



**ЦЕНТРАЛЬНАЯ
КАССА**
ВСЕ ПЛАТЕЖИ ВОВРЕМЯ

Адрес

614087, Россия, г. Пермь,
ул. Малкова 12



8 (342) 240-40-22

info@ckassa.ru

ckassa.ru

УТВЕРЖДЕНЫ Приказом
Общества от 01.07.2019 №18
Версия 3.1

ТЕХНИЧЕСКИЕ ПАРАМЕТРЫ ПРЕДОСТАВЛЕНИЯ УСЛУГ ПО ПРИЕМУ ПЛАТЕЖЕЙ



Оглавление

1.	РЕКВИЗИТНЫЕ ГРУППЫ ДЛЯ ПОСТАВЩИКОВ УСЛУГ	3
1.1.	РЕКВИЗИТНЫЕ ГРУППЫ ДЛЯ ПОСТАВЩИКОВ УСЛУГ ЖКХ.....	3
1.2.	РЕКВИЗИТНЫЕ ГРУППЫ ДЛЯ ПОСТАВЩИКОВ УСЛУГ ОПЕРАТОРОВ СВЯЗИ	7
1.3.	РЕКВИЗИТНЫЕ ГРУППЫ ДЛЯ ИНТЕРНЕТ-МАГАЗИНОВ И ПРОЧЕЙ ОПЛАТЫ ТОВАРОВ И УСЛУГ.....	11
1.3.	РЕКВИЗИТНЫЕ ГРУППЫ ДЛЯ ДИСПЕТЧЕРСКИХ СЛУЖБ И ТАКСОПАРКОВ.	14
1.4.	РЕКВИЗИТНЫЕ ГРУППЫ ДЛЯ ОПЛАТЫ ПОЕЗДКИ ПАССАЖИРОМ	16
1.5.	РЕКВИЗИТНЫЕ ГРУППЫ ДЛЯ ИНТЕРНЕТ-МАГАЗИНОВ И ПРОЧЕЙ ОПЛАТЫ ТОВАРОВ И УСЛУГ.....	17
2.	РЕЕСТР ПРИНЯТЫХ ПЛАТЕЖЕЙ.....	20
3.	ТИПЫ ШАБЛОНОВ РЕЕСТРОВ	21
3.2.	Шаблон №1	21
3.3.	Шаблон №2.....	23
3.4.	Шаблон №3.....	25
3.5.	Шаблон №4.....	27
3.6.	Шаблон №5	29
4.	ОБРАБОТКА ДВУХМЕРНОГО ШТРИХ КОДА	30
5.	ТИПОВЫЕ СПЕЦИФИКАЦИИ ОНЛАЙН-ВЗАИМОДЕЙСТВИЯ.....	32
5.2.	Спецификация №1.....	32
5.3.	Спецификация №2.....	44
5.4.	Спецификация №3.....	48
6.	РАЗМЕЩЕНИЕ ПЛАТЕЖНОГО СЕРВИСА В ПРИЛОЖЕНИИ.....	53
6.1.	Типы подключения	53
6.2.	Технические параметры размещения платежного сервиса в Приложении	55



1. РЕКВИЗИТНЫЕ ГРУППЫ ДЛЯ ПОСТАВЩИКОВ УСЛУГ

- ✓ Реквизитная группа определяется выбранным Тарифом в рамках Договора.
- ✓ Количество символов в любом реквизите каждой реквизитной группы может быть откорректировано.
- ✓ К использованию принимаются все символы, за исключением символов, перечисленных в Таблице 1:

Таблица 1

Символы, не допустимые для использования	Символы, не рекомендуемые для использования
;	` ~ ^ & { } / \

1.1. РЕКВИЗИТНЫЕ ГРУППЫ ДЛЯ ПОСТАВЩИКОВ УСЛУГ ЖКХ

1.1.1. ЖКХ Базовый 1

№ п/п	Код	Наименование	Мин. длина	Макс. длина	Допустимые символы для ввода	Флаги
1	Л_СЧЕТ	Лицевой счет	1	20	Цифры, символы	видимый, редактируемый, обязательный, ключевой
2	ФИО	ФИО	1	60	Буквы	видимый, редактируемый, обязательный
3	АДРЕС	Адрес	1	80	Цифры, буквы, символы	видимый, редактируемый, обязательный
4	БАЛАНС	Баланс	1	10	Цифры, символы	Видимый, отображается после заполнения ключевого реквизита
5	ДОП_ИНФ	Дополнительная информация	0	60	Цифры, буквы, символы	видимый, редактируемый

Отображение для Плательщика при вводе данных:

Лицевой счет

ФИО

Адрес

Дополнительные параметры ↑

Дополнительная информация



1.1.2. ЖКХ Базовый 2

№ п/п	Код	Наименование	Мин. длина	Макс. длина	Допустимые символы для ввода	Флаги
1	Л_СЧЕТ	Лицевой счет	1	20	Цифры, символы	видимый, редактируемый, обязательный, ключевой
2	ФИО	ФИО	1	60	Буквы	видимый, редактируемый, обязательный
3	АДРЕС	Адрес	1	80	Цифры, буквы, символы	видимый, редактируемый, обязательный
4	МЕСЯЦ	Месяц	6	7	Цифры, буквы, символы	видимый, редактируемый, обязательный
5	БАЛАНС	Баланс	1	10	Цифры, символы	Видимый, отображается после заполнения ключевого реквизита
6	ДОП_ИНФ	Дополнительная информация	0	60	Цифры, буквы, символы	видимый, редактируемый

Отображение для Плательщика при вводе данных:

Лицевой счет

ФИО

Адрес

Месяц

ММ.ГГГГ

Дополнительные параметры ↑

Дополнительная информация



1.1.3. ЖКХ Оптимальный 1

Включает в себя реквизитную группу Тарифа ЖКХ Базовый

№ п/п	Код	Наименование	Мин. длина	Макс. длина	Допустимые символы для ввода	Флаги
1	Л_СЧЕТ	Лицевой счет	1	20	Цифры, символы	видимый, редактируемый, обязательный, ключевой
2	ФИО	ФИО	1	60	Буквы	видимый, редактируемый, обязательный
3	АДРЕС	Адрес	1	80	Цифры, буквы, символы	видимый, редактируемый, обязательный
4	МЕСЯЦ	Месяц	6	7	Цифры, буквы, символы	видимый, редактируемый, обязательный
5	ЗАДОЛЖЕННОСТЬ	Задолженность	0	12	Цифры, символы	видимый, редактируемый
6	ПЕНИ	Пени	0	12	Цифры, символы	видимый, редактируемый
7	БАЛАНС	Баланс	1	10	Цифры, символы	Видимый, отображается после заполнения ключевого реквизита
8	ДОП_ИНФ	Дополнительная информация	0	60	Цифры, буквы, символы	видимый, редактируемый

Отображение для Плательщика при вводе данных:

Лицевой счет

ФИО

Адрес

Месяц

ММ.ГГГГ

Дополнительные параметры ↑

Задолженность

Пени

Дополнительная информация



1.1.4. ЖКХ Оптимальный 2

№ п/п	Код	Наименование	Мин. длина	Макс длина	Тип	Флаги
1	Л_СЧЕТ	Лицевой счет	1	20	Цифры, буквы, символы	видимый, редактируемый, обязательный, ключевой
2	ФИО	ФИО	1	60	Буквы	видимый, редактируемый, обязательный
3	АДРЕС	Адрес	1	80	Цифры, буквы, символы	видимый, редактируемый, обязательный
4	МЕСЯЦ	Месяц	6	7	Цифры, буквы, символы	видимый, редактируемый, обязательный
5	ЗАДОЛЖЕННОСТЬ	Задолженность	0	12	Цифры, , символы	видимый, редактируемый
6	ПЕНИ	Пени	0	12	Цифры, символы	видимый, редактируемый
7	ГВС	Счетчик ГВС	1	15	Цифры, символы	видимый, редактируемый
8	ХВС	Счетчик ХВС	1	15	Цифры, символы	видимый, редактируемый
9	БАЛАНС	Баланс	1	10	Цифры, символы	Видимый, отображается после заполнения ключевого реквизита
10	ДОП_ИНФ	Дополнительная информация	0	60	Цифры, буквы, символы	видимый, редактируемый

Отображение для Плательщика при вводе данных:

Лицевой счет

ФИО

Адрес

Месяц

[Дополнительные параметры ↑](#)

Задолженность

Пени

Счетчик ГВС

Счетчик ХВС

Дополнительная информация



1.1.5. ЖКХ Супер

Может включать на выбор одну из реквизитных групп (ЖКХ Базовый 1, ЖКХ Базовый 2, ЖКХ Оптимальный 1, ЖКХ Оптимальный 2) с добавлением новых реквизитов и редактированием существующих реквизитов

1.2. РЕКВИЗИТНЫЕ ГРУППЫ ДЛЯ ПОСТАВЩИКОВ УСЛУГ ОПЕРАТОРОВ СВЯЗИ

1.2.1. Связь 1

№ п/п	Код	Наименование	Мин. длина	Макс. длина	Тип	Флаги
1	Л_СЧЕТ	Лицевой счет	2	15	Цифры, буквы, символы	видимый, редактируемый, обязательный, ключевой
2	БАЛАНС	Баланс	1	10	Цифры, символы	Видимый, отображается после заполнения ключевого реквизита

Отображение для Плательщика при вводе данных:

Лицевой счет

1.2.2. Связь 2

№ п/п	Код	Наименование	Мин. длина	Макс. длина	Тип	Флаги
1	Л_СЧЕТ	Лицевой счет	2	15	Цифры, буквы, символы	Текст: видимый, редактируемый, обязательный, ключевой
2	БАЛАНС	Баланс	1	10	Цифры, символы	Видимый, отображается после заполнения ключевого реквизита
3	ДОП_ИНФ	Дополнительная информация	0	60	Цифры, буквы, символы	видимый, редактируемый

Отображение для Плательщика при вводе данных:

Лицевой счет

[Дополнительные параметры ↑](#)

Дополнительная информация



1.2.3. Связь 3

№ п/п	Код	Наименование	Мин. длина	Макс. длина	Тип	Флаги
1	Л_СЧЕТ	Лицевой счет	2	15	Цифры, буквы, символы	видимый, редактируемый, обязательный, ключевой
2	БАЛАНС	Баланс	1	10	Цифры, символы	Видимый, отображается после заполнения ключевого реквизита
3	ДОП_ИНФ	Дополнительная информация	0	60	Цифры, буквы, символы	видимый, редактируемый

Отображение для Плательщика при вводе данных:

Лицевой счет

[Дополнительные параметры ↑](#)

Дополнительная информация

1.2.4. Связь 4

№ п/п	Код	Наименование	Мин. длина	Макс. длина	Тип	Флаги
1	Л_СЧЕТ	Лицевой счет	2	15	Цифры, буквы, символы	видимый, редактируемый, обязательный, ключевой
2	ФИО	ФИО	1	60	Буквы	видимый, редактируемый, обязательный
3	БАЛАНС	Баланс	1	10	Цифры, символы	Видимый, отображается после заполнения ключевого реквизита
4	ДОП_ИНФ	Дополнительная информация	0	60	Цифры, буквы, символы	видимый, редактируемый

Отображение для Плательщика при вводе данных:

Лицевой счет

ФИО

[Дополнительные параметры ↑](#)

Дополнительная информация

1.2.5. Связь 5

№ п/п	Код	Наименование	Мин. длина	Макс. длина	Тип	Флаги
-------	-----	--------------	------------	-------------	-----	-------



1	Л_СЧЕТ	Лицевой счет	2	15	Цифры, буквы, символы	видимый, редактируемый, обязательный, ключевой
2	ФИО	ФИО	1	60	Буквы	видимый, редактируемый, обязательный
3	АДРЕС	Адрес	0	30	Цифры, буквы, символы	видимый, редактируемый,
4	БАЛАНС	Баланс	1	10	Цифры, символы	Видимый, отображается после заполнения ключевого реквизита
5	ДОП_ИНФ	Дополнительная информация	0	60	Цифры, буквы, символы	видимый, редактируемый

Отображение для Плательщика при вводе данных:

Лицевой счет

ФИО

Дополнительные параметры ↑

Адрес

Дополнительная информация

1.2.6. Связь 6

№ п/п	Код	Наименование	Мин. длина	Макс. длина	Тип	Флаги
1	НОМЕР_ДОГОВОРА	Номер договора	2	15	Цифры, буквы, символы	видимый, редактируемый, обязательный, ключевой
2	ФИО	ФИО	1	60	Буквы	видимый, редактируемый, обязательный
3	БАЛАНС	Баланс	1	10	Цифры, символы	Видимый, отображается после заполнения ключевого реквизита
4	ДОП_ИНФ	Дополнительная информация	0	60	Цифры, буквы, символы	видимый, редактируемый

Отображение для Плательщика при вводе данных:

Номер договора

ФИО

Дополнительные параметры ↑

Дополнительная информация

1.2.7. Связь 7

№ п/п	Код	Наименование	Мин. длина	Макс. длина	Тип	Флаги
-------	-----	--------------	------------	-------------	-----	-------





1	НОМЕР_ДОГОВОРА	Номер договора	2	15	Цифры, буквы, символы	Текст: видимый, редактируемый, обязательный, ключевой
2	БАЛАНС	Баланс	1	10	Цифры, символы	Видимый, отображается после заполнения ключевого реквизита
3	ДОП_ИНФ	Дополнительная информация	0	60	Цифры, буквы, символы	видимый, редактируемый

Отображение для Плательщика при вводе данных

Номер договора

[Дополнительные параметры](#) ↑

Дополнительная информация



1.3. РЕКВИЗИТНЫЕ ГРУППЫ ДЛЯ ИНТЕРНЕТ-МАГАЗИНОВ И ПРОЧЕЙ ОПЛАТЫ ТОВАРОВ И УСЛУГ

1.2.1. Оплата услуг 1

№ п/п	Код	Наименование	Мин. длина	Макс. длина	Тип	Флаги
1	ФИО	ФИО	1	60	Цифры, буквы, символы	видимый, редактируемый, обязательный, ключевой
2	ДОП_ИНФ	Дополнительная информация	1	80	Цифры, буквы, символы	видимый, редактируемый, обязательный

Отображение для Плательщика при вводе данных:

ФИО

Дополнительная информация

1.2.2. Оплата услуг 2

№ п/п	Код	Наименование	Мин. длина	Макс. длина	Тип	Флаги
1	ФИО	ФИО	2	80	Буквы	видимый, редактируемый, обязательный, ключевой
2	НАЗВАНИЕ_МЕРОПРИЯТИЯ	Название мероприятия	1	80		видимый, редактируемый, обязательный
3	Дата	Дополнительная информация	10	10	Цифры, символы	видимый, редактируемый, обязательный

Отображение для Плательщика при вводе данных:

ФИО

Название мероприятия

Дата

1.2.3. Оплата товара 1

№	Код	Наименование	Мин.	Макс.	Тип	Флаги
---	-----	--------------	------	-------	-----	-------



п/п			длина	длина		
1	НОМЕР_ЗАКАЗА	Номер заказа	1	15	Цифры, буквы, символы	видимый, редактируемый, обязательный, ключевой
2	ФИО	ФИО	1	60	Цифры, буквы, символы	видимый, редактируемый, обязательный,
3	E-MAIL	e-mail	1	60	Цифры, буквы, символы	Видимый, не редактируемый. Отображается после заполнения ключевого реквизита
4	ТЕЛЕФОН	e-mail	1	60	Цифры, буквы, символы	Видимый, не редактируемый. Отображается после заполнения ключевого реквизита
5	ГОРОД	Город	1	40	Цифры, буквы, символы	Видимый, не редактируемый. Отображается после заполнения ключевого реквизита

Отображение для Плательщика при вводе данных:

Номер заказа

ФИО

1.2.4. Оплата товара 2

№ п/п	Код	Наименование	Мин. длина	Макс. длина	Тип	Флаги
1	НОМЕР_ЗАКАЗА	Номер заказа	1	50	Цифры, буквы, символы	видимый, редактируемый, обязательный, ключевой
2	НАИМЕНОВАНИЕ ТОВАРА	Наименование товара	1	100	Цифры, буквы, символы	видимый, редактируемый, обязательный,
3	АРТИКУЛ	Артикул	1	100	Цифры, буквы, символы	видимый, редактируемый, обязательный,
4	КОЛИЧЕСТВО	Количество	1	2	Цифры,	видимый, редактируемый, обязательный,

Отображение для Плательщика при вводе данных:

Номер заказа

Наименование товара

Артикул

Количество



1.2.5. Оплата заказа 1

№ п/п	Код	Наименование	Мин. длина	Макс. длина	Тип	Флаги
1	НОМЕР_ЗАКАЗА	Номер заказа	1	20	Цифры, буквы, символы	видимый, редактируемый, обязательный, ключевой
2	ФИО	ФИО плательщика	1	80	Цифры, буквы, символы	видимый, редактируемый, обязательный,

Отображение для Плательщика при вводе данных:

НОМЕР ЗАКАЗА

ФИО плательщика

1.2.6. Оплата заказа 2

№ п/п	Код	Наименование	Мин. длина	Макс. длина	Тип	Флаги
1	НОМЕР_ЗАКАЗА	Номер заказа	1	15	Цифры, буквы, символы	видимый, редактируемый, обязательный, ключевой
2	ФИО	ФИО	1	60	Цифры, буквы, символы	видимый, редактируемый, обязательный,
3	E-MAIL	e-mail	1	60	Цифры, буквы, символы	Видимый
4	ТЕЛЕФОН	e-mail	1	60	Цифры, буквы, символы	Видимый

Отображение для Плательщика при вводе данных:

Номер заказа

ФИО

Телефон

email



1.3. РЕКВИЗИТНЫЕ ГРУППЫ ДЛЯ ДИСПЕТЧЕРСКИХ СЛУЖБ И ТАКСОПАРКОВ

1.3.1. ТАКСИ_1 (Диспетчерские услуги)

№ п/п	Код	Наименование	Мин. длина	Макс. длина	Тип	Флаги
1	ПОЗЫВНОЙ	ПОЗЫВНОЙ	1	9	Целое число	видимый, редактируемый, обязательный, ключевой
2	БАЛАНС	Баланс	1	10	Цифры, символы	Видимый, отображается после заполнения ключевого реквизита

Отображение для Плательщика при вводе данных:

ПОЗЫВНОЙ

1.3.2. ТАКСИ_2 (Диспетчерские услуги)

№ п/п	Код	Наименование	Мин. длина	Макс. длина	Тип	Флаги
1	НОМЕР_ТЕЛЕФОНА	НОМЕР ТЕЛЕФОНА	11	11	Целое число	видимый, редактируемый, обязательный, ключевой
2	БАЛАНС	Баланс	1	10	Цифры, символы	Видимый, отображается после заполнения ключевого реквизита

Отображение для Плательщика при вводе данных:

Номер телефона

1.3.3. ТАКСИ_3 (Диспетчерские услуги)

№ п/п	Код	Наименование	Мин. длина	Макс. длина	Тип	Флаги
1	ПОЗЫВНОЙ	ПОЗЫВНОЙ	1	9	Целое число	видимый, редактируемый, обязательный, ключевой
2	БАЛАНС	Баланс	1	10	Цифры, символы	Видимый, отображается после заполнения ключевого реквизита
3	НОМЕР_ТЕЛЕФОНА	НОМЕР ТЕЛЕФОНА	11	11	Целое число	видимый, редактируемый, обязательный

