов следующего вида:

Таблица 4.5.2. Параметры ответа на запрос

	Формат ответа	
	Параметр	Назначение
1.	pk_name	Название поле в PayKeeper
2.	displayed_name	Название поле для отображения пользователю
3.	required	Признак того, что поле является обязательным для заполнения. Принимает значения true,false.
4.	placeholder	Значение поля по умолчанию
5.	enabled	Признак того, что поле является активным и его нужно использовать. Принимает значения true,false.
6.	type	Тип поля. Может принимать значения: 'sum', 'text', 'phone', 'email', 'bool'.
7.	order	Порядок сортировки в форме. Может принимать целые значения от 0 до 10 000. Сортировка происходит по возрастанию этого значения.

Пример ответа на запрос:

```
[
  {
    "pk_name": "clientid",
    "displayed_name": "Фамилия Имя Отчество",
    "placeholder": "Иванов Иван Иванович",
    "required": "true",
    "enabled": "true",
    "required": "false",
    "order": "20",
    "type": "clientid"
  },
  ...
]
```

Схема 7. Настройка формы оплаты

НАСТРОЙКИ

Общие параметры

Настройка формы оплаты

Получение информации о платежах

Доступ к панели администратора

SMTP-сервер

Адреса перенаправления клиента

НАСТРОЙКА ФОРМЫ ОПЛАТЫ

Этот раздел следует заполнять, если Вы планируете редактировать внешний вид формы, поля или их названия.



Включить параметр «Номер заказа»

Название параметра:

Тип поля:

Значение по умолчанию:

Параметр обязателен для заполнения

Включить параметр «Фамилия Имя Отчество» displayed_name

enabled

Название параметра: displayed_name

Тип поля: type

Значение по умолчанию: placeholder

Параметр обязателен для заполнения required

Включить параметр «Электронный адрес»

Название параметра:

Тип поля:

Значение по умолчанию:

Параметр обязателен для заполнения

Включить параметр «Сумма»

Название параметра:

Значение по умолчанию:

Параметр обязателен для заполнения

4.5
5.2

или [Отменить изменения](#)

4.6. Запрос на получение списка рассылки для уведомления о платежах **/info/organization/reportemails/**

Запрос возвращает список адресов электронной почты, на которую осуществляется рассылка уведомлений о принятых платежах. Для получения данных необходимо выполнить следующий GET-запрос.

Таблица 4.6.1. Параметры запроса

Тип	Формат запроса
GET	/info/organization/reportemails/

В ответ возвращается массив объектов следующего вида:

Таблица 4.6.2. Параметры ответа на запрос

Тип	Формат ответа	
	Параметр	Назначение
1.	id	Идентификатор
2.	email	Адрес электронной почты

Пример ответа на запрос:

```
[  
  {  
    id: "1",  
    email: "support@paykeeper.ru"  
  },  
  ...  
]
```

Схема 8.1. Получение информации о платежах

НАСТРОЙКИ

- Общие параметры
- Настройка формы оплаты
- Получение информации о платежах**
- Доступ к панели администратора
- SMTP-сервер
- Адреса перенаправления клиента

ПОЛУЧЕНИЕ ИНФОРМАЦИИ О ПЛАТЕЖАХ

PayKeeper предоставляет возможность оповещать организацию о принятых платежах с помощью отправки писем на адрес электронной почты, либо POST запросом на указанный URL.

4.3 **Способ получения уведомления о платежах** **informer_type post/email**

4.6 **Электронные адреса для уведомлений**

- ✕
- ✕
- ✕

[Добавить ещё один электронный адрес](#)

5.3, 5.4 или [Отменить изменения](#)

Схема 9.1. Получение информации о платежах

НАСТРОЙКИ

- Общие параметры
- Настройка формы оплаты
- Получение информации о платежах**
- Доступ к панели администратора
- SMTP-сервер
- Адреса перенаправления клиента

ПОЛУЧЕНИЕ ИНФОРМАЦИИ О ПЛАТЕЖАХ

PayKeeper предоставляет возможность оповещать организацию о принятых платежах с помощью отправки писем на адрес электронной почты, либо POST запросом на указанный URL.

4.5 Способ получения уведомления о платежах **informer_type post/email**

URL, на который будут отправляться POST-запросы **informer_url**

Секретное слово **informer_seed**

Используйте латинские буквы в верхнем и нижнем регистре, цифры и знаки препинания

5.1 или [Отменить изменения](#)

4.7. Запрос на получение списка пользователей и их настроек `/info/organization/users/`

Запрос возвращает список пользователей PayKeeper вместе с их настройками. Для получения данных необходимо выполнить следующий GET-запрос.

