

PAYTURE

Порядок технической интеграции для платежей мобильной коммерции по Payture API

Версия 2.0

Дата: 27 мая 2025 г.

Изменения документа

Версия	Описание	Дата
1.0	Первая версия документа	21.03.25
2.0	Вторая версия документа Изменение в поле Success на запрос PhonePay	27.05.25

Содержание

1 Общая информация.....	4
2 Порядок технической интеграции.....	4
3 Схема интеграции.....	5
4 Описание API.....	6
5 Тестирование интеграции.....	7

1 Общая информация

Мобильная коммерция — это специализированный метод приёма платежей, который в качестве основного платёжного инструмента использует баланс номера мобильного телефона Покупателя.

Поскольку в данной модели интеграции не используются данные банковской карты, то от Продавца не требуется сертификация по стандарту PCI DSS, а только право на работу с персональными данными Покупателя.

Участники приёма платежей:

Покупатель - физическое лицо, обладающее действующим номером мобильной связи с положительным балансом

Продавец - юридическое лицо, принимающее в свою пользу платежи через сервисы мобильной коммерции за товары и услуги

Платёжный шлюз/Payture - платёжный шлюз Payture, предоставляющий Продавцу техническую возможность приёма платежей мобильной коммерции

Оператор - сервис, выполняющий технологическое, операционное и расчётное сопровождение платежей мобильной коммерции

В настоящем документе рассматривается взаимодействие между Продавцом и Payture в синхронном режиме через отдельные API, специализированные для работы с мобильной коммерцией.

Служба технической поддержки Payture: support@payturetech.com.

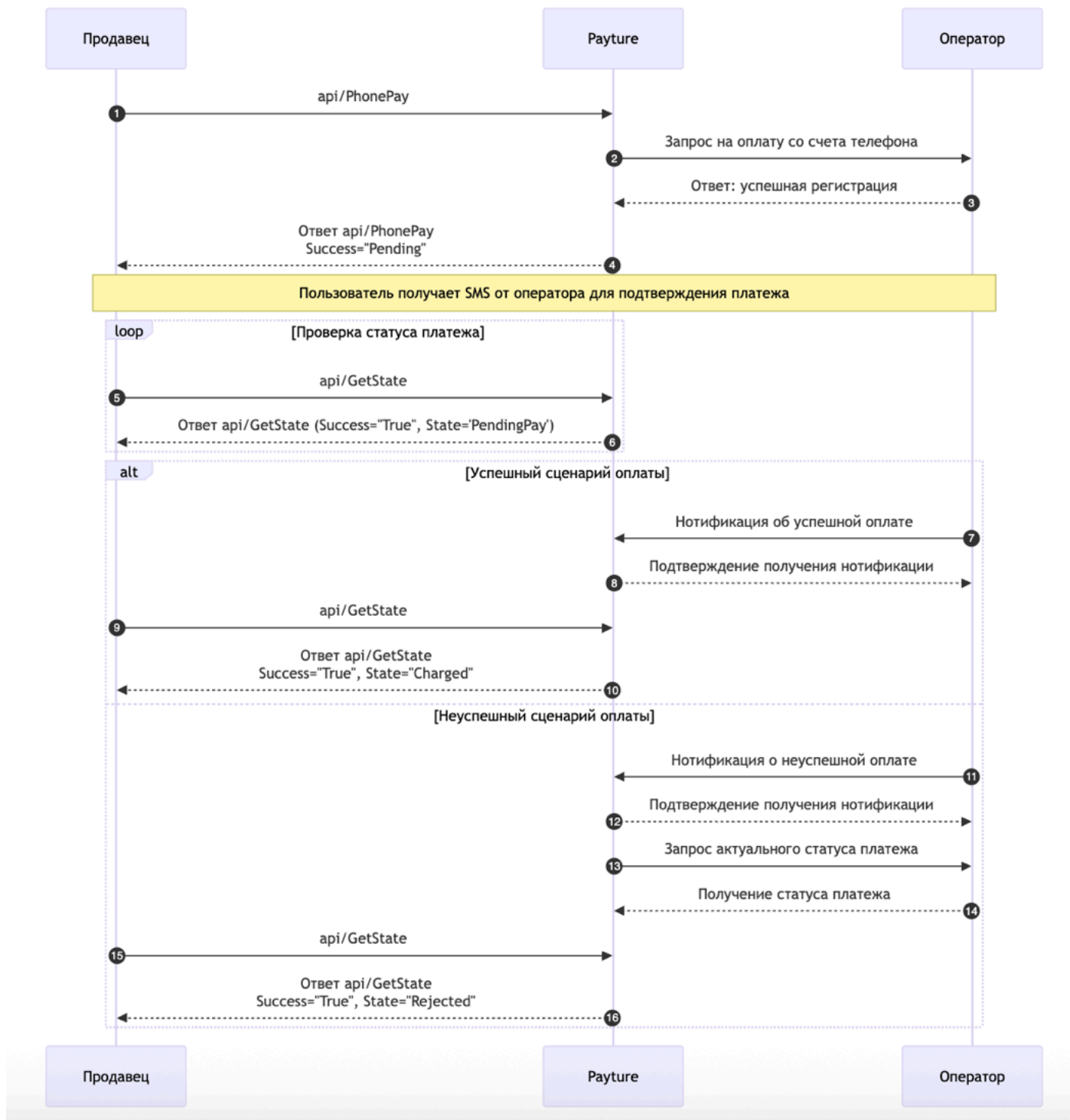
Об ошибках, неточностях, а также о ваших предложениях по улучшению этого документа, пожалуйста, сообщайте службе поддержки Payture.