Отображение для Плательщика при вводе данных:

ПОЗЫВНОЙ

ТЕЛЕФОН

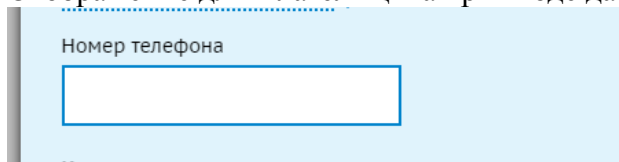


1.3.4. ТАКСИ_7 (Диспетчерские услуги)

Данная реквизитная группа подходит для диспетчерских служб «ВЕЗЕТ»

№ п/п	Код	Наименование	Мин. длина	Макс. длина	Тип	Флаги
1	ЛИЦЕВОЙ_СЧЕТ	НОМЕР ТЕЛЕФОНА	1	10	Целое число	видимый, редактируемый, обязательный, ключевой
2	БАЛАНС	Баланс	1	10	Цифры, символы	Видимый, отображается после заполнения ключевого реквизита

Отображение для Плательщика при вводе данных:



1.3.5. ТАКСИ_8

Данная реквизитная группа подходит при безналичной оплате поездки пассажиром для диспетчерских служб и таксопарков

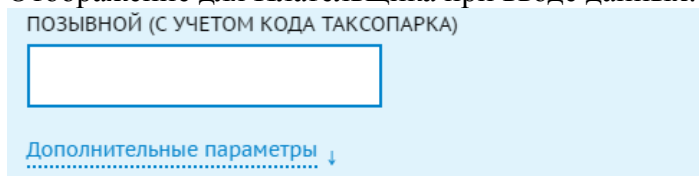
№ п/п	Код	Наименование	Мин. длина	Макс. длина	Тип	Флаги
1	НОМЕР_КОШЕЛЬКА	ID Получателя	8	10	Целое число	видимый, редактируемый, обязательный, ключевой
2	НОМЕР_ТЕЛЕФОНА	ТЕЛЕФОН ОТПРАВИТЕЛЯ	10	10	Целое число	видимый, редактируемый, обязательный
3	БАЛАНС	Баланс	1	10	Цифры, символы	Видимый, отображается после заполнения ключевого реквизита
4	ДОП_ИНФ	КОММЕНТАРИЙ	0	250	Строка	видимый, редактируемый

1.3.6. ТАКСИ_9 (Диспетчерские услуги)

Данная реквизитная группа подходит для диспетчерских служб, использующих ПО «ТАКСИ МАСТЕР»

№ п/п	Код	Наименование	Мин. длина	Макс. длина	Тип	Флаги
1	ПОЗЫВНОЙ	ПОЗЫВНОЙ (С УЧЕТОМ КОДА ТАКСОПАРКА)	9	9	Целое число	видимый, редактируемый, обязательный, ключевой
2	БАЛАНС	БАЛАНС	0	10	Сумма	Видимый, отображается после заполнения ключевого реквизита

Отображение для Плательщика при вводе данных:




1.4. РЕКВИЗИТНЫЕ ГРУППЫ ДЛЯ ОПЛАТЫ ПОЕЗДКИ ПАССАЖИРОМ

1.4.1. Поездка на Кошелек

№ п/п	Код	Наименование	Мин. длина	1.4.2. Макс. длина	Тип	Флаги
1	N_Кош	ID Получателя	8	10	Цифры, символы	видимый, редактируемый, обязательный, ключевой
2	N_Тел	ТЕЛЕФОН ОТПРАВИТЕЛЯ	10	10	Цифры, символы	видимый, редактируемый, обязательный
	ДОП_ИНФ	КОММЕНТАРИЙ	1	250	Цифры, буквы, символы	видимый, редактируемый

Отображение для Плательщика при вводе данных:

НОМЕР ЗАКАЗА

ФИО плательщика

1.4.3. Поездка на расчетный счет

№ п/п	Код	Наименование	Мин. длина	Макс. длина	Тип	Флаги
1	НОМЕР_ЗАКАЗА	Номер заказа	1	100	Цифры, символы	видимый, редактируемый, обязательный, ключевой
2	ПОЗЫВНОЙ	Позывной	1	100	Цифры, символы	видимый, редактируемый, обязательный
	ОПИСАНИЕ	Описание	0	200	Цифры, буквы, символы	видимый, редактируемый

Отображение для Плательщика при вводе данных:

Номер заказа

Позывной

Дополнительные параметры ↑
 Описание



1.5. РЕКВИЗИТНЫЕ ГРУППЫ ДЛЯ ИНТЕРНЕТ-МАГАЗИНОВ И ПРОЧЕЙ ОПЛАТЫ ТОВАРОВ И УСЛУГ

1.5.1. Оплата услуг_1

№ п/п	Код	Наименование	Мин. длина	Макс. длина	Тип	Флаги
1	ФИО	ФИО	1	60	Цифры, буквы, символы	видимый, редактируемый, обязательный, ключевой
2	ДОП_ИНФ	Дополнительная информация	1	80	Цифры, буквы, символы	видимый, редактируемый, обязательный

Отображение для Плательщика при вводе данных:

ФИО

Дополнительная информация

1.5.2. Оплата услуг_2

№ п/п	Код	Наименование	Мин. длина	Макс. длина	Тип	Флаги
1	ФИО	ФИО	2	80	Буквы	видимый, редактируемый, обязательный, ключевой
2	НАЗВАНИЕ_МЕРОПРИЯТИЯ	Название мероприятия	1	80		видимый, редактируемый, обязательный
3	Дата	Дополнительная информация	10	10	Цифры, символы	видимый, редактируемый, обязательный

Отображение для Плательщика при вводе данных:

ФИО

Название мероприятия

Дата

ДД.ММ.ГГГГ



1.5.3. Оплата товара 1

№ п/п	Код	Наименование	Мин. длина	Макс. длина	Тип	Флаги
1	НОМЕР_ЗАКАЗА	Номер заказа	1	15	Цифры, буквы, символы	видимый, редактируемый, обязательный, ключевой
2	ФИО	ФИО	1	60	Цифры, буквы, символы	видимый, редактируемый, обязательный,
3	E-MAIL	e-mail	1	60	Цифры, буквы, символы	Видимый
4	ТЕЛЕФОН	e-mail	1	60	Цифры, буквы, символы	Видимый
5	ГОРОД	Город	1	40	Цифры, буквы, символы	Видимый

1.5.4. Оплата товара 2

№ п/п	Код	Наименование	Мин. длина	Макс. длина	Тип	Флаги
1	НОМЕР_ЗАКАЗА	Номер заказа	1	50	Цифры, буквы, символы	видимый, редактируемый, обязательный, ключевой
2	НАИМЕНОВАНИЕ ТОВАРА	Наименование товара	1	100	Цифры, буквы, символы	видимый, редактируемый, обязательный,
3	АРТИКУЛ	Артикул	1	100	Цифры, буквы, символы	видимый, редактируемый, обязательный,
4	КОЛИЧЕСТВО	Количество	1	2	Цифры,	видимый, редактируемый, обязательный,

Отображение для Плательщика при вводе данных:

Номер заказа

Наименование товара

Артикул

Количество



1.5.5. Оплата заказа 1

№ п/п	Код	Наименование	Мин. длина	Макс. длина	Тип	Флаги
1	НОМЕР_ЗАКАЗА	Номер заказа	1	20	Цифры, буквы, символы	видимый, редактируемый, обязательный, ключевой
2	ФИО	ФИО плательщика	1	80	Цифры, буквы, символы	видимый, редактируемый, обязательный,

Отображение для Плательщика при вводе данных:

НОМЕР ЗАКАЗА

ФИО плательщика

1.5.6. Оплата заказа 2

№ п/п	Код	Наименование	Мин. длина	Макс. длина	Тип	Флаги
1	НОМЕР_ЗАКАЗА	Номер заказа	1	15	Цифры, буквы, символы	видимый, редактируемый, обязательный, ключевой
2	ФИО	ФИО	1	60	Цифры, буквы, символы	видимый, редактируемый, обязательный,
3	E-MAIL	e-mail	1	60	Цифры, буквы, символы	Видимый
4	ТЕЛЕФОН	e-mail	1	60	Цифры, буквы, символы	Видимый

Отображение для Плательщика при вводе данных:



Номер заказа	
ФИО	
Телефон	
email	

2. РЕЕСТР ПРИНЯТЫХ ПЛАТЕЖЕЙ

Реестр принятых платежей - сводная информация по принятым платежам, передаваемая Обществом Поставщику.

В зависимости от формата, реестр подразделяется на:

А) Реестр финансовый – Реестр принятых платежей, содержащий в себе информацию о платежах, перечисленных Обществом Поставщику, отправляемый в день отправки платежного поручения Поставщику согласно условиям Договора. Реестр учитывает платежи, удаленные по согласованию сторон. Информация о принятых платежах отображается в ЛК по факту приема платежа.

Б) Реестр информационный – Реестр принятых платежей, содержащий в себе информацию о платежах, принятых Обществом в пользу Поставщика за предыдущий день, отправляемый Поставщику согласно условиям Договора.

В информационный реестр платежей включаются все платежи, принятые Обществом для передачи Поставщику за прошедшие сутки.

Реестр высылается Обществом до 12:00 следующих суток, по часовому поясу Общества. Реестр высылается по электронной почте. Поставщик проводит сверку платежей, принятых в течение контрольных суток, с платежами в реестре.

Платежи, полученные Поставщиком, но отсутствующие в реестре, считаются спорными. Поставщик должен получить подтверждение таких платежей у Общества. Неподтвержденные платежи удаляются. Поставщик вправе отменить спорные платежи сразу при их выявлении, и должен внести их заново после подтверждения Обществом.

Платежи, присутствующие в реестре, но не полученные Поставщиком, считаются спорными. Поставщик должен получить подтверждение таких платежей у Общества. Подтвержденные платежи заносятся Поставщиком.

Формат и шаблон реестра определяются условиями Договора.





3. ТИПЫ ШАБЛОНОВ РЕЕСТРОВ

3.2. Шаблон №1 (txt) (ПЦ №542)

Данный тип реестра:

3.2.1. формируется в формате txt

3.2.2. Входит в состав тарифов: ЖКХ Базовый, ЖКХ Оптимальный, ЖКХ Супер, Связь Базовый, Связь Оптимальный, Связь Супер.

3.2.3. Подходит для Реквизитных групп:

- ✓ ЖКХ Базовый 1,
- ✓ ЖКХ Базовый 2;
- ✓ ЖКХ Оптимальный 1,
- ✓ ЖКХ Оптимальный 2;
- ✓ ЖКХ Супер
- ✓ Связь 1
- ✓ Связь 2
- ✓ Связь 3
- ✓ Связь 4
- ✓ Связь 5

- **Формат имени файла**

[TRANSLIT]%RECPT_NAME%
%SERV_NAME%[/TRANSLIT]_[BEGIN_DATE]YYYY_MM_DD[/BEGIN_DATE]-
[END_DATE]YYYY_MM_DD[/END_DATE]_BS.txt

- **Пример имени файла реестра:**

firma_fontannaya_999-kapitalnyy_remont__2016_12_13-2016_12_13__BS.txt

- **Шапка**

~Плательщик: %ORG_NAME%

~Счет плательщика: %ORG_ACC%

~Банк плательщика: %BANK_NAME%

~БИК банка плательщика: %ORG_BIK%

~Корр.счет банка плательщика: %CORR_BANK_ACC%

~Получатель: %RECPT_NAME%

~Счет получателя: %SERV_ACC%

~Назначение платежа: Платежи %SERV_NAME% по принятым платежам с

[BEGIN_DATE]DD/MM/YYYY[/BEGIN_DATE] по [END_DATE]DD/MM/YYYY[/END_DATE]; на общую сумму %TOTAL_AMOUNT%, в том числе комиссия %BANK_FEE+AGENT_FEE%, в кол-ве %PAYMENTS_COUNT%, согласно реестру от [REGISTER_DATE]DD/MM/YYYY[/REGISTER_DATE]

~Банк получателя: %RECPT_BANK_NAME%

~БИК банка получателя: %RECPT_BIK%

~Корр.счет банка получателя: %RECPT_CORR_BANK_ACC%

- **Тело реестра**

%TERMINAL%; %PS_PAY_NUM%; [DATE]DD/MM/YYYY[/DATE]; %PACC%; %AMOUNT%; %DESCRIBE%

- **Пример данных о платеже в теле реестра**





ЦЕНТРАЛЬНАЯ
КАССА
ВСЕ ПЛАТЕЖИ ВОВРЕМЯ

Адрес

614087, Россия, г. Пермь,
ул. Малкова 12



8 (342) 240-40-22

info@ckassa.ru

ckassa.ru

ЖКХ Базовый 1:

1029/001; 13626119596; 13/12/2016; 092550138920; 229.67; Л/СЧЕТ: 092550138920; ФИО: ИВАНОВ И И; АДРЕС: ФОНТАННАЯ д.999, кв.999, эт.5; ДОП_ИНФ: ;

ЖКХ Базовый 2:

1025/001; 13626116516; 13/12/2016; 810733001920; 329.73; Л/СЧЕТ: 810733001920; ФИО: ИВАНОВ; АДРЕС: ул.БЕЛЯЕВА,999-99; МЕСЯЦ: 11.2016; ДОП_ИНФ: ;

ЖКХ Оптимальный 1:

1025/001; 13626116516; 13/12/2016; 810733001920; 329.73; Л/СЧЕТ: 810733001920; ФИО: ИВАНОВ; АДРЕС: ул.БЕЛЯЕВА,999-99; МЕСЯЦ: 11.2016;

ЗАДОЛЖЕННОСТЬ: 0; ПЕНИ 0; ДОП_ИНФ: ;

ЖКХ Оптимальный 2:

1025/001; 13626116516; 13/12/2016; 810733001920; 329.73; Л/СЧЕТ: 810733001920; ФИО: ИВАНОВ; АДРЕС: ул.БЕЛЯЕВА,999-99; МЕСЯЦ: 11.2016; ЗАДОЛЖЕННОСТЬ: 0; ПЕНИ 0; ГВС: 0; ХВС: 0; ДОП_ИНФ: ;

ЖКХ Супер:

1025/001; 13626116516; 13/12/2016; 810733001920; 329.73; Л/СЧЕТ: 0099901; ФИО: Иванова Ф.Н.; АДРЕС: ул. Юнатов 99-99; МЕСЯЦ: 12.2016; ЗАДОЛЖЕННОСТЬ: 0; ПЕНИ: 0; ТИП_ПЛАТЕЖА: 0; ПОЛЕ4: 0; ПОЛЕ5: 0; ПОЛЕ6: 0; ПОЛЕ7: 0; ПОЛЕ8: 0; ПОЛЕ9: 0; ПОЛЕ10: 0; ПОЛЕ11: 0; ПОЛЕ13: 0; ПОЛЕ15: 0; ДОП_ИНФ: ;

Связь 1-5:

111/001; 13662014924; 13/03/2017; 0137; 1000.00; ЛИЦЕВОЙ_СЧЕТ: 0000; ФИО: Иванова А.А.;





3.3. **Шаблон №2** (csv) (ПЦ№ 350)

Данный тип реестра:

3.3.1. формируется в формате excel

3.3.2. Входит в состав тарифов: ЖКХ Базовый, ЖКХ Оптимальный, ЖКХ Супер, Связь Базовый, Связь Оптимальный, Связь Супер.

3.3.3. Подходит для Реквизитных групп:

- ✓ ЖКХ Базовый 1,
- ✓ ЖКХ Базовый 2;
- ✓ ЖКХ Оптимальный 1,
- ✓ ЖКХ Оптимальный 2;
- ✓ ЖКХ Супер
- ✓ Связь 1
- ✓ Связь 2
- ✓ Связь 3
- ✓ Связь 4
- ✓ Связь 5

• Формат имени файла

[TRANSLIT]%RECPT_NAME%
%SERV_NAME%[/TRANSLIT]__[BEGIN_DATE]YYYY_MM_DD[/BEGIN_DATE]-
[END_DATE]YYYY_MM_DD[/END_DATE]__BS.csv

• Пример имени файла реестра:

firma-999-kosmonavta_belyaeva_999,_kapitalnyy_remont__2016_12_13-2016_12_13__BS.csv

• Шапка

~Плательщик: %ORG_NAME%

~Счет плательщика: %ORG_ACC%

~Банк плательщика: %BANK_NAME%

~БИК банка плательщика: %ORG_BIK%

~Корр.счет банка плательщика: %CORR_BANK_ACC%

~Получатель: %RECPT_NAME%

~Счет получателя: %SERV_ACC%

~Назначение платежа: Платежи %SERV_NAME% по принятым платежам с

[BEGIN_DATE]DD/MM/YYYY[/BEGIN_DATE] по [END_DATE]DD/MM/YYYY[/END_DATE]; на
общую сумму %TOTAL_AMOUNT%, в том числе комиссия %BANK_FEE+AGENT_FEE%, в кол-ве
%PAYMENTS_COUNT%, согласно реестру от [REGISTER_DATE]DD/MM/YYYY[/REGISTER_DATE]

~Банк получателя: %RECPT_BANK_NAME%

~БИК банка получателя: %RECPT_BIK%

~Корр.счет банка получателя: %RECPT_CORR_BANK_ACC%

Терминал; Номер платежа; Дата платежа; Лицевой счёт; Сумма; Примечание



- **Тело реестра**

%TERMINAL%; %PS_PAY_NUM%; [DATE]DD/MM/YYYY[/DATE]; %PACC%; %AMOUNT%;
%DESCRIBE%

- **Пример данных о платеже в теле реестра:**

ЖКХ Базовый 1:

Терминал; Номер платежа; Дата платежа; Лицевой счёт; Сумма; Примечание
1025/001; 13626116516; 13/12/2016; 810733001920; 329.73; Л/СЧЕТ: 810733001920; ФИО: ИВАНОВ;
АДРЕС: ул.БЕЛЯЕВА,999-99; ДОП_ИНФ: ;

ЖКХ Базовый 2:

Терминал; Номер платежа; Дата платежа; Лицевой счёт; Сумма; Примечание
1025/001; 13626116516; 13/12/2016; 810733001920; 329.73; Л/СЧЕТ: 810733001920; ФИО: ИВАНОВ;
АДРЕС: ул.БЕЛЯЕВА,999-99; МЕСЯЦ: 11.2016; ДОП_ИНФ: ;

ЖКХ Оптимальный 1:

Терминал; Номер платежа; Дата платежа; Лицевой счёт; Сумма; Примечание
1025/001; 13626116516; 13/12/2016; 810733001920; 329.73; Л/СЧЕТ: 810733001920; ФИО: ИВАНОВ;
АДРЕС: ул.БЕЛЯЕВА,999-99; МЕСЯЦ: 11.2016;
ЗАДОЛЖЕННОСТЬ: 0; ПЕНИ 0; ДОП_ИНФ: ;

ЖКХ Оптимальный 2:

Терминал; Номер платежа; Дата платежа; Лицевой счёт; Сумма; Примечание
1025/001; 13626116516; 13/12/2016; 810733001920; 329.73; Л/СЧЕТ: 810733001920; ФИО: ИВАНОВ;
АДРЕС: ул.БЕЛЯЕВА,999-99; МЕСЯЦ: 11.2016; ЗАДОЛЖЕННОСТЬ: 0; ПЕНИ 0; ГВС: 0; ХВС: 0;
ДОП_ИНФ: ;