Таблица 4.7.1. Параметры запроса

Тип	Формат запроса
GET	<code>/info/organization/users/</code>

В ответ возвращается массив объектов следующего вида:

Таблица 4.7.2. Параметры ответа на запрос

Тип	Формат ответа	
	Параметр	Назначение
1.	id	Идентификатор пользователя
2.	login	Логин пользователя
3.	refund	Количество возвратов которые пользователь может сделать за сутки. Если установлено «0», то функция возврата для этого пользователя недоступна.
4.	admin	Признак того, что пользователь является администратором. Принимает значения true/false.

Пример ответа на запрос:

```
[
  {
    id: 1,
    login: "admin",
    refund: "0",
    admin: "true"
  },
  ...
]
```

Схема 10. Доступ к панели администратора

НАСТРОЙКИ

Общие параметры

Настройка формы оплаты

Получение информации о платежах

Доступ к панели администратора

SMTP-сервер

Адреса перенаправления клиента

ДОСТУП К ПАНЕЛИ АДМИНИСТРАТОРА

Пользователь **admin**
✕

Имя пользователя (Login)
Латинские буквы в верхнем и нижнем регистре и цифры

Пароль При начале ввода появится окно подтверждение пароля
Латинские буквы в верхнем и нижнем регистре, цифры и знаки препинания и не менее 6 символов

Подтверждение пароля

Количество операций возврата в сутки ?

Администраторские полномочия

Пользователь **user**
✕

Имя пользователя (Login)
Латинские буквы в верхнем и нижнем регистре и цифры

Пароль Пароль замаскирован ***. Пароль можно только изменить, увидеть нельзя.
Латинские буквы в верхнем и нижнем регистре, цифры и знаки препинания и не менее 6 символов

Количество операций возврата в сутки ?

Администраторские полномочия

4.7

Добавить ещё одного пользователя

5.6, 5.7, 5.8
СОХРАНИТЬ
или [Отменить изменения](#)

4.8. Запрос на получение настроек системы фискализации /info/organization/fiscal/

Запрос вернёт настройки фискализации. Если фискализация выключена, объект настроек будет содержать только индикатор **enabled: false**. Если фискализация включена, будут отображены все параметры.

Таблица 4.88.1. Параметры запроса

Тип	Формат запроса
GET	/info/organization/users/

5. Модификация настроек системы

С помощью протокола PayKeeper можно менять следующие настройки системы:

- 1) Информация об организации
- 2) Настройки формы оплаты
- 3) Поведение системы после получения платежа
- 4) Настройки пользователей PayKeeper
- 5) Адреса перенаправлений

	URI	Назначение
5.1	/change/organization/setting/	Запрос на изменение параметра настроек PayKeeper
5.2	/change/organization/formfield/	Запрос на изменения поля формы
5.3	/change/organization/addreportemail/	Запрос на добавление адреса в список рассылки уведомлений о принятых платежах
5.4	/change/organization/deletereporremail/	Запрос на удаление адреса из списка рассылки уведомлений о принятых платежах
5.5	НЕ ИСПОЛЬЗУЕТСЯ	
5.6	/change/user/add/	Запрос на добавление пользователя
5.7	/change/user/update/	Запрос на изменение существующего пользователя
5.8	/change/user/delete/	Запрос на удаление пользователя

5.1. Запрос на изменение параметра /change/organization/setting/

Запрос модификации параметра PayKeeper. Для модификации нужно задать имя модифицируемого параметра (name) и его новое значение (value). Если новое значение не указано модификации не произойдет.