ЖКХ Супер:

Терминал; Номер платежа; Дата платежа; Лицевой счёт; Сумма; Примечание
1025/001; 13626116516; 13/12/2016; 810733001920; 329.73; Л/СЧЕТ: 0099901; ФИО: Иванова Ф.Н.;
АДРЕС: ул. Юнатов 99-99; МЕСЯЦ: 12.2016; ЗАДОЛЖЕННОСТЬ: 0; ПЕНИ: 0; ТИП_ПЛАТЕЖА: 0;
ПОЛЕ4: 0; ПОЛЕ5: 0; ПОЛЕ6: 0; ПОЛЕ7: 0; ПОЛЕ8: 0; ПОЛЕ9: 0; ПОЛЕ10: 0; ПОЛЕ11: 0; ПОЛЕ13: 0;
ПОЛЕ15: 0; ДОП_ИНФ: ;

Связь 1-5:

Терминал; Номер платежа; Дата платежа; Лицевой счёт; Сумма; Примечание
111/001; 13662014924; 13/03/2017; 0000; 1000.00; ЛИЦЕВОЙ_СЧЕТ: 0000; ФИО: Иванова М.А.;





3.4. **Шаблон №3** (xml) (ПЦ №590)

Данный тип реестра:

- 3.4.1. формируется в формате XML, Кодировка UTF-8
- 3.4.2. Подходит для расчетных центров
- 3.4.3. Входит в состав тарифа: ЖКХ Базовый, ЖКХ Оптимальный, ЖКХ Супер, Связь Базовый, Связь Оптимальный, Связь Супер.
- 3.4.4. Подходит для Реквизитных групп:
 - ✓ ЖКХ Базовый 1,
 - ✓ ЖКХ Базовый 2;
 - ✓ ЖКХ Оптимальный 1,
 - ✓ ЖКХ Оптимальный 2;
 - ✓ ЖКХ Супер
 - ✓ Связь 1
 - ✓ Связь 2
 - ✓ Связь 3
 - ✓ Связь 4
 - ✓ Связь 5

• **Формат имени файла**

%RECPT_NAME%_[BEGIN_DATE]YYYY_MM_DD[/BEGIN_DATE]-
[END_DATE]YYYY_MM_DD[/END_DATE]__BS12.xml

• **Пример имени файла реестра:**

ФИРМА__2016_12_13-2016_12_13__BS12.xml

• **Шапка**

```
<?xml version="1.0" encoding="UTF-8" ?>
```

```
<registry>
```

```
<header>
```

```
<registry_version>1.0</registry_version>
```

```
<registry_id>306</registry_id>
```

```
<payer_id>%ORG_NAME%</payer_id>
```

```
<acceptor_id>%RECPT_NAME%</acceptor_id>
```

```
<registry_create_time>[REGISTER_DATE]YYYY-MM-DD"T"HH24:MI:SS[/REGISTER_DATE]</registry_create_time>
```

```
<payer_name>%ORG_NAME%</payer_name>
```

```
<payer_bank_account>%ORG_ACC%</payer_bank_account>
```

```
<payer_bank_name>%BANK_NAME%</payer_bank_name>
```

```
<payer_bank_BIK>%ORG_BIK%</payer_bank_BIK>
```

```
<payer_bank_corresponding_account>%CORR_BANK_ACC%</payer_bank_corresponding_account>
```

```
<acceptor_name>%RECPT_NAME%</acceptor_name>
```

```
<acceptor_bank_account>%RECPT_ACC%</acceptor_bank_account>
```

```
<acceptor_bank_name>%RECPT_BANK_NAME%</acceptor_bank_name>
```

```
<acceptor_bank_BIK>%RECPT_BIK%</acceptor_bank_BIK>
```

```
<acceptor_bank_corresponding_account>%RECPT_CORR_BANK_ACC%</acceptor_bank_corresponding_acc  
ount>
```





```

<first_payment_date>[FIRST_PAYMENT_DATE]YYYY-MM-
DD"T"HH24:MI:SS[/FIRST_PAYMENT_DATE]</first_payment_date>
<last_payment_date>[LAST_PAYMENT_DATE]YYYY-MM-
DD"T"HH24:MI:SS[/LAST_PAYMENT_DATE]</last_payment_date>
<registry_summ>%TOTAL_AMOUNT%</registry_summ>

<tax_summ>[SUB]%BANK_FEE+AGENT_FEE%[/NDS]%BANK_FEE+AGENT_FEE%[/NDS][[/SUB]</tax_s
umm>
<tax_summ_nds>[NDS]%BANK_FEE+AGENT_FEE%[/NDS]</tax_summ_nds>
<record_count>%PAYMENTS_COUNT%</record_count>
</header>
<data>
  • Тело реестра
    <record rec_num="%NUMBER%">
      <payment_id>%PS_PAY_NUM%</payment_id>
      <date>[DATE]YYYY-MM-DD"T"HH24:MI:SS[/DATE]</date>
      <account>%РАСС|. {0,10}%</account>
      <summ>%AMOUNT%</summ>
      <fio>#ФИО#</fio>
      <address>#АДРЕС#</address>
      <type>текущий</type>
      <period_from>[МЕСЯЦ]ММ.YYYY|YYYY-ММ[/МЕСЯЦ]</period_from>
      <period_to>[МЕСЯЦ]ММ.YYYY|YYYY-ММ[/МЕСЯЦ]</period_to>
      <services>
        <penaltyfee>0</penaltyfee>
        <insurance>0</insurance>
      </services>
    </record>

  • Подвал
</data>
</registry>

```

• Пример данных о платеже в теле реестра:

```

<record rec_num="1">
  <payment_id>13626116963</payment_id>
  <date>2016-12-13T21:00:10</date>
  <account>0150903999</account>
  <summ>2962.64</summ>
  <fio>Иванова Т.Г.</fio>
  <address>ул. Иванова д. 99, кв. 99, эт.99</address>
  <type>текущий</type>
  <period_from>2016-11</period_from>
  <period_to>2016-11</period_to>
  <services>
    <penaltyfee>0</penaltyfee>
    <insurance>0</insurance>
  </services>
</record>

```



3.5. **Шаблон №4** (xml) (ПЦ №589)

Данный тип реестра:

- 3.5.1. формируется в формате XML, Кодировка Windows-1251
- 3.5.2. Подходит для расчетных центров
- 3.5.3. Входит в состав тарифа: ЖКХ Базовый, ЖКХ Оптимальный, ЖКХ Супер, Связь Базовый, Связь Оптимальный, Связь Супер
- 3.5.4. Подходит для Реквизитных групп:
 - ✓ ЖКХ Базовый 1,
 - ✓ ЖКХ Базовый 2;
 - ✓ ЖКХ Оптимальный 1,
 - ✓ ЖКХ Оптимальный 2;
 - ✓ ЖКХ Супер
 - ✓ Связь 1
 - ✓ Связь 2
 - ✓ Связь 3
 - ✓ Связь 4
 - ✓ Связь 5

- **Формат имени файла**

%RECPT_NAME%_[BEGIN_DATE]YYYY_MM_DD[/BEGIN_DATE]-
[END_DATE]YYYY_MM_DD[/END_DATE]__BS12.xml

- **Пример имени файла реестра:**

ФИРМА__2016_12_13-2016_12_13__BS12.xml

- **Шаблон**

```
<?xml version="1.0" encoding=" Windows-1251" ?>
<registry>
  <header>
    <registry_version>1.0</registry_version>
    <registry_id>306</registry_id>
    <payer_id>%ORG_NAME%</payer_id>
    <acceptor_id>%RECPT_NAME%</acceptor_id>
    <registry_create_time>[REGISTER_DATE]YYYY-MM-
DD"T"HH24:MI:SS[/REGISTER_DATE]</registry_create_time>
    <payer_name>%ORG_NAME%</payer_name>
    <payer_bank_account>%ORG_ACC%</payer_bank_account>
    <payer_bank_name>%BANK_NAME%</payer_bank_name>
    <payer_bank_BIK>%ORG_BIK%</payer_bank_BIK>
    <payer_bank_corresponding_account>%CORR_BANK_ACC%</payer_bank_corresponding_account>
    <acceptor_name>%RECPT_NAME%</acceptor_name>
    <acceptor_bank_account>%RECPT_ACC%</acceptor_bank_account>
    <acceptor_bank_name>%RECPT_BANK_NAME%</acceptor_bank_name>
    <acceptor_bank_BIK>%RECPT_BIK%</acceptor_bank_BIK>
```



<acceptor_bank_corresponding_account>%RECPT_CORR_BANK_ACC%</acceptor_bank_corresponding_acc
ount>

<first_payment_date>[FIRST_PAYMENT_DATE]YYYY-MM-
DD"T"HH24:MI:SS[/FIRST_PAYMENT_DATE]</first_payment_date>

<last_payment_date>[LAST_PAYMENT_DATE]YYYY-MM-
DD"T"HH24:MI:SS[/LAST_PAYMENT_DATE]</last_payment_date>

<registry_summ>%TOTAL_AMOUNT%</registry_summ>

<tax_summ>[SUB]%BANK_FEE+AGENT_FEE%[/NDS]%BANK_FEE+AGENT_FEE%[/NDS][SUB]</tax_s
umm>

<tax_summ_nds>[NDS]%BANK_FEE+AGENT_FEE%[/NDS]</tax_summ_nds>

<record_count>%PAYMENTS_COUNT%</record_count>

</header>

<data>

- **Тело реестра**

<record rec_num="%NUMBER%">

<payment_id>%PS_PAY_NUM%</payment_id>

<date>[DATE]YYYY-MM-DD"T"HH24:MI:SS[/DATE]</date>

<account>%PACC|. {0,10}%</account>

<summ>%AMOUNT%</summ>

<fio>#ФИО#</fio>

<address>#АДРЕС#</address>

<type>текущий</type>

<period_from>[МЕСЯЦ]ММ. YYYY|YYYY-ММ[/МЕСЯЦ]</period_from>

<period_to>[МЕСЯЦ]ММ. YYYY|YYYY-ММ[/МЕСЯЦ]</period_to>

<services>

<penaltyfee>0</penaltyfee>

<insurance>0</insurance>

</services>

</record>

- **Подвал**

</data>

</registry>

- **Пример данных о платеже в теле реестра:**

<record rec_num="1">

<payment_id>13626116963</payment_id>

<date>2016-12-13T21:00:10</date>

<account>0150903999</account>

<summ>2962.64</summ>

<fio>Иванова Т.Г.</fio>

<address>ул. Иванова д. 99, кв. 99, эт.99</address>

<type>текущий</type>

<period_from>2016-11</period_from>

<period_to>2016-11</period_to>

<services>

<penaltyfee>0</penaltyfee>



<insurance>0</insurance>
</services>
</record>

3.6. Шаблон №5 (txt) (ПЦ №367)

Данный тип реестра:

3.6.1. формируется в формате txt

3.6.2. Подходит для расчетных центров

3.6.3. Входит в состав тарифов: ЖКХ Базовый, ЖКХ Оптимальный, ЖКХ Супер.

3.6.4. Подходит для Реквизитных групп:

- ✓ ЖКХ Базовый 1,
- ✓ ЖКХ Базовый 2;
- ✓ ЖКХ Оптимальный 1,
- ✓ ЖКХ Оптимальный 2;
- ✓ ЖКХ Супер

- **Формат имени файла**

bs-%RECP CODE%-[TRANSLIT]%ACT_NAME%[/TRANSLIT]-
[BEGIN_DATE]yyyymmdd[/BEGIN_DATE][END_DATE_IF_DIF]"-yyyymmdd[/END_DATE_IF_DIF].txt

- **Пример имени файла реестра:**

bs-11683-kap_remont-20170425-20170524.txt

- **Шапка**

~Плательщик: %ORG_NAME%
~Счет плательщика: %ORG_ACC%
~Банк плательщика: %BANK_NAME%
~БИК банка плательщика: %ORG_BIK%
~Корр.счет банка плательщика: %CORR_BANK_ACC%
~Получатель: %RECP NAME%
~Счет получателя: %RECP_ACC%
~Назначение платежа: ЗА ЖКУ; по принятым платежам с
[BEGIN_DATE]DD/MM/YYYY[/BEGIN_DATE] по [END_DATE]DD/MM/YYYY[/END_DATE];
на общую сумму %TOTAL_AMOUNT%, в том числе комиссия %AGENT_FEE%, в кол-
ве %PAYMENTS_COUNT%, согласно реестру
№%RECP_ORG_ID%[REGISTER_DATE]YYYYMMDD[/REGISTER_DATE] от
[REGISTER_DATE]DD/MM/YYYY[/REGISTER_DATE]
~Банк получателя: %RECP_BANK_NAME%
~БИК банка получателя: %RECP_BIK%
~Корр.счет банка получателя: %RECP_CORR_BANK_ACC%
~Договор № %CONTRACT_INFO%

- **Тело реестра**

0 0 [DATE]DD/MM/YYYY[/DATE] %UNICUMCODE% %SERV_CODE% %SERV_ACC% %AMOUNT%
%AGENT_FEE_AMOUNT% %DESCRIBE%

- **Пример данных о платеже в теле реестра**



0 0 25/04/2017 13677581462 111-11683-2 40705810649770001122 150.66 0.00 Л_СЧЕТ:
2910001111; ФИО: Иванов ИИ; АДРЕС: Б Гагарина26а 228;

4. ОБРАБОТКА ДВУХМЕРНОГО ШТРИХ КОДА (ДШК)

4.2. Общество принимает в обработку ДШК (Двумерный штрих код), сформированный в соответствии с стандарт квитанций и платёжных документов с двухмерным штрих-кодом (ДШК) - ГОСТ Р 56042-2014.

4.3. Данная форма ДШК аналогична форме принимаемой ПАО Сбербанк

4.4. Ниже приведены Правила формирования ДШК, (на примере ОАО «Кузбассэнергосбыт»)

4.5. Поставщик самостоятельно определяет перечень полей, которые используются в ДШК.

Критериями являются: использование обязательных реквизитов и соответствие имени реквизита стандарту ДШК.

Перечень полей, используемых в ДШК			
№ п/п	Наименование реквизита	Наименование реквизита	Пример
Обязательные реквизиты			
1	Идентификатор стандарта	ST, версия стандарта, код кодировки текста	ST00011
2	Name	Наименование получателя платежа	Южное межрайонное отделение ОАО "Кузбассэнергосбыт"
3	PayeeINN	ИНН получателя платежа	4205109214
4	PersonalAcc	Номер счета получателя платежа	40702810005090000071
5	BankName	Наименование банка получателя платежа	Кузбасский филиал ОАО "Углеметбанк" г.Кемерово
6	BIC	БИК	43207707
7	CorrespAcc	Номер кор./сч. банка получателя платежа	30101810500000000707
8	lastName	Фамилия плательщика	Иванов
9	firstName	Имя плательщика	Иван
10	middleName	Отчество плательщика	Иванович
11	payerAddress	Адрес плательщика	Пермь, ул. Дорожная, д.5 кв.3
12	paymPeriod	Период оплаты	62015
13	Sum	Сумма платежа, в копейках	53890
14	AddAmount	Пени (в случае отсутствия или не начисления указывать 000)	0
15	persAcc	Номер лицевого счета плательщика в организации (в системе учета ПУ)	504052854
16	category	Идентификатор реестра для биллинга (номер услуги)	1300000000
Дополнительные реквизиты			
1	Purpose	Наименование платежа (назначение)	Электроэнергия
2	serviceName	Код услуги	40
3	counterVal	Показание прибора учета	12615

Пример считанного ДШК(QR-кода) АО ВЦ "Инкомус":



ST00011|Name=АО ВЦ "Инкомус"|PersonalAcc=40702810149090110428|BankName=в Пермском
отделении №6984 ЗУБ ПАО
Сбербанк|BIC=045773603|CorrespAcc=30101810900000000603|PayeeINN=5902181851|KPP=59020
1001|Sum=727732|Purpose=Жилищно-коммунальные услуги|payerFio=Иванов И И
|payerAddress=Дорожная д.62, кв.2,
эт.1|persAcc=019041662222|paymPeriod=01.2016|TechCode=02|calc=412133|debt=315483|penaltyfee
=116|insurance=10060

Значение QR=кода считано из приложенной квитанции:

СЧЕТ-ИЗВЕЩЕНИЕ

019041662216010072732222 **Лицевой счет** **01.2016** **итого к опл.** **7 277,32**

АО ВЦ «Инкомус», ИНН/КПП 5902181851/590201001, р/с 40702810149090110428
в Пермском отделении №6984 ЗУБ ПАО Сбербанк, БИК 045773603, к/с 30101810900000000603

добровольное
страхование
100,60

Прибор учета	Пред.пок.	Тек.пок.	Расход	Вид услуг	Текущие показания общедом. ИПУ	Объем потребл. по общедом. ИПУ	Объем потребл. в помещени-ях дома	Объем услуги на ОДН	Составл. от норматива на ОДН	Коэф. прожита-и в доме	Общ. пл. жилья в многоквартирн. постройке	Объем услуги на единицу общ. пл.
ХВ 1	34	35	1	1	2	3	4	5	6	7	8	9
ХВ 2	200	207	7	ГВ(вод.) на ОДН		1230,000	939,900	212,836		563	12069,10	0,017635
ГВ 1	27	28	1	ГВ(под.) на ОДН		1613,000	1261,257	212,836		563	12069,10	0,017635
ГВ 2	53	54	1	Отопление		332,717		0,000			10631,90	0,031294
				Гор.вода(подог.)				0,000				0,054000

Подпись

СЧЕТ-КВИТАНЦИЯ

Лицевой счет **01.2016** Площадь помещения общедомовая 50,30 /29,60 Количество проживающих постоянно/врем.отсут. 5 /0

VISA MasterCard
Онлайн оплата ЖКУ на inkomus.ru

Вид услуг	Ед. изм.	Кол-во/норматив объема на ед.	Объем потребленных услуг	Тариф/ Размер платы	Всего начислено за расчетный период	Снято/ добавлено	Код прич. пересчета	Льготы	Итого к оплате за расчетный период
1	2	3	4	5	6	7	8	9	10
Содержание жилья	м²		50,30000	17,870	898,86				898,86
Текущий ремонт	м²		50,30000	7,470	375,74				375,74
Отопление	Гкал	0,031294	1,57409	1338,380	2 106,73				2 106,73
Водоотведение	м³		10,00000	17,330	173,30				173,30
Холодная вода	м³		8,00000	30,410	243,28	26,97			270,25
Гор.вода(ХВС)	м³	0,017635	2,00000	0,8870	60,82	26,97			87,79
Гор.вода(подог.)	м³	0,017635	0,10800	0,8870	1338,380	144,55			208,66

Итого за расч. период **4 121,33**

Задолженность (1 мес.) **3 154,83**

Пени на 16.01.2016 **1,16**

Итого **7 277,32**

пн,пт.9-17.,вт.9.-11.30,ср.9.-19.
прием заяв. 213-31-82 (круглосут.)

Подпись

Прийем показаний ИПУ с 20 по 25 число: 2-700-717, 2-040-717 (круглосуточно) или на сайте www.inkomus.ru

Пример настройки ДШК в программе 1С можно посмотреть по этой ссылке: [Формат QR штрихкода в программе 1С: Учет в управляющих компаниях ЖКХ, ТСЖ и ЖСК](#)



5. ТИПОВЫЕ СПЕЦИФИКАЦИИ ОНЛАЙН-ВЗАИМОДЕЙСТВИЯ

Термины и определения:

Общество –юридическое лицо, действующее по поручению Поставщика, являющееся посредником между Поставщиком и конечным потребителем услуги.

Поставщик – Поставщик услуг - юридическое лицо или индивидуальный предприниматель, уполномочивающее другое юридическое лицо действовать в качестве Общества.

Плательщик (клиент) – физическое лицо, осуществляющее внесение Обществу денежных средств в целях исполнения денежных обязательств физического лица перед Поставщиком, с которым заключен Клиентский Договор, и оплачивающий Услуги, оказываемые Поставщиком, через ППП.

ПО – Программное обеспечение

При возникновении проблем с подключением по выбранной Спецификации в течение 1 (одного) месяца с момента начала технической интеграции, подключение осуществляется по стандартному протоколу взаимодействия без онлайн-проверки ключевого реквизита и передачи информации о платеже в режиме реального времени.

Стандартный протокол взаимодействия - это протокол технического взаимодействия, при котором реквизиты плательщика вносятся в базу данных Общества по мере их накопления и выдаче этих реквизитов плательщику для верификации с третьего платежа по ключевому реквизиту.

5.2. Спецификация №1

5.2.1. Общие положения

Примечание: Описание протокола общее. Конкретные данные, передающиеся в запросах и ответах, определяются условиями Договора.

Поставщик принимает запросы с данными о платежах от Общества. Для обработки запросов Поставщику необходимо ПО: HTTP-сервер, программы (скрипты), которые обрабатывают запросы Общества и заносят платежи в базу данных Поставщика. (Чаще используют Apache и скрипты, написанные на PHP.)

Обмен данными проходит по протоколу HTTPS. Для авторизации Общества используется MD5-хэш данных запроса с паролем, в шестнадцатеричном виде. ПО Поставщика должно проверять IP-адрес, с которого поступают запросы, и MD5-хэш. Поставщик обязан отвергать все запросы, совершенные с других IP-адресов, и/или с неверным значением MD5-хэша. Для идентификации Поставщика используется MD5-хэш данных ответа с паролем. В случае получения неверного значения хэша в ответе, Общество считает ответ ошибочным и должен сообщить об этом Поставщику.

5.2.2. Формат запросов и ответов

Тип запросов POST, кодировка windows-1251 или UTF-8. Кодировку выбирает Поставщик. Все запросы Общества и ответы Поставщика должны быть в одной кодировке. Тело POST-запроса содержит один параметр params, в котором передается XML-запрос. Данные URL-кодированные. Заголовок Content-Type: application/x-www-form-urlencoded. Аналогичный запрос будет, если создать HTML-форму с элементом <textarea name="params" ...>, поместить в элемент текст XML-запроса и отправить средствами браузера.



Ответы Поставщика в формате XML, кодировка соответственно кодировке запросов. В HTTP-заголовке желательно установить Content-Type: text/xml; charset=<кодировка>.

Ограничения XML. Запрещены пробелы до или после имени тэга (между <>). Общество гарантирует отсутствие таких пробелов в запросах. В связи с этим, и с тем, что структура XML простая, возможно не использовать для обработки XML специальные компоненты, а обрабатывать функциями работы со строками.

(Достаточно написать одну строковую функцию, типа содержимое_тэга = get_tag_content(запрос, имя_тэга); или использовать регулярные выражения.)

Общий формат запроса:

```
<?xml version="1.0" encoding="{кодировка}"?>
<request>
  <params>
    {параметры}
  </params>
  <sign>{подпись}</sign>
</request>
```

Элементы запроса:

request – корневой элемент, содержит элементы:

params – содержит элементы запроса

sign – содержит подпись запроса. Правила формирования описаны в п.4.

Общий формат ответа:

```
<?xml version="1.0" encoding="{кодировка}"?>
<response>
  <params>
    <err_code>{кодошибки}</err_code>
    <err_text>[текстошибки]</err_text>
    [параметры]
  </params>
  <sign>{подпись}</sign>
</response>
```

Элементы ответа:

response – корневой элемент, содержит элементы:

params – содержит элементы запроса

sign – содержит подпись запроса. Правила формирования описаны в п.4.

Типы данных

Тип	Описание
N	Целое число
S[n]	Текстовое значение длиной не более n символов
DATETIME	Дата и время в формате "YYYY-MM-DDTHH24:MI:SS"

5.2.3. Запрос на проверку параметров платежа

Общество может предварительно проверить возможность занесения платежа с помощью запроса проверки.

Поля запроса (элемент params)

Поле	Обязательность	Тип	Назначение	Комментарий
act	Да	S[2]	Тип запроса	1 – проверка параметров будущего платежа



Поле	Обязательность	Тип	Назначение	Комментарий
account	Да	S[100]	Идентификатор плательщика	Обычно лицевой счет клиента или телефон
pay_amount	Нет	N	Сумма платежа	Сумма платежа в копейках, целое число.
agent_code	Нет	S[30]	Код Общества	Используется, когда у Поставщика есть другие Общества работающие по данному протоколу
serv_code	Нет	S[32]	Код услуги	Используется когда у Поставщика могут совпадать номера лицевых счетов разных услуг.
agent_date	Нет	DATETIME	Дата учета, для сверки платежей	Используется дата получения платежа Обществом по часовому поясу Поставщика
param_name1 param_name2 ...	Нет	S	Дополнительные параметры платежа	Имена могут быть выбраны Оператором.

Сумма платежа в запросе проверки не является обязательным параметром. О передаче дополнительных реквизитов Поставщику следует договариваться сначала с Обществом, принимающим платежи у клиентов. Формат реквизитов также обговаривается между Поставщиком и СубОбществом, т.к. Общество передает эти параметры без изменений.

Поля ответа (элемент params)

Поле	Обязательность	Тип	Назначение	Комментарий
err_code	Да	N	Код ошибки	0, если запрос успешный. Коды ошибок в приложении
err_text	Нет	S[200]	Текст ошибки	Текст показывается плательщику
desired_amount	Нет	N	Желательная сумма платежа в копейках	
param_name1 param_name2 ...	Нет	S	Дополнительные реквизиты	Имена могут быть выбраны Оператором.

Для удобства клиентов рекомендуются дополнительные элементы:

1. *client_name* – фамилия, имя, отчество клиента;
2. *balance* - баланс л/счета клиента, в рублях. Суммы в дополнительных реквизитах передаются в рублях, копейки разделены точкой. Эти реквизиты будут отображены на терминале во время оплаты.

Внимание! Дополнительные реквизиты показываются плательщику без изменений, поэтому все суммы желательно указывать в рублях. В то же время параметр *desired_amount* – это целое число в копейках.

Пример запроса проверки возможности занесения платежа:

```
<?xml version="1.0" encoding="windows-1251"?>
```





```
<request>
<params>
  <act>1</act>
  <agent_date>2009-04-15T11:22:33</agent_date>
  <account>54321</account>
  <serv_code>53001</serv_code>
  <pay_amount>10000</pay_amount>
  <client_name>Иванов Иван Иванович</client_name>
</params>
<sign>827CCB0EEA8A706C4C34A16891F84E7B</sign>
</request>
```

Пример ответа Поставщика на запрос проверки:

```
<?xml version="1.0" encoding="windows-1251"?>
<response>
<params>
  <err_code>0</err_code>
  <err_text>ОК</err_text>
  <account>54321</account>
  <client_name>Иванов Иван Иванович</client_name>
  <balance>50.00</balance>
</params>
<sign>827CCB0EEA8A706C4C34A16891F84E7B</sign>
</response>
```

Пример ответа Поставщика при ошибке:

```
<?xml version="1.0" encoding="windows-1251"?>
<response>
<params>
  <err_code>20</err_code>
  <err_text>Номер счета не найден</err_text>
</params>
<sign>827CCB0EEA8A706C4C34A16891F84E7B</sign>
</response>
```

5.2.4. Запрос на проведение платежа

Общество может не использовать предварительный запрос проверки. При проведении платежа все параметры запроса проверяются Поставщиком так же как при запросе проверки.

Поля запроса

Поле	Обязательность	Тип	Назначение	Комментарий
act	Да	S[2]	Тип запроса	2 – проведение платежа
account	Да	S[100]	Идентификатор плательщика	Обычно лицевой счет клиента или телефон
pay_amount	Да	N	Сумма платежа	Сумма платежа в копейках
pay_id	Да	S[50]	Уникальный номер платежа	
pay_date	Да	DATETIME	Дата платежа	Дата принятия платежа у клиента по часовому поясу в месте принятия платежа.
pay_type	Нет	S[10]	Вид платежа	Используется, например, когда платежи надо делить на наличные и безналичные
agent_code	Нет	S[30]	Код Общества	Используется, когда у





Поле	Обязательность	Тип	Назначение	Комментарий
				Поставщика есть другие Общественные работающие по данному протоколу
serv_code	Нет	S[32]	Код услуги	Используется когда у Поставщика могут совпадать номера лицевых счетов разных услуг.
agent_date	Нет	DATETIME	Дата учета, для сверки платежей	Используется дата получения платежа Обществом по часовому поясу Поставщика
Sign	Да	S[32]	Подпись запроса	Правила формирования описаны в п.6.
param_name1 param_name2 ...	Нет	S	Дополнительные параметры платежа	Имена могут быть выбраны Поставщиком.

Поля ответа

Поле	Обязательность	Тип	Назначение	Комментарий
err_code	Да	N	Код ошибки	0, если запрос успешный. Коды ошибок в приложении
err_text	Нет	S[200]	Текст ошибки	Текст показывается плательщику
reg_id	Нет	S[50]	Идентификатор платежа у Оператора	
reg_date	Нет	DATETIME	Дата регистрации платежа у Оператора	

Параметры reg_id и reg_date добавляются по усмотрению Поставщика. Могут быть использованы для сверки платежей за период по параметрам Поставщика. Дополнительные реквизиты и desired_amount в ответе на запрос проведения платежа не передаются.

Пример запроса на проведение платежа:

```
<?xml version="1.0" encoding="windows-1251"?>
<request>
  <params>
    <act>2</act>
    <agent_date>2009-04-15T11:22:33</agent_date>
    <pay_id>2345</pay_id>
    <pay_date>2009-04-15T11:00:12</pay_date>
    <account>54321</account>
    <pay_amount>10000</pay_amount>
    <client_name>Иванов</client_name>
    <month>08.2012</month>
  </params>
  <sign>827CCB0EEA8A706C4C34A16891F84E7B</sign>
</request>
```



Пример ответа Поставщика на запрос проведения:

```
<?xml version="1.0" encoding="windows-1251"?>
<response>
  <params>
    <err_code>0</err_code>
    <err_text>OK</err_text>
    <reg_id>3456</reg_id>
    <reg_date>2009-04-15T11:22:55</reg_date>
  </params>
  <sign>827CCB0EEA8A706C4C34A16891F84E7B</sign>
</response>
```

5.2.5. Запрос на получение статуса платежа

Иногда статус платежа у Поставщика не конечный на момент выдачи ответа на запрос проведения. В таких случаях Поставщик отвечает кодом 2 (Платеж ожидает обработки у Поставщика). После этого Общество будет посылать запросы на получение статуса. На такие запросы Поставщик отвечает кодам 0 (платеж проведен), 2 (платеж в процессе обработки), 40 (временная ошибка обработки) или 41 (окончательная ошибка обработки).

Поля запроса

Поле	Обязательность	Тип	Назначение	Комментарий
act	Да	S[2]	Тип запроса	4– проверка статуса платежа
pay_id	Да	S[50]	Уникальный номер платежа	Такой же, который использовался при проведении платежа

Поля ответа аналогичные как у запроса проведения платежа.

Пример запроса на проверку статуса:

```
<?xml version="1.0" encoding="windows-1251"?>
<request>
  <params>
    <act>4</act>
    <pay_id>2345</pay_id>
  </params>
  <sign>827CCB0EEA8A706C4C34A16891F84E7B</sign>
</request>
```

Пример ответа Поставщика на запрос статуса:

```
<?xml version="1.0" encoding="windows-1251"?>
<response>
  <params>
    <err_code>0</err_code>
    <err_text>Платеж обработан</err_text>
  </params>
  <sign>827CCB0EEA8A706C4C34A16891F84E7B</sign>
</response>
```

Пример ответа Поставщика на запрос статуса в случае временной ошибки:

```
<?xml version="1.0" encoding="windows-1251"?>
<response>
  <params>
    <err_code>40</err_code>
    <err_text>Ошибка подключения к серверу получателя.</err_text>
  </params>
  <sign>827CCB0EEA8A706C4C34A16891F84E7B</sign>
</response>
```

5.2.6. Запрос на возврат платежа.



В некоторых случаях требуется отмена платежа проведенного Поставщику. Запрос на возврат передает, как бы, заявку на возврат платежа. Поставщик рассматривает возможность отмены платежа. При возможности отменяет платежи, меняет их статус. Общество может узнать об итоге операции запросом статуса платежа, или письменным уведомлением.

Запрос на возврат не обязателен для протокола. Поставщик решает сам, нужна ли ему такая возможность.

Поля запроса

Поле	Обязательность	Тип	Назначение	Комментарий
act	Да	S[2]	Тип запроса	8– запрос на возврат платежа
agent_code	Нет	S[30]	Код Общества	Используется, когда у Поставщика есть другие Общества работающие по данному протоколу
pay_id	Да	S[50]	Уникальный номер платежа	Такой же, который использовался при проведении платежа
pay_date	Да	DATETIME	Дата платежа	Дата принятия платежа у клиента по часовому поясу в месте принятия платежа.
account	Да	S[100]	Идентификатор плательщика	Обычно лицевой счет клиента или телефон
pay_amount	Да	N	Сумма платежа	сумма платежа в копейках
reg_id	Да	S[50]	Идентификатор платежа у Поставщика	

Поля ответа аналогичные как у запроса проведения платежа.

Пример запроса на возврат платежа:

```
<?xml version="1.0" encoding="windows-1251"?>
<request>
  <params>
    <act>8</act>
    <pay_id>2345</pay_id>
    <pay_date>2009-04-15T11:00:12</pay_date>
    <account>54321</account>
    <pay_amount>10000</pay_amount>
    <reg_id>5432</reg_id>
  </params>
  <sign>827CCB0EEA8A706C4C34A16891F84E7B</sign>
</request>
```

Пример ответа на запрос по возврату платежа:

```
<?xml version="1.0" encoding="windows-1251"?>
<response>
  <params>
    <err_code>0</err_code>
    <err_text>Запрос принят</err_text>
  </params>
  <sign>827CCB0EEA8A706C4C34A16891F84E7B</sign>
</response>
```



Пример ответа на запрос по возврату платежа в случае ошибки:

```
<?xml version="1.0" encoding="windows-1251"?>
<response>
  <params>
    <err_code>80</err_code>
    <err_text>Операция невозможна.</err_text>
  </params>
  <sign>827CCB0EEA8A706C4C34A16891F84E7B</sign>
</response>
```

5.2.7. Проверка уникальности платежа

При получении запроса проведения платежа, Поставщик должен проверить уникальность платежа. Номер платежа *pay_id* должен быть уникальным по Обществу. При получении запроса Поставщик должен выполнить в своей базе поиск платежа с таким же *pay_id*, как у полученного в запросе.

Если такой платеж будет найден, надо сверить реквизиты *account* и *pay_amount* платежа в запросе с соответствующими реквизитами платежа в базе.

Если они совпадают, то это просто повтор запроса. В этом случае ответ Поставщика должен быть как на первый удачный запрос проведения, с теми же *reg_id* и *reg_date*, если они используются, но с кодом ошибки 1.

Если *account* или *pay_amount* различаются, то это нарушение правил учета платежей у Общества, и Поставщик должен ответить ошибкой с кодом 30. *reg_id* и *reg_date* в этом случае не передаются.

5.2.8. Правила формирования подписи

Предварительно Общество и Оператор обмениваются паролем. Используется один (одинаковый) пароль для Общества и Оператора. В качестве подписи используется MD5-хэш в виде последовательности из 32 шестнадцатеричных цифр. Регистр букв значения не имеет.

Формирование подписи в запросах Общества

К сформированному содержимому тэга *params* конкатенируется пароль. Содержимое тэга *params* берется как есть, вместе с вложенными тэгами, переносами строк. Для полученной строки генерируется подпись, которая передается в параметре *sign*.

Получив запрос, Оператор извлекает из него подстроку между тэгами *params*, генерирует для нее подпись и сверяет ее с подписью в запросе. Оператор должен обрабатывать подпись запроса в любом регистре, поэтому перед сравнением надо приводить подписи к одному регистру.

Например, для запроса

```
<?xml version="1.0" encoding="windows-1251"?>
<request>
  <params>
    <act>1</act>
    <account>758</account>
  </params>
  <sign>724870FC6BC385D7A29F4A259B6E9A6B</sign>
</request>
```

нужно делать хэш для строки "<act>1</act><account>758</account>пароль".

Формирование подписи в ответах Поставщика

К сформированному содержимому тэга *params* конкатенируются подпись полученного ранее запроса и пароль. Подпись полученного ранее запроса должна быть в том регистре, в котором получена от Общества. Для полученной строки генерируется подпись, которая передается в параметре *sign*.

Получив ответ, Общество извлекает из него подстроку между тэгами *params*, генерирует для нее подпись и сверяет ее с подписью в ответе.

Внимание! Если в запросе подписи не было, или подпись неверная, то подпись в ответе с ошибкой не передавать. (Имеется ввиду, что при этом будет только ошибка "Указаны не все необходимые параметры" или "Неверная цифровая подпись".)



5.2.9. Коды ошибок, возвращаемых Оператором

Код ошибки возвращается Оператором в тэге `err_code`.

Коды ошибок

Код ошибки	Описание ошибки
0	Успешное выполнение операции
1	Платеж уже был проведен
2	Платеж ожидает обработки у оператора
10	Запрос выполнен с неразрешенного адреса
11	Указаны не все необходимые параметры
12	Неверный формат параметров
13	Неверная цифровая подпись
20	Указанный номер счета отсутствует
21	Запрещены платежи на указанный номер счета
22	Запрещены платежи для указанной услуги
23	Запрещены платежи для указанного Общества
29	Неверные параметры платежа. В тэге <code>err_text</code> конкретное описание причины
30	Был другой платеж с указанным номером
40	Предварительная ошибка обработки платежа
41	Окончательная ошибка обработки платежа
80	Отказ на возврат платежа, в тэге <code>err_text</code> причина отказа.
90	Временная техническая ошибка
99	Прочие ошибки Поставщика. В тэге <code>err_text</code> указывается описание причины

5.2.10. Информационный реестр платежей

В информационный реестр платежей включаются все платежи, принятые Обществом для передачи Поставщику за прошедшие сутки. Платежи выбираются по дате учета Общества – `agent_date`. Реестр высылается Обществом до 12 час следующих суток по часовому поясу Общества. Реестр высылается по почте. Поставщик проводит сверку платежей, принятых в течение контрольных суток, с платежами в реестре.

Платежи, полученные Поставщиком, но отсутствующие в реестре, считаются спорными. Поставщик должен получить подтверждение таких платежей у Общества. Неподтвержденные платежи удаляются. Оператор вправе отменить спорные платежи сразу при их выявлении и должен внести их заново после подтверждения Обществом.