Список настроек, которые можно редактировать с помощью запроса:

Таблица 5.1.1. Параметры запроса

	Параметр	Назначение
5.1.1	support_email	Почта для отображения в организацию
5.1.2	informer_type	Режим работы информера, принимает значение post или email
5.1.3	informer_url	URL уведомления для информера по которому отправляется информация о принятом платеже
5.1.4	informer_seed	Секретное слово для подписи сообщений информера
5.1.5	smtp_use_custom	Флаг использования стандартных настроек SMTP сервера

		PayKeeper. Если указано значение 'true', то для отправки писем используются настройки указанные в параметрах smtp_host, smtp_port, smtp_user, smtp_password. Если указано 'false', то используются настройки почтового сервера PayKeeper по умолчанию и параметры smtp_host, smtp_port, smtp_user, smtp_password не возвращаются в запросе.
5.1.6	smtp_host	Адрес SMTP сервера.
5.1.7	smtp_port	Порт SMTP сервера.
5.1.8	smtp_user	Имя пользователя SMTP в формате адреса электронной почты.
5.1.9	smtp_password	Пароль пользователя SMTP.
5.1.10	skip_default_pages	Флаг указывает на то, что нужно пропускать стандартные страницы возврата paykeeper и сразу отправлять пользователя на сайт торговой точки. Принимает значения on или off
5.1.11	success_url	URL для возврата в случае успешной оплаты
5.1.12	fail_url	URL для возврата в случае ошибки при оплате

Для изменения параметра нужно выполнить POST запрос со следующими параметрами:

Таблица 5.1.2. Параметры запроса

Тип	Формат запроса	
POST	/change/organization/setting/	
	Параметр	Назначение
1.	name	Название параметра PayKeeper
2.	value	Новое значение настройки
3.	token	Токен безопасности. (Подробнее в главе 6 данного руководства)

Результатом данного запроса будет объект

```
{
  'result' : success
}
```

5.2. Запрос на изменение поля /change/organization/formfield/

Запрос модификации поля, которое используются в форме оплаты, в платежах и в выставяемых счетах. Запрос модифицирует одно из полей по имени поля (pk_name) и устанавливает новые значения для placeholder, displayed_name, required, placeholder, enabled, type, order. Если какое-либо из новых значений не указано, то старое изменено не будет. Поле sum изменить нельзя, в случае указания значения pk_name равным sum будет возвращена ошибка.

Таблица 5.2.1. Параметры запроса

Тип	Формат запроса	
POST	/change/organization/formfield/	
	Параметр	Назначение
1.	pk_name	Название поле в PayKeeper.
2.	displayed_name	Название поля для отображения пользователю.
3.	required	Признак того, что поле является обязательным для заполнения. Принимает значения true,false.
4.	placeholder	Значение поля по умолчанию.
5.	enabled	Признак того, что поле является активным и его нужно использовать. Принимает значения true,false.
6.	type	Тип поля. Может принимать значения: 'sum', 'clientid', 'orderid', 'text', 'phone', 'email', 'bool'.
7.	order	Порядок сортировки в форме. Может принимать целые значения от 0 до 10 000. Сортировка происходит по возрастанию этого значения.
8.	token	Токен безопасности. (Подробнее в главе 6 данного руководства)

Результатом данного запроса будет объект

```
{
  'result' : success
}
```

5.3. Запрос на добавление адреса в список рассылки уведомлений о принятых платежах /change/organization/addreportemail/

Запрос добавляет новый адрес для рассылки почтовых уведомлений о принятых платежах. Для добавления адреса нужно выполнить следующий POST-запрос:

Таблица 5.3.1. Параметры запроса

Тип	Формат запроса	
POST	/change/organization/addreportemail/	
	Параметр	Назначение
1.	email	Адрес электронной почты
2.	token	Токен безопасности. (Подробнее в главе 6 данного руководства)

Результатом данного запроса будет объект

```
{
  'result' : success
}
```

5.4. Запрос на удаление адреса из списка рассылки уведомлений о принятых платежах /change/organization/deletereportemail/

Запрос удаляет адрес из списка рассылки почтовых уведомлений. Для удаления адреса нужно выполнить следующий POST-запрос, передав идентификатор почтового адреса.

Таблица 5.4.1. Параметры запроса

Тип	Формат запроса	
POST	/change/organization/deletereportemail/	
	Параметр	Назначение
1.	id	Идентификатор адреса электронной почты
2.	token	Токен безопасности. (Подробнее в главе 6 данного руководства)

Результатом данного запроса будет объект

```
{
  'result' : success
}
```

5.5. НЕ ИСПОЛЬЗУЕТСЯ

5.6. Запрос на добавление пользователя /change/user/add/

Запрос добавляет нового пользователя в систему PayKeeper. Для добавления нужно выполнить следующий POST-запрос:

Таблица 5.6.1. Параметры запроса

Тип	Формат запроса	
POST	/change/user/add/	
	Параметр	Назначение

1.	login	Логин пользователя
2.	password	Пароль пользователя
3.	refund	Количество возвратов, которое пользователь может осуществить за сутки.
4.	admin	Признак того, что пользователь является администратором. Принимает значения true/false.
5.	token	Токен безопасности. (Подробнее в главе 6 данного руководства)