Платежи, присутствующие в реестре, но не полученные Оператором, считаются спорными. Оператор должен получить подтверждение таких платежей у Общества. Подтвержденные платежи заносятся Оператором.

Формат реестра

Реестр платежей представляет собой XML-файл в кодировке windows-1251.

Элементы реестра:

`registry` – корневой элемент. Имеет атрибуты:

`format` – обозначение формата реестра. Для данного формата всегда "P03".

`form_date` – дата формирования реестра по часовому поясу Общества.

`registry` содержит элементы:

`reg_date` – дата за которую реестр содержит данные.

`agent_name` – наименование организации Общества.

`prov_code` – код организации Оператора у Общества.

`prov_name` – наименование организации Оператора.

`pays` – содержит список элементов платежей.

`pays` содержит элементы:



pay – содержит данные о платежах в виде атрибутов.

Атрибуты pay:

agent_date – дата учета, для сверки. Такая же как в запросе платежа.

pay_date – дата платежа.

pay_id – номер платежа у Общества.

account – л/счет (номер договора, номер телефона).

pay_amount – сумма платежа в копейках, целое число, как и в онлайн-запросах.

serv_code – код услуги.

serv_name – наименование услуги

reg_id – номер платежа у Оператора, из ответа Оператора на запрос проведения.

note – примечание. В случае ошибки проведения, note содержит описание ошибки.

При наличии дополнительных параметров передаваемых оператору в запросе платежа, они тоже добавляются в виде атрибутов тэга pay.

Пример реестра платежей

```
<?xml version="1.0" encoding="windows-1251"?>
<registry format="P03" form_date="2011-05-13 12:00:00">
  <reg_date>2011-05-12</reg_date>
  <agent_name>ООО Общество</agent_name>
  <prov_name>ООО Оператор</prov_name>
  <pays>
    <pay agent_date="2011-05-12 11:22:33" pay_id="2345" pay_date="2011-05-12 11:00:12"
account="54321" pay_amount="10000" serv_code="123/1" serv_name="Интернет" reg_id="98765"
err_code="0" note="" />
    <pay agent_date="2011-05-12 11:22:35" pay_id="2346" pay_date="2011-05-12 11:00:17"
account="65432" pay_amount="20000" serv_code="123/1" serv_name="Интернет" reg_id="98767"
err_code="99" note="Ошибкаподключенияксерверуоператора"/>
    ...
  </pays>
</registry>
```

Имя файла итогового реестра формируется следующим образом:

bs-Код-YYYYMMDD.xml

здесь:

bs – идентификатор Общества

Код – код Поставщика у Общества

YYYY – год полностью

MM – двузначный номер месяца в году

DD – двузначный номер дня месяца

Реестр отправляется отдельным сообщением электронной почты в виде вложения.

Приложение 3. (Для программистов)

1. XML используется с ограничениями, без атрибутов (кроме реестра). Для извлечения данных нет необходимости использовать парсер, достаточно использовать функцию типа:

```
function getTagContent(string request, string tag) return string
{
    string beginTag = "<" + tag + ">";
```





```
string endTag = "</" + tag + ">";
int p1 = pos(request, beginTag); //позиция первого тэга
if (p1 > 0)
{
    p1 = p1 + length(beginTag); //смещение на начало содержимого
    int p2 = pos(request, endTag); //позиция конечного тэга
    if (p2 > p1)
        return SubString(request, p1, p2 - p1); //выбор содержимого
}
return "";
```

или регулярные выражения:

```
function getTagContent($string, $tagname)
{
    $pattern = "|<".$tagname.">(.*?)</".$tagname.">|si";
    preg_match($pattern, $string, $matches);
    return $matches[1];
}
```

Эта же функция позволяет легко взять из запроса/ответа данные для подписи. ...getTagContent(request, "params");

2. Вы можете тестировать себя из браузера. Создайте html-файл с текстовым полем, вставляйте в поле XML-запросы и отправляйте.

```
<html>
<head>
  <meta http-equiv="Content-Type" content="text/html; charset=windows-1251">
  <title>Тест протокола БС</title>
</head>
<body>
  <form name="text" action="https://ваш адрес" method="post">
    XML-запрос:
    <br> <textarea name="params" rows="20" cols="80"></textarea>
    <br> <br>
    <input type="submit" value="Отправить">
    <input type="reset" value="Очистить">
  </form>
</body>
</html>
```

Рекомендуется проверить запрос проверки на существующий номер, запрос проверки на несуществующий номер (ошибка 20), запрос платежа на существующий номер, повтор запроса платежа (ошибка 1), повтор того же запроса платежа, но с измененной суммой платежа или л/счетом (ошибка 30), запрос платежа на несуществующий номер (ошибка 20).

История изменений

Редакция	Дата	Описание изменений
----------	------	--------------------





ЦЕНТРАЛЬНАЯ
КАССА
ВСЕ ПЛАТЕЖИ ВОВРЕМЯ

Адрес

614087, Россия, г. Пермь,
ул. Малкова 12

8 (342) 240-40-22
info@ckassa.ru
ckassa.ru

3.5.8	12.09.2013	Версия протокола 3.5
3.5.9	11.08.2015	Стилистические правки
3.6.1	13.08.2015	Добавлен запрос проверки статуса платежа
3.7.3	13.04.2017	agent_date - дата получения платежа Обществом по часовому поясу Поставщика. Ошибка 80 - Отказ на возврат платежа, в тэге err_text причина отказа. 3.4 Запрос на возврат платежа. Приложение 3.



5.3. Спецификация №2

Описание технологии работы по on-line запросам платёжной системы к серверу Поставщика

Контактные данные:

Специалист	E-mail	Телефон
Общество		
Отдел поддержки клиентов	support@ckassa.ru	+7 (342) 240 38 56
Поставщик		

Требования к интерфейсу Поставщика

Интерфейс должен принимать запросы по протоколу HTTPS с/на конкретные IP-адреса Общества и Поставщика.

Интерфейс должен обрабатывать параметры, передаваемые Обществом методом GET.

Интерфейс должен формировать ответ Общества в формате XML в кодировке UTF-8 (если ответ содержит символы национальных алфавитов)

Обмен информацией ведется в режиме запрос-ответ, при этом скорость ответа не должна превышать 30 секунд, в противном случае Общество разрывает соединение по таймауту.

Параметры запросов Общества:

Параметр	Значение	Назначение	Примечание	check	payment
ACTION	Предопределенная строка. Возможные значения: check, payment,	Определяет тип запроса	check – поиск Плательщика (проверка номера); payment – создание платёжной транзакции;	+	+
ACCOUNT	Строка до 15 символов	Лицевой счет Плательщика		+	+
AMOUNT	Число	Сумма платежа	Разделитель “.” (точка).		+
PAY_ID	Число	Идентификатор платёжной транзакции	Положительное длинное целое число. Генерируется платёжной системой и используется для идентификации платёжных транзакций.		+
PAY_DATE	Дата и время	Дата и время операции в платёжной системе	Дата и время операции в платёжной системе по часовому поясу платёжной системы в формате		+



Параметр	Значение	Назначение	Примечание	check	payment
			“DD.MM.YYYY_HH24:MI:SS”. Здесь между датой и временем стоит символ “_” (подчеркивание).		

Формат ответа сервера Поставщика

Ответы сервера Поставщика возвращаются в виде XML-сообщений. Атрибут “encoding” должен иметь значение “windows-1251” и соответствует кодировке, используемой в XML-сообщении.

Атрибуты, используемые в ответах сервера Поставщика на запросы Общества

Атрибут	Значение	Назначение	Примечание	check	payment
CODE	Целое число	Код статуса операции		+	+
FIO	текст	Фамилия Плательщика		+	
ADDRESS	текст	Адрес Плательщика		+	
MESSAGE	Текст до 512 символов	Текстовое сообщение о статусе операции	Для ошибок, может содержать текстовое описание.	+	+
ACCOUNT_BALANCE	Число со знаком	Остаток на счёте Плательщика	Сумма остатка на счёте Плательщика в рублях с копейками. Разделитель целой и дробной части «.» (точка). Отрицательная сумма обозначает задолженность Плательщика.	+	
REG_DATE	Дата и время	Дата и время операции в биллинговой системе	Дата и время регистрации операции в Организации по часовому поясу Организации в формате “DD.MM.YYYY_HH24:MI:SI”. Здесь между датой и временем вставлен символ «_» (подчёркивание).		+

Значения кодов возврата (code)



Код ответа	Назначение	Примечание	check	payment
-1	Внутренняя ошибка Организации		+	+
0	Успешное завершение операции	Операция прошла успешно. Для операции проверки состояния транзакции означает, что транзакция подтверждена и платежи в системе Поставщика созданы.	+	+
2	Неизвестный тип запроса	Неизвестное значение поля "ACTION".	+	+
3	Платательщик не найден		+	+
4	Неверная сумма платежа	Недопустимое значение для поля платежа ("AMOUNT").		+
5	Неверное значение идентификатора транзакции	Недопустимое значение поля идентификатора платёжной транзакции в платёжной системе ("PAY_ID").		+
6	Неверное значение даты	Недопустимое значение поля даты платежа ("PAY_DATE").		+
8	Дублирование транзакции	Транзакция с указанным идентификатором уже была введена в систему.		+
12	Транзакция не подтверждена			

Шаблоны ответов сервера Поставщика

Ответ сервера Поставщика на запрос проверки номера (check) должен подчиняться следующему шаблону DTD

```
<?xml version="1.0" encoding="windows-1251"?>
<!DOCTYPE response [
<!ELEMENT response (CODE, MESSAGE, FIO, ADDRESS, ACCOUNT_BALANCE)>
]>
```

Ответ сервера Поставщика на запрос создания транзакции платежа (payment) должен подчиняться следующему шаблону DTD

```
<?xml version="1.0" encoding="windows-1251"?>
<!DOCTYPE response [
<!ELEMENT response (CODE, MESSAGE, REG_DATE)>
]>
```

Ответ сервера Поставщика на запрос подтверждения транзакции (commit) должен подчиняться следующему шаблону DTD





```
<?xml version="1.0" encoding="windows-1251"?>
<!DOCTYPE response [
<!ELEMENT response (CODE, MESSAGE, REG_DATE, PAY_ID_EXT*)>
] >
```

Примеры обменов сообщениями «запрос-ответ»

Поиск Плательщика по номеру телефона

Запрос	https://<host>/<path>?ACTION=check&ACCOUNT=8462333333
Ответ	<pre><?xml version="1.0" encoding="windows-1251"?> <response> <CODE>0</CODE> <MESSAGE>OK</MESSAGE> <FIO>Иванов Иван Иванович</FIO> <ADDRESS>Москва</ADDRESS> <ACCOUNT_BALANCE>-34.27</ACCOUNT_BALANCE> </response></pre>
Запрос	https://<host>/<path>?ACTION=check&ACCOUNT=24
Ответ	<pre><?xml version="1.0" encoding="windows-1251"?> <response> <CODE>3</CODE> <MESSAGE>Абонент не найден</MESSAGE> </response></pre>

Создание транзакции платежа

Запрос	https://<host>/<path>?ACTION=payment&ACCOUNT=8462333333&AMOUNT=340.24&PAY_ID=11223344&PAY_DATE=12.12.2005_12:45:18
Ответ	<pre><?xml version="1.0" encoding="windows-1251"?> <response> <CODE>0</CODE> <MESSAGE></MESSAGE> <REG_DATE>12.12.2005_12:46:03</REG_DATE> </response></pre>
Запрос	https://<host>/<path>?ACTION=payment&ACCOUNT=8462333333&TYPE=15&AMOUNT=340.24&PAY_ID=11223344&PAY_DATE=12.12..2005_12:45:18
Ответ	<pre><?xml version="1.0" encoding="windows-1251"?> <response> <CODE>6</CODE> <MESSAGE>Не верное значение даты платежа</MESSAGE> </response></pre>

Адреса серверов Общества

Основной адрес - *agent.bisys.ru*



5.4. Спецификация №3

Описание технологии пополнения баланса в режиме on-line

Контактные данные:

Специалист	E-mail	Телефон
Общество		
Отдел поддержки клиентов	support@ckassa.ru	+7 (342) 240 38 56
Поставщик		

Требования к интерфейсу Поставщика:

Интерфейс должен принимать запросы по протоколу HTTPS с/на конкретные IP-адреса Общества и Поставщика с использованием публичного сертификата сервера:

Для авторизации Общества в процессинговой системе Поставщика при регистрации платежа используются логин и пароль.

Интерфейс должен обрабатывать параметры, передаваемые Обществом, методом GET

Интерфейс должен формировать ответ Общества в формате XML в кодировке UTF-8 (если ответ содержит символы национальных алфавитов)

Обмен информацией ведется в режиме запрос-ответ, при этом скорость ответа не должна превышать 60 секунд, в противном случае, Общество разрывает соединение по таймауту.

Если предполагаемое количество платежей за услуги подключаемого Поставщика, ожидается достаточно интенсивным (до 10 платежей в минуту и более), желательно, чтобы интерфейс спокойно переносил многопоточную коммуникацию до 10-15 одновременных соединений.

Основные принципы работы интерфейса

Каждый платеж Общества имеет уникальный идентификатор, который передается Поставщику в переменной `txn_id` – целое число длиной до 20 знаков. По этому идентификатору производится дальнейшая сверка взаиморасчетов и решение спорных вопросов.

Сумма платежа принимается от Плательщика и передается Поставщику в рублях в переменной `sum` – дробное число 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»

В запросе на добавление платежа, Общество передает дату платежа (под датой платежа в системе подразумевается дата получения запроса от клиента) в переменной `txn_date` – дата в формате ГГТТММДДЧЧММСС. Эту дату необходимо использовать для проведения бухгалтерских взаиморасчетов. Так как у Общества учет платежей ведется по дате получения запроса от Плательщика, то и расчеты с Поставщиком необходимо вести по этой дате.

Например:

Плательщик прислал Обществу запрос 31.12.2016 в 23:59:59, учитывая задержку на обработку данных и пересылку информации по каналам связи, Общество смог отправить запрос Поставщику 01.01.2017 00:00:05, соответственно, платеж будет учтен в системе Поставщика в другом отчетном периоде, что вызовет некоторые проблемы при проведении сверок. Чтобы избежать такой ситуации, Общество передает Принциалу дату, в которой нужно учитывать платеж.

Поставщик идентифицирует своего Плательщика по уникальному идентификатору (Идентификатор Плательщика). Перед отправкой Поставщику, Идентификатор проходит проверку корректности в



соответствии с регулярным выражением, которое должен предоставить Поставщик. Идентификатор Плательщика передается в переменной account – строка, содержащая буквы, цифры и спецсимволы, длиной до 200 символов.

Оплата услуг Поставщика производится системой в 2 этапа – проверка состояния Плательщика и непосредственно проведение платежа. Тип запроса передается Обществом в переменной command – строка, принимающая значения «check» и «pay». При проверке статуса (запрос «check»), Поставщик должен проверить наличие в своей базе абонента с указанным Идентификатором и выполнить внутренние проверки Идентификатора, суммы платежа в соответствии с принятой логикой пополнения лицевых счетов через платежные системы. При проведении платежа (запрос «pay»), Поставщик должен произвести пополнение баланса Плательщика.

В случае если любой из запросов Поставщику завершается ошибкой, то Поставщик возвращает код ошибки в соответствии с таблицей, приведенной ниже. Все ошибки имеют признак фатальности. Для Общества фатальная ошибка означает, что повторная отправка запроса с теми же параметрами, приведет к 100% повторению той же ошибки – следовательно, Общество прекращает обработку запроса и завершает его с ошибкой. Нефатальная ошибка означает для Общества, что повторение запроса с теми же параметрами через некоторый промежуток времени, возможно, приведет к успеху. Общество будет повторять запросы, завершающиеся нефатальной ошибкой, постоянно увеличивая интервал, пока операция не завершится успехом или фатальной ошибкой, либо пока не истечет срок жизни запроса – 24 часа. Отсутствие связи с сервером Поставщика является нефатальной ошибкой. Отсутствие в ответе элемента <result> (некорректный XML, страница Servicetemporarilyunavailable и т.д.) – является фатальной ошибкой. Запросы получают отказ с ошибкой 300 – Другая ошибка Поставщика.

В базе Поставщика не должно содержаться двух успешно проведенных платежей с одним и тем же номером txn_id. Если Общество повторно присылает запрос с уже существующим в базе Поставщика txn_id, то Поставщик должен вернуть результат обработки предыдущего запроса.

Поставщик возвращает ответ на запросы Общества в формате XML со следующей структурой:

```
<?xml version="1.0" encoding="UTF-8"?>
```

```
<response>
```

```
  <osmp_txn_id></osmp_txn_id>
```

```
  <prv_txn></prv_txn>
```

```
  <sum></sum>
```

```
  <result></result>
```

```
  <comment></comment>
```

```
</response>
```

<response> - тело ответа

<osmp_txn_id> - номер транзакции в системе Общества, который передается Поставщику в переменной txn_id.

<prv_txn> - уникальный номер операции пополнения баланса Плательщика (в базе Поставщика), целое число длиной до 20 знаков. Этот элемент должен возвращаться Поставщикам после запроса на пополнение баланса (запроса «pay»). При ответе на запрос на проверку состояния Абонента (запрос «check») его возвращать не нужно – он все равно не обрабатывается.

<sum>- сумма платежа, передаваемая Поставщику, дробное число 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»

<result> - код результата завершения запроса.

<comment> - необязательный элемент – комментарий завершения операции.



Список кодов завершения

При обработке запросов от системы Общества, Поставщик 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.