Результатом данного запроса будет объект

```
{
  'result' : success
}
```

5.7. Запрос на изменение существующего пользователя /change/user/update/

Запрос на изменение пользователя системы PayKeeper. Запрос корректирует настройки пользователя с заданными login. Если часть параметров не указано, то они будут пропущены. Для выполнения запроса нужно выполнить следующий POST-запрос:

Таблица 5.7.1. Параметры запроса

Тип	Формат запроса	
POST	/change/user/update/	
	Параметр	Назначение
1.	id	Идентификатор пользователя
2.	login	Логин пользователя
3.	password	Пароль пользователя
4.	refund	Количество возвратов, которое пользователь может осуществить за сутки.
5.	admin	Признак того, что пользователь является администратором. Принимает значения true/false.
6.	token	Токен безопасности. (Подробнее в главе 6 данного руководства)

Результатом данного запроса будет объект

```
{
  'result' : success
}
```

```
}
```

5.8. Запрос на удаление пользователя /change/user/delete/

Запрос на удаление пользователя из системы PayKeeper. Запрос удаляет пользователя с указанным id. Для удаления пользователя нужно выполнить следующий POST-запрос:

Таблица 5.8.1. Параметры запроса

Тип	Формат запроса	
POST	/change/user/delete/	
	Параметр	Назначение
1.	id	Идентификатор пользователя
2.	token	Токен безопасности. (Подробнее в главе 6 данного руководства)

Результатом данного запроса будет объект

```
{  
  'result' : success  
}
```

6. Токен безопасности

6.1. Получение токена безопасности `/info/settings/token/`

Для обеспечения безопасности и предотвращения XSS атак, любые POST запросы выполняются с передачей специального дополнительного параметра – токена. Чтобы получить токен безопасности нужно выполнить запрос следующего вида:

Таблица 6.1.

Тип	Формат запроса
GET	<code>/info/settings/token/</code>

Результатом данного запроса будет объект

```
{  
  'token' : 'fd50e47596cff6c63c7ea0a633c0d2a8'  
}
```

Полученный токен необходимо использовать как дополнительный параметр для каждого POST запроса. Смена токена происходит раз в 24 часа.

7. Экспорт данных

Предусмотрены следующие запросы для экспорта данных из Личного Кабинета:

Таблица 8.1. Методы экспорта данных

	URI	Назначение
8.1	/export/payments/	Экспорт истории платежей

7.1. Запрос экспорта истории платежей /export/payments/

Для загрузки файла с историей платежей необходимо выполнить GET-запрос со следующими параметрами:

Таблица 8.1.1. Параметры запроса экспорта истории платежей

Тип	Пример запроса	
GET	/export/payments/?type=xlsx&start=2014-04-14&end=2014-06-14&status[]=success&status[]=canceled&status[]=failed&payment_system_id[]=9&payment_system_id[]=1&from=0&limit=1000	
	Параметр	Назначение
1.	type	Желаемый тип файла для загрузки. Доступные значения: xlsx
2.	start	Дата начала периода в формате YYYY-MM-DD
3.	end	Дата конца периода в формате YYYY-MM-DD
4.	payment_system_id[]	Идентификатор платежной системы. Применяется для фильтрации по платежной системе. Обязательный параметр.
5.	status[]	Статус платежей. Применяется для фильтрации платежей по статусам. Может принимать значения: pending, obtained, canceled, success, failed, stuck, refunded, refunding, partially_refunded . Обязательный параметр.
6.	from	Пропускает первые N значений. Обязательный параметр.
7.	limit	Максимальное количество элементов в выборке. Не более 1 000.

В результате данного запроса веб-сервер сгенерирует файл указанного типа, содержащий платежи, удовлетворяющие заданным условиям.

8. Работа с чеками 54-ФЗ

PayKeeper поддерживает автоматическую генерацию чеков согласно 54-ФЗ.