Код	Комментарий	Фатальность
0	ОК	
1	Временная ошибка. Повторите запрос позже	
4	Неверный формат идентификатора Плательщика	+
5	Идентификатор Плательщика не найден (Ошиблись номером)	+
7	Прием платежа запрещен Поставщиком	+
8	Прием платежа запрещен по техническим причинам	+
79	Счет Плательщика не активен	+
90	Проведение платежа не окончено	
241	Сумма слишком мала	+
242	Сумма слишком велика	+
243	Невозможно проверить состояние счета	+
300	Другая ошибка Поставщика	+

Пример запроса на проверку состояния счета Плательщика

Платежное приложение Поставщика payment_app.cgi, располагается по адресу service.someprv.ru, сервер поддерживает HTTPS соединения на порт 8443. Для проверки состояния Плательщика, Общество генерирует запрос следующего вида:

https://service.someprovider.ru:8443/payment_app.cgi?command=check&txn_id=1234567&account=4957835959&sum=10.45

Строка запроса содержит переменные:

command=check – запрос на проверку состояния Плательщика
 txn_id=1234567 – внутренний номер платежа в системе Общества
 account=4957835959 – идентификатор Плательщика в информационной системе Поставщика
 sum=10.45 – сумма к зачислению на лицевой счет Плательщика

Ответ Поставщика 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>0</result>
```



```
<comment>account exists</comment>  
</response>
```

Возвращение result=0 на запрос «check» свидетельствует о том, что лицевой счет Плательщика с соответствующим ему номером osmp_txn_id может быть пополнен на сумму, указанную в запросе. После успешной проверки состояния счета Плательщика система переходит к формированию и отправке запроса на пополнение баланса (запрос «pay»).

Пример запроса на пополнение лицевого счета

```
https://service.someprovider.ru:8443/payment_app.cgi?command=pay&txn_id=1234567&txn_date=20050815120133&account=4957835959&sum=10.45
```

Запрос содержит переменные:

command=pay	– запрос на пополнение баланса Плательщика
txn_id=1234567	– внутренний номер платежа в системе Плательщика
txn_date=20050815120133	– дата учета платежа в системе Общества
account=4957835959	– идентификатор Плательщика в информационной системе Поставщика
sum=10.45	– сумма к зачислению на лицевой счет Плательщика

Пример ответа:

```
<?xml version="1.0" encoding="UTF-8"?>  
<response>  
<osmp_txn_id>1234567</osmp_txn_id>  
<prv_txn>2016</prv_txn>  
<sum>10.45</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>10.45</sum>  
<result>0</result>  
<comment>OK</comment>  
</response>
```

Возвращая result=0 на запрос «pay», Поставщик сообщает об успешном завершении операции пополнения баланса. Система полностью завершает обработку данной транзакции.

Для передачи дополнительных параметров, таких как баланс, ФИО, и т.д., вам надо добавить в ответ на запрос проверки тэг <bisys_params>, в котором тэги с нужными параметрами. Названия тэгов можете выбрать сами. Обычные <client_name> и <balance>.

Сумма баланса передается в рублях, копейки отделяются точкой.

Например:

```
<?xml version="1.0" encoding="UTF-8"?>
```



```
<response>
  <result>0</result>
  <osmp_txn_id>1001</osmp_txn_id>
  <comment></comment>
  <bisys_params>
    <client_name>Иванов</client_name>
    <balance>-200.00</balance>
  </bisys_params>
```

ПОРЯДОК ИСПОЛЬЗОВАНИЯ SSL-СЕРТИФИКАТОВ

1. Поставщик предоставляет Обществу корневой сертификат Поставщика (для обеспечения доверия сертификатам, выданным Центром сертификации Поставщика) в виде, пригодном для установления его принадлежности Поставщику, то есть в виде base-64 кодированного файла формата PKCS#10 и на бумажном носителе, заверенном собственноручной подписью руководителя и оттиском печати Поставщика.

2. Полученный сертификат Поставщика вводится Обществом в действие, не позднее следующего рабочего дня после его предоставления Поставщиком на бумажном носителе.

3. При истечении срока действия корневого сертификата Поставщика, Поставщик не позднее, чем за 5 (пять) рабочих дней до окончания срока действия активного корневого сертификата предоставляет Обществу новый корневой сертификат в соответствии с п.1.

4. При компрометации или подозрении на компрометацию закрытого ключа сертификата Поставщика (т.е. при ознакомлении или подозрении на ознакомление неуполномоченного лица с закрытым ключом сертификата, а также при несанкционированном использовании или подозрении на несанкционированное использование закрытого ключа сертификата) Общество извещается в простой письменной форме о прекращении действия указанного сертификата. С момента уведомления Поставщик прекращает электронный документооборот с Обществом с использованием указанного сертификата;

5. Общество, после получения сообщения о компрометации сертификата, выводит соответствующий сертификат из действия в максимально короткие сроки, но не позднее следующего рабочего дня после получения сообщения о компрометации;

6. Клиент имеет право в любое время производить замену собственных сертификатов.

Адреса серверов платежной системы Общества

Основной адрес – *agent.bisys.ru*



6. РАЗМЕЩЕНИЕ ПЛАТЕЖНОГО СЕРВИСА В ПРИЛОЖЕНИИ

Терминология:

CardToken - уникальный идентификатор карты, выдается сервером после регистрации карты.

MerchantToken - уникальный идентификатор получателя электронно-денежных средств с использованием сервиса «Ckassa Wallet, выдается сервером после регистрации получателя.

ShopSecKey - уникальный ключ, используется для подписи сообщений (клиентом и сервером), выдается в Личном кабинете.

ShopToken - уникальный идентификатор организации, выдается в Личном кабинете.

UserToken - уникальный идентификатор плательщика, выдается сервером после регистрации плательщика.

Анонимный платеж – платеж, созданный без сохранения данных банковской карты. Заморозка средств у плательщика (холд) - предварительная блокировка денежных средств плательщика на сумму заказа на срок до 5 (пяти) дней. Списание производится с обязательным подтверждением оказания услуг.

Кошелек - электронное средство платежа «Ckassa Wallet», которое предназначено для осуществления Участником операций, проводимых в соответствии с Офертой об осуществлении перевода электронных денежных средств (<https://ckassa.ru/offer-ckk.html>)

Плательщик – физическое лицо, осуществляющее внесение денежных средств в целях исполнения денежных обязательств физического лица перед Поставщиком, с которым заключен Клиентский Договор, и оплачивающий Услуги, оказываемые Поставщиком, через Пункты приема платежей.

Получатель - В зависимости от типа подключения может выступать Поставщик либо Кошелек «Ckassa Wallet»

Подтверждение проведения платежа с использованием технологии 3-D Secure – операция с применением 3-D Secure - протокола, который используется как дополнительный уровень безопасности при осуществлении онлайн-платежей с банковских карт. Добавляет ещё один шаг аутентификации при совершении онлайн-платежей.

Рекуррентный платеж - платеж, созданный по ранее сохраненным данным банковской карты.

6.1. Типы подключения

6.1.1. Тип подключения «Магазин. Анонимные платежи»

Данный тип подключения подходит для оплаты товаров и услуг и интернет-магазинов, использующих способ приема платежей без сохранения банковской карты.

6.1.2. Тип подключения «Магазин. Рекуррентные платежи»

Данный тип подключения подходит для оплаты товаров и услуг и интернет-магазинов, использующих способ рекуррентных платежей с сохранением банковской карты в системе.

6.1.3. Тип подключения «Поездка. Оплата на р/с»

Данный тип подключения подходит для служб такси, в ситуации оплаты пассажиром поездки с банковской карты с зачислением на расчетный счет службы такси с заморозкой денежных средств.

6.1.4. Тип подключения «Поездка. Оплата на р/с и/или кошелек»



Данный тип подключения подходит для служб такси, в ситуации оплаты пассажиром поездки с банковской карты с зачислением на кошелек «Центральная касса» водителю или расчетный счет службы такси с заморозкой денежных средств

Тип подключения				
Доступные опции	Магазин. Анонимные платежи	Магазин. Рекуррентные платежи	Поездка. Оплата на р/с	Поездка. Оплата на р/с и/или кошелек
Создание анонимного платежа	+	+	-	-
Для рекуррентных платежей				
Рекуррент с 3ds	-	+	+	+
Рекуррентный платеж без с 3ds	-	+	+	+
Параметр enableSMSConfirm	-	+	+	+
Регистрация плательщика	-	+	+	+
Статус плательщика	-	+	+	+
Регистрация карты	-	+	+	+
Получение списка карт	-	+	+	+
Деактивации/удаление карты	-	+	+	+
Создание рекуррентного платежа	-	+	+	+
Уведомление о регистрации карты	-	+	+	+
Для оплаты на кошелек				
Регистрация получателя	-	-	-	+
Статус получателя	-	-	-	+
Создание платежа в пользу получателя	-	-	-	+
Создание платежа в пользу получателя, оплата с баланса сотового телефона	-	-	-	+
Получение баланса получателя	-	-	-	+
Заморозка средств у плательщика (холд)				
Заморозка суммы заказа	-	-	+	+
Обновить получателя забронированных средств	-	-	+	+
Подтверждение оказания услуги (заморозки)	-	-	+	+
Отмена замороженных/забронированных средств	-	-	+	+

Таблица 1. Набор опций по типам подключения



6.2. Технические параметры размещения платежного сервиса в Приложении

[Общее описание](#)

[Алгоритм работы](#)

[Подключение](#)

[Формирование подписи](#)

[Получение статуса платежа](#)

[Уведомление об оплате](#)

[Примеры](#)

[Список ошибок](#)

[Коды ошибок проводки платежа](#)

[Для анонимных платежей](#)

[Создание анонимного платежа](#)

[Для рекуррентных платежей](#)

[Обычный платеж](#)

[Рекуррентный платеж](#)

[Рекуррент с 3ds](#)

[Параметр enableSMSConfirm](#)

[Регистрация пользователя](#)

[Статус пользователя](#)

[Регистрация карты](#)

[Получение списка карт](#)

[Деактивации/удаление карты](#)

[Создание рекуррентного платежа](#)

[Уведомление о регистрации карты](#)

[Список ошибок для рекуррентных платежей](#)

[Коды ошибок проводки рекуррентных платежей](#)

[Для оплаты на кошелек](#)

[Регистрация мерчанта](#)

[Статус мерчанта](#)

[Создание платежа в пользу мерчанта](#)

[Создание платежа в пользу мерчанта, оплата с баланса сотового телефона](#)

[Получение баланса мерчанта](#)

[Заморозка средств у плательщика \(холд\)](#)

[Бронирование суммы](#)

[Обновить получателя забронированных средств](#)

[Подтверждение оказания услуги \(заморозки\)](#)

[Отмена замороженных/забронированных средств](#)



Общее описание

Алгоритм работы

- 1) Создание платежа
- 2) Оплата платежа пользователем (ввод данных карты, прохождение 3ds)
- 3) Получение уведомления об оплате

Оплата считается совершенной (можно оказывать услугу), если получен статус платежа 'Оплачен':

- в уведомлении об оплате (см. пункт 'Уведомление об оплате')
- при запросе статуса платежа (см. пункт shopAPI 'Получение статуса платежа')

Подключение

Подключение host to host.

Обмен сообщениями производится в формате JSON.

Protocol не ниже: TLSv1.2

При подключении обязательно использование сертификата клиента, для подписи сообщений серверу.

Сертификат выдается по запросу.

Тестовое

Тестовый url: <https://tst.autopays.ru/api-shop>

Тестовая карта:

400000000000000002

exp 01/21

Боевое

url: <https://api.autopays.ru/api-shop/rs/shop>

Рекомендуемые таймауты

На создание платежа 1 минута.

Формирование подписи

Строка для вычисления подписи составляется из всех полей запроса (если не указано обратное)

Для запросов: без shopToken, разделенных '&' + postfix.

Для ответов: разделенных '&' + postfix.

Порядок полей как в описании протокола.

Postfix:



Для запросов: shopToken + '&' + shopSecKey

Для ответов: shopSecKey

Если параметр отсутствует (значение null), то он не участвует в подписи.

Sign в формировании подписи не участвует.

md5(md5(param1+'&'+param2+'&'+param3+'&'+...+paramN+'&'+postfix).ToUPPER()).ToUPPER()

ToUPPER() - перевод строки в верхний регистр

Результат записывается в параметр: sign

Если подпись не верна в ответ придет сообщение об ошибке:

400 Bad Request {"message": "Incorrect signature", "userMessage": "Не корректная подпись", "code": 2700}

Получение статуса платежа

Отправка уведомлений производится по https.

Делается несколько попыток отправки уведомления с нарастающим интервалом.

Возможны повторные уведомления для одного платежа, для таких случаев ожидается успешный ответ.

POST запрос от ckassa до сервера организации

POST запрос

URL: /check/payment/state

При оплате рекуррентом с обязательным вводом 3ds см. пункт 'Рекуррент с 3ds' shopAPI, в ответ будет получено исключение: NeedPass3dsException (код 2365, см. Список ошибок).

Для прохождения 3ds проверки необходимо перенаправить пользователя по ссылке в параметре: securePageURL (метод GET).

Параметры запроса:

regPayNum*: номер платежа (выдается после успешного создания платежа)

shopToken*: идентификатор организации

sign*: подпись

****** - обязательный параметр, не указывается для анонимных платежей

Параметры ответа:

state**: статус платежа

totalAmount*: сумма платежа в копейках

createdDate*: время создания платежа

providerServCode*: уникальный код провайдера услуг



providerName*: название провайдера услуг

errorCode: код ошибки, см. пункт "Коды ошибок проводки платежа"

Носит информативный характер, используется для идентификации проблемы.

Не может использоваться для определения успешности/не успешности платежа, для этого необходимо использовать параметр: **state** ('статус платежа').

error: текст ошибки (используется для идентификации проблемы, не должно выдаваться пользователю)

message: сообщение пользователю (ошибка, пояснение...)

provisionServices: флаг, была ли оказана услуга (true или false)

procDate: время проведения платежа (некоторые платежи могут проводиться с задержкой по времени)

sign*: подпись

* - обязательные параметры.

Время в формате: уууу-ММ-дд НН:мм:сс.

****Статус платежа:**

created_error - ошибка при создании платежа

created - платеж создан, но не оплачен

rejected - платеж отменен (при оплате не рекурентом, возможен переход в статус payed, например, при повторном вводе данных карты)

refunded - оплата возвращена

Платеж оплачен и ожидает исполнения (можно оказывать услугу):

payed - платеж оплачен

holded - средства заморожены

processed - платеж оплачен и успешно проведен

error - ошибка при проведении

Платеж оплачен, но при проведении возникла ошибка

Пример запроса:

```
{"sign":"F6217B7EE96969ECCDA81ABBB973E8C6","shopToken":"SHOP_TOKEN","regPayNum":"1310958041"}
```

Пример ответа:

```
{'sign':'633411CA7D9EDEF678423898C8E5F399','errorCode':'1001','provisionServices':'true',totalAmount:'100000','serviceCode':'serviceCode','providerName':'providerName','state':'payed','createdDate':'2012-10-06 03:02:01','error':'error','message':'message','procDate':'2015-01-05 12:14:18'}
```

Уведомление об оплате

По умолчанию отправляются только успешные платежи.

При не успешном создании платежа уведомление не отправляется.



URL: {cb url}

Параметры запроса:

regPayNum*: номер платежа в системе ckassa

amount*: сумма платежа в копейках

comission*: сумма комиссии в копейках

state**: статус платежа

errorCode: код ошибки

errorMsg: текст ошибки

shopToken*: идентификатор организации

sign*: подпись

* - обязательный параметр

В ответ ожидается любой ответ со статусом 200

****Статус платежа:**

created_error - ошибка при создании платежа

created - платеж создан, но не оплачен

rejected - платеж отменен (при оплате не рекурентом, возможен переход в статус payed, например, при повторном вводе данных карты)

refunded - оплата возвращена

Платеж оплачен и ожидает исполнения (можно оказывать услугу):

payed - платеж оплачен

holded - средства заморожены

processed - платеж оплачен и успешно проведен

error - ошибка при проведении

Платеж оплачен, но при проведении возникла ошибка

Пример запроса:

```
{"sign":"XXXXXXXXX","shopToken":"SHOP_TOKEN","regPayNum":"1310958041","amount":"100000","comission":"1000","state":"payed"}
```

Примеры

Создание платежа:

Запрос:



`md5(md5(serviceCode+'&'+userToken+'&'+amount+'&'+comission+'&'+cardToken+'&'+payType+'&'+clientType+'&'+properties[0].name+'&'+properties[0].value+'&'+...+properties[N].name+'&'+properties[N].value+'&'+shopToken+'&'+shopSecKey).ToUPPER()).ToUPPER()`

`md5(md5('109-5804-1'+&'+USER_TOKEN'+&'+10000'+13300'+&'+CARD_TOKEN'+&'+card'+&'+web'+&'+Л/ЧЕТ'+&'+9503006477'+&'+ФИО'+&'+Иванов Н П'+&'+АДРЕС'+&'+Ленина10'+&'+МЕСЯЦ'+&'+05.2015'+&'+СУММА_ПЕНИ'+&'+10000'+&'+SHOP_TOKEN'+&'+SEC_KEY'`

В результате sign: "AB6291B7642B14885D1A3E40038C683F"

Ответ:

`md5(md5(userToken+'&'+shopToken+'&'+regPayNum+'&'+methodType+'&'+payUrl+'&'+shopSecKey).ToUPPER()).ToUPPER()`

`md5(md5('USER_TOKEN'+&'+SHOP_TOKEN'+&'+1234567890'+&'+GET'+&'+https://autopays.ru/payment/#!se arch\_payment+'SEC_KEY').ToUPPER()).ToUPPER()`

В результате sign: "C622C723C2A4266340E4E87AADD6B82D"

Получение статуса платежа:

Запрос:

`md5(md5(regPayNum+'&'+shopToken+'&'+shopSecKey).ToUPPER()).ToUPPER()`

`md5(md5('1310958041'+&'+SHOP_TOKEN'+&'+SEC_KEY').ToUPPER()).ToUPPER()`

В результате sign: "F6217B7EE96969ECCDA81ABBB973E8C6"

Ответ:

`md5(md5(state+'&'+totalAmount+'&'+createdDate+'&'+providerServCode+'&'+providerName+'&'+error+'&'+message+'&'+procDate+'&'+shopSecKey).ToUPPER()).ToUPPER()`

`md5(md5('payed'+&'+100000'+&'+2012-10-0603:02:01'+&'+serviceCode'+&'+providerName'+&'+error'+&'+message'+&'+2015-01-0512:14:18'+&'+SEC_KEY').ToUPPER()).ToUPPER()`

В результате sign: "98CBD963E52F64A2736BF5055B2E5F9B"

Список ошибок

Ответ на запрос: 400 Bad Request.

Если в ответе присутствует параметр sign, необходимо проверять подпись сообщения.

Алгоритм формирования подписи аналогичен, см. Формирование подписи.

Авторизация:

- code:2001 {"message":"Incorrect request param {0}","userMessage":"Некорректный параметр запроса {0}","code":2001}

- code:2700 {"message":"Incorrect signature","userMessage":"Не корректная подпись","code":2700}



- code 2701 {"message":"Unknown shop token","userMessage":"Не известный идентификатор магазина","code":2701}
- code 2703 {"message":"User not registered for shop","userMessage":"Пользователь не зарегистрирован для данного магазина","code":2703}
- code 2709 {"message":"Merchant not registered for shop","userMessage":"Мерчант не зарегистрирован для магазина","code":2709}

Получение списка услуг:

- code 1263 {"message":"Can't pay for number","userMessage":"Невозможно оплатить данный номер","code":1263}

Активация:

- code 2350 {"message":"Activate card error","userMessage":"Ошибка привязки карты","code":2350}
- code 2356 {"message":"Incorrect sms code","userMessage":"Неверный SMS код","code":2356}
- code 2600 {"message":"SMS to number {0} already sent","userMessage":"Отправили СМС на номер {0}","code":2600, "canSendNewSMSThrough":60000}

canSendNewSMSThrough - время в миллисекундах до следующей попытки отправить SMS

Проверка статуса платежа:

- code 2365 {"message":"For recurrent need pass 3ds":"Необходимо ввести 3ds код","code":2365,"regPayNum":"1234567890","securePageURL":"<https://ya.ru>"}
- code 2704 {"message":"User not owned payment","userMessage":"Платеж не принадлежит пользователю","code":2704}
- code 2705 {"message":"Shop not owned payment","userMessage":"Платеж не принадлежит магазину","code":2705}

Создание платежа:

- code 1295 {"message":"Can't create payment, please try later","userMessage":"В данный момент принять платеж невозможно, попробуйте позже","code":1295}
- code 2354 {"message":"Create recurrent payment error. Try other car","userMessage":"Ошибка проведения платежа. Попробуйте другую карту"}
- code 2360 {"message":"Can't create recurrent payment","userMessage":"Мы сожалеем, но вы не можете совершить этот платеж привязанной картой из соображений вашей безопасности"}

Общие:

- code:1300 {"message":"Provider not found","userMessage":"Оператор не найден","code":1300}
- code:2702 {"message":"Provider forbidden for shop","userMessage":"Поставщик запрещен для данного магазина","code":2702}
- code:2706 {"message":"Incorrect user state:{0}","userMessage":"Не корректное состояние пользователя","code":2706}

Остальные ошибки:

- code:1262 {"message":"Unknown error, try later","userMessage":"Непредвиденная ошибка, попробуйте позже","code":1262}



Приведенный список основных ошибок может быть дополнен.

Коды ошибок проводки платежа

Носит информативный характер, используется для идентификации проблемы.

Не служит для определения успешности/не успешности платежа, для этого необходимо использовать параметр: **state** ('статус платежа'), см. пункт 'Получение статуса платежа'.

Полученные от эквайринга*:

- 101: **SystemError**, "Системная ошибка. Обратитесь к эквайеру"
- 102: **TemporaryError**, "Временный отказ"
- 103: **BankInteractionError**, "Ошибка взаимодействия с банком"
- 104: **AFSBlocked**, "Операция отклонена"
- 105: **UnsupportedSecurityAction**, "Запрошен 3DS для операции, в которой его использование невозможно"
- 106: **UnexpectedBankResponse**, "Неожиданный ответ банка"
- 107: **AcqAccessDenied**, "Магазину доступ запрещен"
- 108: **AcqIncorrectData**, "Не известный тип заказа, номер заказа или еще что-то в этом духе"
- 109: **ParallelProcessing**, "Попытка параллельной оплаты"
- 110: **ProcessingError**, "Ошибка, связанная с обработкой конкретного заказа"
- 111: **AuthFailed**, "Ошибки авторизации запроса (подпись...)"
- 112: **UnknownCard**, "Неизвестная карта. Необходимо заново зарегистрировать карту"
- 113: **UnknownOrder**, "Неизвестный заказ"
- 201: **ChannelTransactionDeclined**, "Эквайер отклонил транзакцию"
- 202: **ChannelSecurityCheckFailed**, "Эквайер отклонил транзакцию"
- 203: **ChannelCardExpired**, "Действие карты истекло"
- 204: **ChannelCardDeclined**, "Банк отклонил транзакцию"
- 205: **ChannelInsufficientFunds**, "Недостаточно средств"
- 206: **ChannelCardBlocked**, "Карта заблокирована"
- 207: **ChannelNotPermitted**, "Эквайер отклонил транзакцию"
- 208: **ChannelTemporaryDecline**, "Эквайер временно отклонил транзакцию"
- 209: **Channel3DSFailed**, "Эквайер отклонил транзакцию"

Общие:

- 1295: **CreatePaymentFailed**, "Не удалось создать платеж"

Список кодов может быть дополнен.



Для анонимных платежей

Создание анонимного платежа

Вызывается после успешной проверки возможности создания платежа.

В ответ приходит номер платежа, url оплаты и способ перехода.

POST запрос

URL: /do/payment/anonymous

Параметры запроса:

serviceCode*: код провайдера

amount*: сумма платежа в копейках (сумма которая идет на счет пользователю)

comission*: комиссия платежа в копейках (при отсутствии комиссии передается 0)

cardToken: идентификатор карты (для рекуррентного платежа), см. пункт: 'Получение списка карт'

payType: способ оплаты

Возможные значения:

- card - оплата банковской картой (по умолчанию)
- yd - оплата яндекс деньгами
- sms - оплата с баланса мобильного телефона (см параметр userPhone)

clientType: тип клиента (устанавливает вид формы оплаты)

Возможные значения:

- web - веб форма (по умолчанию)
- mobile - мобильная форма
- iframe - фрейм

userPhone**: для СМС платежа номер телефона с которого будет произведена оплата.

С кодом страны без '+', например: 79020000000

userEmail***: e-mail для отправки фискального чека

fiscalType***: нужно или нет отправлять фискальный чек

Возможные значения:

- email
- отправить фискальный чек на почту
- none (по умолчанию)
- фискальный чек не нужен

properties*: массив реквизитов.



Должны быть заполнены все обязательные поля. Заполненные не обязательные поля должны иметь корректные значения.

'name' - название, 'value' - значение.

shopToken*: идентификатор организации

sign*: подпись

Параметры ответа:

shopToken*: идентификатор организации

regPayNum*: номер платежа

methodType*: метод перехода по url для оплаты (GET/POST)

payUrl*: url оплаты

sign*: подпись

* - обязательные параметры.

** - для СМС платежа обязательно наличие номера телефона (payType = sms)

*** - если fiscalType не none, обязательно наличие соответствующего параметра

Пример запроса:

```
{"sign":"71308F2268DB7F9A51ADD93E2CD57A76","clientType":"web","payType":"card","shopToken":"SHOP_T  
OKEN","properties":[{"name":"Л/СЧЕТ","value":"9503006477"}, {"name":"ФИО","value":"Иванов Н  
П"}, {"name":"АДРЕС","value":"Ленина  
10"}, {"name":"МЕСЯЦ","value":"05.2015"}, {"name":"СУММА_ПЕНИ","value":"10000"}], "serviceCode":"109-  
5804-1","amount":"10000","comission":"13300"}
```

Пример ответа:

```
{"sign":"C622C723C2A4266340E4E87AADD6B82D","regPayNum":"1234567890","methodType":"GET","payUrl":"https://  
autopays.ru/payment/#!search_payment","userToken":"USER_TOKEN","shopToken":"SHOP_TOKEN"}
```

Для рекуррентных платежей

Обычный платеж

- 1) Создание платежа (используется UserToken)
- 2) Оплата платежа пользователем (ввод данных карты, прохождение 3ds)

Рекуррентный платеж

Подготовка:

- 1) Регистрация карты
- 2) Получение списка карт (получение CardToken)



Для каждого платежа:

- 3) Создание рекуррентного платежа (используется UserToken + CardToken)
- 4) Оплата платежа пользователем (ввод данных карты, прохождение 3ds)

Рекуррент с 3ds

Для некоторых рекуррентных платежей не возможна оплата без 3ds.

При этом будет сгенерировано исключение NeedPass3dsException (код 2365, см. Список ошибок), которое содержит номер платежа - regPayNum и securePageURL.

Url на который необходимо перенаправить пользователя для прохождения 3ds (метод GET)

Исключение может быть получено на этапе создания рекуррента (пункт 'Создание платежа') и проверки статуса платежа (пункт 'Получение статуса платежа').

Логика работы задается параметром enableSMSConfirm.

Параметр enableSMSConfirm

Используется только для рекуррентных платежей.

Может ли пользователь участвовать в проведении платежа (прохождение 3ds, подтверждение по СМС)

Задается при создании платежа, см пункт 'Создание платежа'

- **false** (по умолчанию)

Пользователь не участвует в проводке платежа, платеж не создается, будет сгенерировано исключение: CreatePaymentException (код 1295), либо CantDoRecurrent(код 2360)..., см. Список ошибок, Shop API

- **true**

Пользователь участвует в проводке платежа. Создается платеж который необходимо подтвердить, будет сгенерировано исключение:

а) NeedPass3dsException (код 2365, см. Список ошибок).

Необходимо пройти 3ds, перенаправив пользователя по ссылке в параметре: securePageURL (метод GET).

Регистрация пользователя

Попытка повторной регистрации не считается ошибкой, и возвращает данные ранее зарегистрированного пользователя.

POST запрос

URL: /user/registration

Параметры запроса:



login*: номер телефона (с кодом, но без '+')

Например 79020000000.

email: e-mail (отправляются чеки после оплаты)

name: имя

surName: фамилия

middleName: отчество

shopToken*: идентификатор организации

sign*: подпись

Параметры ответа:

login*: номер телефона

email: e-mail

name: имя

surName: фамилия

middleName: отчество

state*: статус

active - активный (для данного пользователя доступны все операции)

Значение по умолчанию.

Пользователь может отозвать свое согласие в ЛК autopays (тогда необходимо повторно получить его согласие на работу, см. пункт 'Активация пользователя').

inactive - не активный (операции не доступны, требуется активация см. пункт 'Активация пользователя')

disable - заблокирован (операции не доступны, активация не возможна,

например: данный пользователь отказался от обработки своих данных сторонней организацией)

userToken*: идентификатор пользователя

sign*: подпись

* - обязательные параметры.

Часть дополнительных параметров может быть изменена/дополнена сервером, например, если пользователь ранее был зарегистрирован в системе autopays.

Пример запроса:

```
{"sign":"4A7F7A58BBB46D9B128F7DC4E9215C2A","login":"Login","email":"Email","name":"Name","surName":"SurName","middleName":"MiddleName","shopToken":"SHOP_TOKEN"}
```

Пример ответа:

```
{"sign":"A97650D73FD1EF71FBA67F5BBD3ECAD8","state":"active","login":"getLogin()","email":"getEmail()","name":"getName()","surName":"getSurName()","middleName":"getMiddleName()","userToken":"USER_TOKEN"}
```



Статус пользователя

Если пользователь не зарегистрирован будет сгенерировано исключение UserNotRegisteredForShop (код 2703, см. Список ошибок)

POST запрос

URL: /user/status

Параметры запроса:

login*: номер телефона (с кодом, но без '+')

Например 79020000000.

shopToken*: идентификатор организации

sign*: подпись

Параметры ответа:

login*: номер телефона

email: e-mail

name: имя

surName: фамилия

middleName: отчество

state*: статус

active - активный (для данного пользователя доступны все операции)

Значение по умолчанию.

Пользователь может отозвать свое согласие в ЛК autopays (тогда необходимо повторно получить его согласие на работу, см. пункт 'Активация пользователя').

inactive - не активный (операции не доступны, требуется активация см. пункт 'Активация пользователя')

disable - заблокирован (операции не доступны, активация не возможна,

например: данный пользователь отказался от обработки своих данных сторонней организацией)

userToken*: идентификатор пользователя

sign*: подпись

* - обязательные параметры.

Пример запроса:

```
{"sign":"2ADD30013A2E5DD4FF870EC2833564DF","login":"79150000000","shopToken":"SHOP_TOKEN"}
```

Пример ответа:



```
{"sign":"A97650D73FD1EF71FBA67F5BBD3ECAD8","state":"active","login":"getLogin()","email":"getEmail()","name":"getName()","surName":"getSurName()","middleName":"getMiddleName()","userToken":"USER_TOKEN"}
```

Регистрация карты

При регистрации карты будет списано до 5 рублей, после добавления карты деньги будут возвращены.

После подтверждения оплаты карта появится в списке и станет доступной для создания рекуррентных платежей, см. пункт 'Получение списка карт'.

Для тестового соединения нужно использовать тестовую карту, см. пункт 'Тестовое подключение'.

В ответ приходит номер платежа, url регистрации и способ перехода.

POST запрос

URL: /card/registration

Параметры запроса:

clientType: тип клиента (устанавливает вид формы оплаты)

Возможные значения:

- web - веб форма (по умолчанию)
- mobile - мобильная форма
- iframe - фрейм

userToken*: идентификатор пользователя

Параметры ответа:

userToken*: идентификатор пользователя

shopToken*: идентификатор организации

regPayNum*: номер платежа

methodType*: метод перехода по url для оплаты (GET/POST)

payUrl*: url оплаты

sign*: подпись

* - обязательные параметры.

Пример запроса:

```
{"sign":"F8C4A945567047ACB55FEC5D222D4786","shopToken":"SHOP_TOKEN","userToken":"USER_TOKEN","clientType":"web"}
```

Пример ответа:

```
{"sign":"C622C723C2A4266340E4E87AADD6B82D","regPayNum":"1234567890","methodType":"GET","payUrl":"https://autopays.ru/payment/#!/search_payment","userToken":"USER_TOKEN","shopToken":"SHOP_TOKEN"}
```



Получение списка карт

У пользователя может быть только одна карта.

Для того что бы карта появилась в списке нужно произвести регистрацию карты см. пункт 'Регистрация карты'.

POST запрос

URL: /get/cards

Параметры запроса:

userToken*: идентификатор пользователя

shopToken*: идентификатор организации

sign*: подпись

Параметры ответа:

userToken*: идентификатор пользователя

shopToken*: идентификатор организации

cards*: список карт

Карта

cardMask*: маскированный номер карты

cardToken*: токен карты (используется для создания рекуррентного платежа, активации карты)

cardType*: тип карты

undefined - не определен

visa

master_card

maestro

mir

state*: статус карты

active - активна (доступна для оплаты)

inactive (не доступна для оплаты, требуется активация, см. пункт: 'Активация карты')

deleted (не доступна для оплаты и активации, например: прошел срок действия)

sign*: подпись

* - обязательные параметры.

Пример запроса:

```
{ "sign": "41C1EF2481885FEB0E510CE71F0CAD1B", "userToken": "USER_TOKEN", "shopToken": "SHOP_TOKEN" }
```



Пример ответа:

```
{"sign":"SIGN","userToken":"USER_TOKEN","shopToken":"SHOP_TOKEN",  
"cards":[{"cardMask":"546949*****9929","cardToken":"77a3adeb-1f50-400a-88a9-  
7127dc9ea9f3","cardType":"master_card","state":"active"}]}
```

Деактивации/удаление карты

Состояние карты становится inactive и она будет не доступна для оплаты (необходимо провести повторную привязку карты, см пункт 'Регистрация карты')

POST запрос

URL: /card/deactivation/

Параметры запроса:

userToken*: идентификатор пользователя

shopToken*: идентификатор организации

cardToken*: идентификатор карты

sign*: подпись

resultState*: результат

success - успех

fail - неудача

desc: описание (пояснение)

userToken*: идентификатор пользователя

shopToken*: идентификатор магазина

sign*: подпись

* - обязательные параметры.

Пример запроса:

```
{"sign":"301032A23C66C3A6F9F403147DDFDF52","userToken":"USER_TOKEN","cardToken":"CARD_TOKEN",  
"shopToken":"SHOP_TOKEN"}
```

Пример ответа:

```
{"sign":"05A2BE11C2A9F41854688848A279747A","resultState":"success","desc":"desc  
Success","userToken":"USER_TOKEN","shopToken":"SHOP_TOKEN"}
```

Создание рекуррентного платежа

В ответ приходит номер платежа, url оплаты и способ перехода.

При рекуррентном платеже url ссылкой на чек.



POST запрос

URL: /do/payment

Параметры запроса:

serviceCode*: код провайдера

userToken*: идентификатор пользователя

amount*: сумма платежа в копейках (сумма которая идет на счет пользователю)

comission*: комиссия платежа в копейках (при отсутствии комиссии передается 0)

cardToken: идентификатор карты (для рекуррентного платежа), см. пункт: 'Получение списка карт'

enableSMSConfirm: возможно или нет участие пользователя в проводке рекуррентного платежа (см пункт 'Параметр enableSMSConfirm', shopAPI)

- false (по умолчанию) - пользователь не участвует в проводке платежа

- true - пользователь участвует в проводке платежа

payType: способ оплаты

Возможные значения:

- card - оплата банковской картой (по умолчанию)

- yd - оплата Яндекс деньгами

- sms - оплата с баланса мобильного телефона (см параметр userPhone)

- wallet - кошелек ЦК

clientType: тип клиента (устанавливает вид формы оплаты)

Возможные значения:

- web - веб форма (по умолчанию)

- mobile - мобильная форма

- iframe - фрейм

- **userPhone****: для СМС платежа номер телефона с которого будет произведена оплата

С кодом страны без '+', например: 79020000000

userEmail***: e-mail для отправки фискального чека

fiscalType***: нужно или нет отправлять фискальный чек

Возможные значения:

- email

отправить фискальный чек на почту

- none (по умолчанию)

фискальный чек не нужен

holdTtl: время заморозки средств (подключение должно поддерживать работу через заморозку)

Время в секундах от исполнения заморозки.



Минимум 10 минут, максимум 4 дня, по умолчанию 30 минут

Если услуга не будет подтверждена произойдет авто возврат средств на карту клиента.

properties*: массив реквизитов.

Должны быть заполнены все обязательные поля. Заполненные не обязательные поля должны иметь корректные значения.

'name' - название, 'value' - значение.

shopToken*: идентификатор организации

sign*: подпись

Параметры ответа:

userToken*: идентификатор пользователя

shopToken*: идентификатор организации

regPayNum*: номер платежа

methodType*: метод перехода по url для оплаты (GET/POST)

payUrl*: url оплаты

sign*: подпись

* - обязательные параметры.

** - для СМС платежа обязательно наличие номера телефона (payType = sms)

*** - если fiscalType не none, обязательно наличие соответствующего параметра

Пример запроса:

```
{"sign":"AB6291B7642B14885D1A3E40038C683F","cardToken":"CARD_TOKEN","value":"10000"},"shopToken":  
"SHOP_TOKEN","clientType":"web","payType":"card","properties":[{"name":"Л/СЧЕТ","value":"9503006477"}, {"n  
ame":"ФИО","value":"Иванов Н П"}, {"name":"АДРЕС","value":"Ленина  
10"}, {"name":"МЕСЯЦ","value":"05.2015"}, {"name":"СУММА_ПЕНИ","value":"10000"}], "serviceCode":"109-  
5804-1","userToken":"USER_TOKEN","amount":"10000","comission":"13300"}
```

Пример ответа:

```
{"sign":"C622C723C2A4266340E4E87AADD6B82D","regPayNum":"1234567890","methodType":"GET","payUrl":"https://  
/autopays.ru/payment/#!search\_payment","userToken":"USER_TOKEN","shopToken":"SHOP_TOKEN"}
```

Уведомление о регистрации карты

URL: {cb url}/card-reg

Параметры запроса:

userToken*: идентификатор пользователя

cardToken*: токен карты

cardMask*: маскированный номер карты



cardType*: тип карты

undefined - не определен

visa

master_card

maestro

mir

clientInfo: идентификатор клиента (задается при создании регистрации карты)

shopToken*: идентификатор организации

sign*: подпись

* - обязательный параметр

В ответ ожидается любой ответ со статусом 200

Пример запроса:

```
{"sign":"XXXXXXXXX","shopToken":"SHOP_TOKEN","userToken":"USER_TOKEN","cardMask":"546949*****9  
929","cardToken":"77a3adeb-1f50-400a-88a9-7127dc9ea9f3","cardType":"master_card"}
```

Список ошибок для рекуррентных платежей

Ответ на запрос: 400 Bad Request.

Если в ответе присутствует параметр sign, необходимо проверять подпись сообщения.

Алгоритм формирования подписи аналогичен, см. Формирование подписи.

Ошибки при которых требуется заново регистрировать карту:

- code:112 {"message":"Unknown card","userMessage":"Необходимо заново зарегистрировать карту","code":112}
- code:116 {"message":"Card record expired":"Необходимо заново зарегистрировать карту","code":116}
- code:117 {"message":"Unknown card","userMessage":"Необходимо заново зарегистрировать карту","code":117}
- code:203 {"message":"Card expired","userMessage":"Действие карты истекло","code":203}
- code:204 {"message":"Channel card declined","userMessage":"Карта в стоп-листе","code":204}
- code:206 {"message":"Card locked","userMessage":"Карта заблокирована","code":206}

Авторизация:

- code:2001 {"message":"Incorrect request param {0}","userMessage":"Некорректный параметр запроса {0}","code":2001}
- code:2700 {"message":"Incorrect signature","userMessage":"Не корректная подпись","code":2700}
- code:2701 {"message":"Unknown shop token","userMessage":"Не известный идентификатор магазина","code":2701}



- code 2703 {"message":"User not registered for shop","userMessage":"Пользователь не зарегистрирован для данного магазина","code":2703}

- code 2709 {"message":"Merchant not registered for shop","userMessage":"Мерчант не зарегистрирован для магазина","code":2709}

Получение списка услуг:

- code 1263 {"message":"Can't pay for number","userMessage":"Невозможно оплатить данный номер","code":1263}

Регистрация:

- code 1951 {"message":"Can't create new user","userMessage":"Не удалось зарегистрировать нового пользователя","code":1951}

- code 2751 {"message":"Invalid user login","userMessage":"Не корректный логин","code":2751}

Активация:

- code 2350 {"message":"Activate card error","userMessage":"Ошибка привязки карты","code":2350}

- code 2356 {"message":"Incorrect sms code","userMessage":"Неверный SMS код","code":2356}

- code 2600 {"message":"SMS to number {0} already sent","userMessage":"Отправили СМС на номер {0}","code":2600, "canSendNewSMSThrough":60000}

canSendNewSMSThrough - время в миллисекундах до следующей попытки отправить SMS

Проверка статуса платежа:

- code 2365 {"message":"For recurrent need pass 3ds":"Необходимо ввести 3ds код","code":2365,"regPayNum":"1234567890","securePageURL":"https://ya.ru"}

- code 2704 {"message":"User not owned payment","userMessage":"Платеж не принадлежит пользователю","code":2704}

- code 2705 {"message":"Shop not owned payment","userMessage":"Платеж не принадлежит магазину","code":2705}

Создание платежа:

- code 1295 {"message":"Can't create payment, please try later","userMessage":"В данный момент принять платеж невозможно, попробуйте позже","code":1295}

- code 2354 {"message":"Create recurrent payment error. Try other car","userMessage":"Ошибка проведения платежа. Попробуйте другую карту"}

- code 2360 {"message":"Can't create recurrent payment","userMessage":"Мы сожалеем, но вы не можете совершить этот платеж привязанной картой из соображений вашей безопасности"}

Общие:

- code:1300 {"message":"Provider not found","userMessage":"Оператор не найден","code":1300}

- code:2702 {"message":"Provider forbidden for shop","userMessage":"Поставщик запрещен для данного магазина","code":2702}

- code:2706 {"message":"Incorrect user state:{0}","userMessage":"Не корректное состояние пользователя","code":2706}



Остальные ошибки:

- code:1262 {"message":"Unknown error, try later","userMessage":"Непредвиденная ошибка, попробуйте позже","code":1262}

Приведенный список основных ошибок может быть дополнен.

Коды ошибок проводки рекуррентных платежей

Носит информативный характер, используется для идентификации проблемы.

Не служит для определения успешности/не успешности платежа, для этого необходимо использовать параметр: **state** ('статус платежа'), см. пункт 'Получение статуса платежа'.

Полученные от эквайринга*:

- 101: **SystemError**, "Системная ошибка. Обратитесь к эквайеру"
- 102: **TemporaryError**, "Временный отказ"
- 103: **BankInteractionError**, "Ошибка взаимодействия с банком"
- 104: **AFSBlocked**, "Операция отклонена"
- 105: **UnsupportedSecurityAction**, "Запрошен 3DS для операции, в которой его использование невозможно"
- 106: **UnexpectedBankResponse**, "Неожиданный ответ банка"
- 107: **AcqAccessDenied**, "Магазину доступ запрещен"
- 108: **AcqIncorrectData**, "Не известный тип заказа, номер заказа или еще что-то в этом духе"
- 109: **ParallelProcessing**, "Попытка параллельной оплаты"
- 110: **ProcessingError**, "Ошибка, связанная с обработкой конкретного заказа"
- 111: **AuthFailed**, "Ошибки авторизации запроса (подпись...)"
- 112: **UnknownCard**, "Неизвестная карта. Необходимо заново зарегистрировать карту"
- 113: **UnknownOrder**, "Неизвестный заказ"
- 201: **ChannelTransactionDeclined**, "Эквайер отклонил транзакцию"
- 202: **ChannelSecurityCheckFailed**, "Эквайер отклонил транзакцию"
- 203: **ChannelCardExpired**, "Действие карты истекло"
- 204: **ChannelCardDeclined**, "Банк отклонил транзакцию"
- 205: **ChannelInsufficientFunds**, "Недостаточно средств"
- 206: **ChannelCardBlocked**, "Карта заблокирована"
- 207: **ChannelNotPermitted**, "Эквайер отклонил транзакцию"
- 208: **ChannelTemporaryDecline**, "Эквайер временно отклонил транзакцию"
- 209: **Channel3DSFailed**, "Эквайер отклонил транзакцию"

Общие:

- 1295: **CreatePaymentFailed**, "Не удалось создать платеж"



2354: **CreateRecurrentFailed**, "Не удалось создать рекуррентны платеж"

- Список кодов может быть дополнен.

Для оплаты на кошелек

Регистрация мерчанта

POST запрос

URL: /registration/merchant

Параметры запроса:

phone*: номер телефона (с кодом, но без '+')

Например: 79020000000

email: e-mail

name: имя

surName: фамилия

middleName: отчество

callName: позывной

region: регион / город мерчанта

docList: список документов (используется для идентификации)

- type - тип документа

а) passport - паспорт

б) snils - снилс

в) inn - ИНН

г) car_drive - водительское удостоверение

- number - номер

shopToken*: идентификатор организации

sign*: подпись

Параметры ответа:

merchantToken*: идентификатор мерчанта

state*: статус

- active - активный (для данного мерчанта доступны все операции)

- inactive - не активный (операции не доступны, требуется активация)

- disable - заблокирован (операции не доступны, активация не возможна,



например: данный мерчант более не работает)

phone*: номер телефона

sign*: подпись

* - обязательные параметры.

Пример запроса:

```
{"sign":"FB1A57BB4DBF026E24375723D0BFE61D","phone":"79020000000","email":"assa@mail.ru","name":"Name","surName":"SurName","middleName":"MiddleName","idenList":[{"type":"passport","number":"453543543"},{"type":"inn","number":"1231231"}]}
```

Пример ответа:

```
{"sign":"E547AC5283ABF43E7FC3F1169F8829C3","state":"active","phone":"79020000000","merchantToken":"MERCHANT_TOKEN"}
```

Статус мерчанта

Если мерчант не зарегистрирован будет сгенерировано исключение MerchantNotRegisteredForShop (код 2709, см. Список ошибок)

POST запрос

URL: /merchant/status

Параметры запроса:

login*: номер телефона (с кодом, но без '+')

Например 79020000000.

shopToken*: идентификатор организации

sign*: подпись

Параметры ответа:

merchantToken*: идентификатор мерчанта

state*: статус

- active - активный (для данного мерчанта доступны все операции)
- inactive - не активный (операции не доступны, требуется активация)
- disable - заблокирован (операции не доступны, активация не возможна, например: данный мерчант более не работает)

phone*: номер телефона

sign*: подпись

* - обязательные параметры.



Пример запроса:

```
{"sign":"2ADD30013A2E5DD4FF870EC2833564DF","login":"79150000000","shopToken":"SHOP_TOKEN"}
```

Пример ответа:

```
{"sign":"E547AC5283ABF43E7FC3F1169F8829C3","state":"active","phone":"79020000000","merchantToken":"MERCHANT_TOKEN"}
```

Создание платежа в пользу мерчанта

В ответ приходит номер платежа, url с ссылкой на чек и способ перехода.

POST запрос

URL: /do/payment

Параметры запроса:

serviceCode*: код провайдера

Выдается менеджером

amount*: сумма платежа в копейках (сумма которая идет на счет пользователю)

comission*: комиссия платежа в копейках (при отсутствии комиссии передается 0)

Актуальный размер комиссии и суммы мин./мах платежа, см. пункт: 'Получение списка доступных провайдеров', Shop API

orderId*: уникальный номер заказа на стороне магазина

description: дополнительные сведения (до 255 символов)

например: город и адрес заказа для идентификации в случае обращения клиента

userToken*: идентификатор пользователя

cardToken*: идентификатор карты

enableSMSConfirm: возможно или нет участие пользователя в проводке рекуррентного платежа (см пункт 'Параметр enableSMSConfirm', shopAPI)

- false (по умолчанию) - пользователь не участвует в проводке платежа

- true - пользователь участвует в проводке платежа

merchantToken*: идентификатор мерчанта

callName**: позывной мерчанта

extraPhone**: дополнительный телефон мерчанта

holdTtl: время заморозки средств

Время в секундах от исполнения заморозки.

Минимум 10 минут, максимум 4 дня, по умолчанию 30 минут.

Если услуга не будет подтверждена произойдет авто возврат средств на карту клиента.



payType:

- card: оплата картой (по умолчанию)
- vk: оплата VK pay (возможность уточняйте у менеджеров)

shopToken*: идентификатор организации

sign*: подпись

Параметры ответа:

userToken*: идентификатор пользователя

merchantToken*: идентификатор мерчанта

shopToken*: идентификатор организации

regPayNum*: номер платежа

methodType*: метод перехода по url для просмотра чека (GET/POST)

payUrl*: url на страницу с чеком

param*:** параметры для оплаты VK pay

sign*: подпись

* - обязательные параметры.

** - используется для дополнительной идентификации мерчанта (поиск платежа, спорные вопросы, смена номера телефона...)

*** - обязателен при оплате VK pay

Пример запроса:

```
{ "sign": "FD3649DF9D4B71C828ACD4D7FB64BF07", "shopToken": "SHOP_TOKEN", "serviceCode": "SERVICE_C  
ODE", "userToken": "USER_TOKEN", "amount": "100000", "comission": "2000"  
,"orderId": "OrderId", "description": "Desc", "cardToken": "CARD_TOKEN", "merchantToken": "MERCHANT_TOKEN"  
}
```

Пример ответа:

```
{ "sign": "63444480E68FAA4ABAAD18CA2EC06438", "regPayNum": "1234567890", "methodType": "GET", "payUrl": "https://autopays.ru/payment/#!/receipt=12345", "merchantToken": "MERCHANT_TOKEN", "userToken": "USER_TOKEN", "shopT  
oken": "SHOP_TOKEN" }
```

Создание платежа в пользу мерчанта, оплата с баланса сотового телефона

В ответ приходит номер платежа, url с ссылкой на чек и способ перехода.

Дополнительная комиссия оплачивается пользователем и зависит от оператора сотовой связи.



Окончательная сумма платежа будет в СМС с подтверждением.

На данный момент поддерживаются операторы: МТС, Мегафон и Теле2

POST запрос

URL: /do/payment/mobile

Параметры запроса:

serviceCode*: код провайдера

Выдается менеджером

amount*: сумма платежа в копейках (сумма которая идет на счет пользователю)

comission*: комиссия платежа в копейках (при отсутствии комиссии передается 0)

Актуальный размер комиссии и суммы мин./маж платежа, см. пункт: 'Получение списка доступных провайдеров', Shop API

orderId*: уникальный номер заказа на стороне магазина

description: дополнительные сведения (до 255 символов)

userToken**: идентификатор пользователя

если клиент ранее был зарегистрирован, то оплата будет произведена с телефона указанного при регистрации (параметр login)

merchantToken*: идентификатор мерчанта

userPhone**: для СМС платежа номер телефона с которого будет произведена оплата

С кодом страны без '+', например: 79020000000

userEmail***: e-mail для отправки фискального чека

fiscalType***: нужно или нет отправлять фискальный чек (только для анонимных платежей)

Возможные значения:

- email

отправить фискальный чек на почту

- none (по умолчанию)

фискальный чек не нужен

shopToken*: идентификатор организации

sign*: подпись

Один из параметров обязателен, отправка сразу обоих параметров не допускается.

Параметры ответа:

userPhone**: номер телефона пользователя

userToken**: идентификатор пользователя

merchantToken*: идентификатор мерчанта

shopToken*: идентификатор организации



regPayNum*: номер платежа

methodType*: метод перехода по url для просмотра чека (GET/POST)

payUrl*: url на страницу с чеком

sign*: подпись

* - обязательные параметры.

** - если клиент ранее был зарегистрирован ожидается userToken, иначе userPhone

*** - если fiscalType не none, обязательно наличие соответствующего параметра.

Пример запроса:

```
{"sign":"474E7210149F773AA9DFAEE595BC6A25","shopToken":"SHOP_TOKEN","serviceCode":"SERVICE_CODE","userPhone":"7902000000","amount":"100000","comission":"2000","orderId":"OrderId","description":"Desc","merchantToken":"MERCHANT_TOKEN"}
```

Пример ответа:

```
{"sign":"311CCF38A1792BB6DCEA54F66770294A","regPayNum":"1234567890","methodType":"GET","payUrl":"https://ckassa.ru/payment/#!receipt=12345","merchantToken":"MERCHANT_TOKEN","userToken":"USER_TOKEN","shopToken":"SHOP_TOKEN"}
```

Получение баланса мерчанта

POST запрос

URL: /get/merchant/wallet/balance

Параметры запроса:

merchantToken*: идентификатор мерчанта

shopToken*: идентификатор организации

sign*: подпись

Параметры ответа:

balance*: баланс в копейках

walletCode*: номер кошелька в системе autopays

merchantToken*: идентификатор мерчанта

sign*: подпись

* - обязательные параметры.

Пример запроса:

```
{"sign":"21C3B7EF125C0163B423CBE387D7AFD3","merchantToken":"MERCHANT_TOKEN","shopToken":"SHOP_TOKEN"}
```



Пример ответа:

```
{"sign":"7CB4C5F56717FA770BED533734966E2D","balance":"1000","walletCode":"WALLET_CODE","merchantToken":"MERCHANT_TOKEN"}
```

Заморозка средств у плательщика (холд)

Бронирование суммы

Бронирует сумму на карте клиента без установки конечного получателя.

Для проводки платежа необходимо задать получателя (merchant) и подтвердить оказание услуги.

В ответ приходит номер платежа, url с ссылкой на чек и способ перехода.

POST запрос

URL: /do/payment-reserv

Параметры запроса:

serviceCode*: код провайдера

Выдается менеджером

amount*: сумма платежа в копейках (сумма которая идет на счет пользователю)

comission*: комиссия платежа в копейках (при отсутствии комиссии передается 0)

Актуальный размер комиссии и суммы мин./мах платежа, см. пункт: 'Получение списка доступных провайдеров', Shop API

orderId*: уникальный номер заказа на стороне магазина

description: дополнительные сведения (до 255 символов)

например: город и адрес заказа для идентификации в случае обращения клиента

userToken*: идентификатор пользователя

cardToken*: идентификатор карты

enableSMSConfirm: возможно или нет участие пользователя в проводке рекуррентного платежа (см пункт 'Параметр enableSMSConfirm', shopAPI)

- false (по умолчанию) - пользователь не участвует в проводке платежа

- true - пользователь участвует в проводке платежа

holdTtl: время заморозки средств

Время в секундах от исполнения заморозки.

Минимум 10 минут, максимум 4 дня, по умолчанию 30 минут.

Если услуга не будет подтверждена произойдет авто возврат средств на карту клиента.



shopToken*: идентификатор организации

sign*: подпись

Параметры ответа:

userToken*: идентификатор пользователя

shopToken*: идентификатор организации

regPayNum*: номер платежа

methodType*: метод перехода по url для просмотра чека (GET/POST)

payUrl*: url на страницу с чеком

sign*: подпись

* - обязательные параметры.

** - используется для дополнительной идентификации мерчанта (поиск платежа, спорные вопросы, смена номера телефона...)

Пример запроса:

```
{"sign":"XXXX","shopToken":"SHOP_TOKEN","serviceCode":"SERVICE_CODE","userToken":"USER_TOKEN",  
"amount":"100000","comission":"2000","orderId":"OrderId","description":"Desc","cardToken":"CARD_TOKEN"}
```

Пример ответа:

```
{"sign":"XXXX","regPayNum":"1234567890","methodType":"GET","payUrl":"https://autopays.ru/payment/#!receipt=12345,  
"userToken":"USER_TOKEN","shopToken":"SHOP_TOKEN"}
```

Обновить получателя забронированных средств

Обновить получателя платежа можно только для бронирования денежных средств (платеж создан через метод 'Бронирование суммы' и находится в статусе hold).

Для зачисления денежных средств платеж необходимо подтвердить см. 'Подтверждение оказания услуги'

POST запрос

URL: /update/payment/merchant

Параметры запроса:

regPayNum*: номер платежа в системе ckassa

merchantToken*: идентификатор мерчанта

callName**: позывной мерчанта

extraPhone**: дополнительный телефон мерчанта

shopToken*: идентификатор организации



sign*: подпись

Параметры ответа:

regPayNum*: номер платежа в системе ckassa

merchantToken*: идентификатор мерчанта

shopToken*: идентификатор организации

sign*: подпись

* - обязательные параметры.

** - используется для дополнительной идентификации мерчанта (поиск платежа, спорные вопросы, смена номера телефона...)

Подтверждение оказания услуги (заморозки)

Необходимость подтверждения оказания услуги определяется настройками терминала.

Подтвердить можно только оплаченный платеж (см. пункт 'Получение статуса платежа')

После получения подтверждения, ckassa переводит деньги на кошелек мерчант.

Услуга считается оплаченной и выполненной.

В случае ошибки нужно повторить запрос.

Если в течении времени жизни заморозки параметр holdTtl (по умолчанию 30 минут), после создания платежа, не было получено 'Подтверждение оказания услуги', то услуга считается не оказанной, платеж отклоняется и деньги возвращаются на карту клиента.

POST запрос

URL: /provision-services/confirm

Параметры запроса:

regPayNum*: номер платежа в системе autopays (выдается при 'Создание платежа')

orderId*: уникальный номер заказа на стороне магазина

amount: сумма платежа в копейках.

Если сумма меньше замороженной будет выполнен частичный возврат, по умолчанию подтверждается полная сумма платежа.

Только для платежей с 'Бронированием суммы'.

shopToken*: идентификатор организации

sign*: подпись



Параметры ответа:

resultState*: результат

success - успех

fail - неудача

desc: описание (пояснение)

shopToken*: идентификатор организации

sign*: подпись

* - обязательные параметры.

Пример запроса:

```
{ "sign": "FE4416FBCCA4E928704E0C158262B568", "regPayNum": "1234567890", "orderId": "12345", "shopToken": "SHOP_TOKEN" }
```

Пример ответа:

```
{ "sign": "XXXX", "resultState": "success", "desc": "Success update provision services to confirm. RegPayNum:1234567890", "shopToken": "SHOP_TOKEN" }
```

Отмена замороженных/забронированных средств

Отменить заморозку и вернуть деньги на карту клиента можно только если платеж находится в статусе hold (провели заморозку, но еще не подтвердили оказание услуги).

После отмены заморозки, ckassa откатывает деньги на карту клиента.

В случае ошибки нужно повторить запрос.

POST запрос

URL: /provision-services/refund

Параметры запроса:

regPayNum*: номер платежа в системе ckassa (выдается при 'Создание платежа')

orderId*: уникальный номер заказа на стороне магазина

shopToken*: идентификатор организации

sign*: подпись

Параметры ответа:

resultState*: результат

success - успех

fail - неудача



desc: описание (пояснение)

shopToken*: идентификатор организации

sign*: подпись

* - обязательные параметры.

Пример запроса:

```
{"sign":"FE4416FBCCA4E928704E0C158262B568","regPayNum":"1234567890","orderId":"12345","shopToken":"SHOP_TOKEN"}
```

Пример ответа:

```
{"sign":"XXXXX","resultState":"success","desc":"Success update provision services to refund. RegPayNum:1234567890","shopToken":"SHOP_TOKEN"}
```