Таблица 8.1. Методы для работы с чеками

	URI	Назначение
8.1	ЗАРЕЗЕРВИРОВАНО	Получение реестра чеков за период, с фильтром по типам.
8.2	ЗАРЕЗЕРВИРОВАНО	Получение количеств чеков за период, сгруппированных по типам
8.3	/info/receipts/bypaymentid/	Получение информации о чеках, выданных по указанному платежу
8.4	/info/receipts/byid/	Получение информации о чеке по его внутреннему номеру
8.5	ЗАРЕЗЕРВИРОВАНО	Поиск чеков по фискальным атрибутам
8.6	ЗАРЕЗЕРВИРОВАНО	ЗАРЕЗЕРВИРОВАНО
8.7	ЗАРЕЗЕРВИРОВАНО	ЗАРЕЗЕРВИРОВАНО
8.9	ЗАРЕЗЕРВИРОВАНО	ЗАРЕЗЕРВИРОВАНО
8.10	ЗАРЕЗЕРВИРОВАНО	ЗАРЕЗЕРВИРОВАНО
8.11	/change/receipt/print/	Сгенерировать чек по платежу или с произвольными атрибутами
8.12	ЗАРЕЗЕРВИРОВАНО	Отправить запрос о состоянии чеков кассовому устройству. Если у чека нет фискальных атрибутов, этот запрос может их получить.

8.1. В настоящее время не используется

8.2. В настоящее время не используется

8.3. [Получение чеков по платежу /info/receipts/bypaymentid/](#)

Запрос покажет чеки окончательного расчёта и произвольные чеки, сгенерированные по данному платежу. В настоящее время запрос не покажет самый первый чек, выданный при получении платежа.

Таблица 8.3.1. Параметры запроса

Тип	Формат запроса
GET	/info/receipts/bypaymentid/?payment_id=2211

Если по данному платежу не было чеков окончательного расчёта или на произвольные суммы, то, ответ будет пустым. Если чеки есть, они вернутся в виде массива структур, описанных в таблице 8.4.2

8.4. Получение чека по его идентификатору /info/receipts/byid/

С помощью этого запроса можно получить чек по его идентификатору:

Таблица 8.4.1. Параметры запроса

Тип	Формат запроса
GET	/info/receipts/byid/?id=4201

Если чек с таким номером не будет найден, ответ будет пустым. Если чек будет найден, то в ответ вернётся объект следующего вида:

Таблица 8.4.2. Параметры ответа на запрос

Тип	Формат ответа	
	Параметр	Назначение
1.	id	Идентификатор чека
2.	payment_id	Номер платежа, по которому был сгенерирован чек. Может быть пустым, если чек генерировался не по платежу
3.	type	Тип чека, sale, refund .
4.	is_post_sale	Был ли данный чек сгенерирован как чек окончательного расчёта
5.	refund_id	Порядковый номер возврата в рамках указанного платежа, если чек был сгенерирован по определённому возврату.
6.	status	Статус чека. Может быть created, request_sent, success, timeout, failed .
7.	contact	Почта или телефон плательщика
8.	sum_cashless	Сумма, оплаченная безналичными
9.	sum_cash	Сумма, оплаченная наличными
10.	sum_advance	Сумма ранее проведённых предоплат или внесённых авансов
11.	sum_credit	Сумма выданного кредита
12.	cart	Корзина товаров , таблица 12.1. Строка, кодирующая массив объектов в JSON-формате.
13.	receipt_properties	Дополнительные свойства чека , таблица 12.5. Строка, кодирующая объект в JSON-формате.
14.	fpd	Фискальная подпись документа
15.	fnd	Фискальный номер документа
16.	fn	Номер фискального накопителя
17.	ts	Момент времени формирования чека, строка в формате YYYYMMDDTHHmm
18.	rnkkt	Регистрационный номер ККТ

19.	shift_number	Номер смены
20.	receipt_number	Номер чека в смене
21.	obtain_datetime	Дата/Время проведения платежа
22.	request_datetime	Дата/Время информирования магазина о проведенном платеже
23.	fop_receipt_key	Ключ чека. Часть адреса постоянной страницы чека.
24.	fop_uuid	Уникальный идентификатор запроса к кассовому устройству
25.	fop_url	Строка содержимого QR-кода чека

Пример ответа на запрос:

```
{
  "id": "4201",
  "payment_id": "2098",
  "type": "sale",
  "is_post_sale": "1",
  "refund_id": null,
  "status": "success",
  "contact": "client@mail.ru",
  "sum_cashless": "0.00",
  "sum_cash": "0.00",
  "sum_advance": "15529.10",
  "sum_credit": "0.00",
  "cart": "[{"name": "TEST
3\\", "quantity": 1, "price": 15529.10, "sum": 15529.10, "tax": "none", "payment_
type": "full"}]",
  "receipt_properties": "null",
  "fpd": "2020473855",
  "fnd": "8930",
  "fn": "9999078900001341",
  "ts": "20190621T1638",
  "rnkkt": "0000000400054952",
  "shift_number": "7123",
  "receipt_number": "6162",
  "obtain_datetime": "2019-05-11 09:12:22",
  "request_datetime": "2019-05-11 09:12:01",
```

```
"fop_receipt_key": "VhamK7kl",
"fop_uuid": "6ee-cs-130",
"fop_url": "t=20190511T0912&s=15529.10&fn=9999078900001341&
i=8930&fp=2020473855&n=1"
}
```

8.5. В настоящее время не используется

8.6. В настоящее время не используется

8.7. В настоящее время не используется

8.8. В настоящее время не используется

8.9. В настоящее время не используется

8.10. В настоящее время не используется

8.11. Печать произвольного номера /change/receipt/print

Печать произвольного чека происходит в асинхронном режиме. Необходимо отправить данный запрос, а результат запроса проверить через некоторое время (рекомендуется не ранее, чем через 2 минуты).

Таблица 011.1. Параметры запроса

Тип	Формат запроса	
POST	/change/receipt/print/	
	Параметр	Назначение
1.	payment_id	Номер платежа, по которому нужно выдать чек. Если чек не должен быть привязан к конкретному платежу в таблице платежей, можно не передавать.
2.	type	Тип чека, строка sale или refund .
3.	is_post_sale	Пометить как чек окончательного расчёта, строка true или false
4.	refund_id	Порядковый номер возврата в рамках указанного платежа. Может быть пустым.
5.	contact	Почта или телефон плательщика
6.	sum_cashless	Сумма, оплаченная безналичными
7.	sum_cash	Сумма, оплаченная наличными
8.	sum_advance	Сумма ранее проведённых предоплат или внесённых авансов
9.	sum_credit	Сумма выданного кредита
10.	cart	Корзина товаров , таблица 12.1. Строка, кодирующая массив объектов в JSON-формате.
11.	receipt_properties	Дополнительные свойства чека , таблица 12.5. Строка, кодирующая объект в JSON-формате.
12.	receipt_key	Уникальный ключ чека, строка длиной не менее 6 символов. Будет

		отображаться как часть адреса постоянной страницы чека.
--	--	---

При генерации чеков, не связанных с платежами в PayKeeper, рекомендуется оставлять пустыми `payment_id` и `refund_id`, а для сопоставления чека с операцией в информационной системе предприятия заполнять `receipt_key` своим уникальным идентификатором. Ключ чека рекомендуется составлять из символов `a-z`, `A-Z`, `_`, `-`, `0-9` достаточно длинным, чтобы не была возможна атака перебором. Результатом данного запроса будет объект с идентификатором чека:

```
{  
  'receipt_id' : 816  
}
```

Система не примет запрос, если с данными `payment_id`, `refund_id`, `is_post_sale`, `type`, `receipt_key` уже есть успешно сгенерированный или находящийся в процессе генерации чек, или чек, по которому наступил таймаут генерации. В последнем случае ранее отосланный запрос может оказаться успешным, и статус чека будет загружен позднее. Если требуется отправить чек, сохранив связь с платежом или возвратом, но с такими данными чек уже ранее был сгенерирован, нужно сгенерировать для него новый ключ чека.

8.12. В настоящее время не используется

Приложение 1. Диаграмма состояний платежа.

Статусы платежей изменяются в соответствии с диаграммой, приведённой ниже. Статусы «получен» (*obtained*), «превышено число повторов» (*stuck*), «совершён» (*success*) эквивалентны с точки зрения поступления средств на расчётный счёт предприятия. Все они означают, что у держателя была списана сумма транзакции, и она либо уже зачислилась, либо в ближайшее время зачислится на расчётный счёт предприятия.

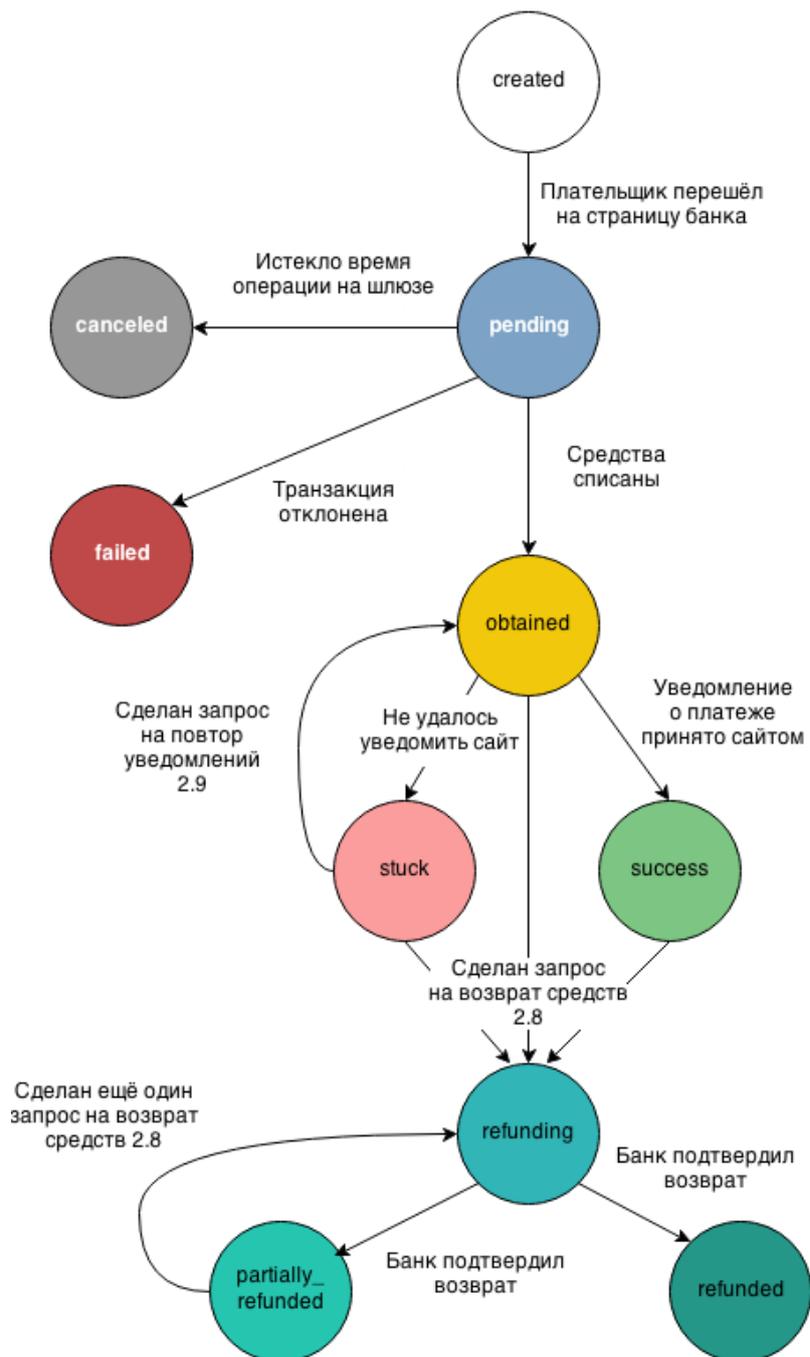


Схема П1.1. Диаграмма состояний платежа.

Приложение 2. Список изменений.

Таблица П2.1.

Дата	Методы	Изменение
2015-03-01		Версия 2.5.2.
2015-03-12	4.4	Расширен объект информационного сообщения. Введены типы сообщений, предусмотрены поля для хранения ссылок.
2015-03-16	8.1	Добавлен запрос экспорта платежей, поддерживающий новый формат фильтров.
2016-04-01	2.11	Добавлен запрос поиска платежей по параметрам заказа, плательщика или карточной транзакции
2016-09-09	2.12	Добавлен запрос для списания средств по авторизованной операции
2017-07-14	2.10	В ответ на запрос списания добавлено дополнительное поле, позволяющее различить синхронное и асинхронное списание.
2017-10-06	2.8	Добавлен параметр для передачи перечня возвращаемых товаров
2018-01-12	2.8	Добавлена рекомендация по автоматическому контролю результатов возвратов
2018-05-15	3.4	Добавлено поле invoice_url , содержащее полный адрес выставленного счёта.
2019-04-05	2.13	Добавлен метод для простой генерации чека окончательного расчёта по платежам предоплаты
2019-05-27	8.*	Добавлено расширенное API для работы с чеками.
2019-06-25	4.8	Добавлен метод для получения настроек фискализации