2 Порядок технической интеграции

В рамках интеграции Продавцу необходимо выполнить следующие шаги:

- 1) Получить от службы поддержки Payture параметры тестового и продуктивного доступа
- 2) Реализовать необходимые для Продавца сценарии выполнения операций
- 3) Подготовить собственную платёжную форму, для размещения на сайте Продавца
- 4) Провести внутреннее тестирование на тестовом или продуктивном окружении (по согласованию с Оператором)
- 5) Выполнить переход в продуктивное окружение

3 Схема интеграции



4 Описание API

Для приёма платежей через сервисы мобильной коммерции на стороне платёжного шлюза Payture подготовлены 2 метода API:

- метод **api/PhonePay** - асинхронный метод оплаты, который выполняет отправку Оператору заявки на оплату со счёта мобильного телефона;
- метод **api/GetState** - синхронный метод получения актуального статуса оплаты со счёта мобильного телефона.

Запрос api/PhonePay

```
https://{Environment}.payture.com/api/PhonePay
```

Параметры:

Поле	Описание	Формат	
Key	Наименование платёжного Терминала	Строка	M
OrderId	Уникальный идентификатор платежа в системе Продавца	Строка (максимум 50 символов)	M
Amount	Сумма платежа в копейках	Цифры, не содержащие десятичных или других разделителей	M
PhoneNumber	Номер телефона Покупателя	Цифры, не содержащие десятичных или других разделителей (11 символов в виде 79031234567)	M

Пример запроса api/PhonePay:

```
curl https://{Environment}.payture.com/api/PhonePay \
-X POST \
-H "Content-Type: application/x-www-form-urlencoded" \
-d Key=Merchant \
-d OrderId=865f86a3-d692-b544-4f0d-ae567fca9a67 \
-d Amount=100 \
-d PhoneNumber=79031234567\
```

Ответ:

XML строка с элементом PhonePay

Поле	Описание	Формат	
Key	Наименование платёжного Терминала	Строка	
OrderId	Идентификатор платежа в системе Продавца	Строка (максимум 50 символов)	M
	Соответствует переданному в запросе		

Поле	Описание	Формат	
Success	Флаг успешности выполнения запроса	Pending— операция отправки заявки на оплату успешна, фактический статус оплаты необходимо получить выполнением метода api/GetState False — операция отправки заявки на оплату неуспешна	M
Amount	Сумма платежа в копейках Передается, если «Success=Pending»	Цифры	O
ErrCode	Код ошибки Передается, если «Success=False»	см. коды ошибок	O

Пример успешного ответа api/PhonePay

```
<PhonePay OrderId="865f86a3-d692-b544-4f0d-ae567fca9a67" Success="Pending" Amount="100" />
```

Запрос api/GetState

```
https://{Environment}.payture.com/api/GetState
```

1) Параметры

Поле	Описание	Формат	
Key	Наименование платежного Терминала	Строка	M
OrderId	Уникальный идентификатор платежа в системе Продавца	Строка (максимум 50 символов)	M

Пример запроса api/GetState

```
curl https://{Environment}.payture.com/api/GetState \
-X POST \
-H "Content-Type: application/x-www-form-urlencoded" \
-d Key=Merchant \
-d OrderId=865f86a3-d692-b544-4f0d-ae567fca9a67 \
```

2) Ответ

XML строка с элементом GetState

Поле	Описание	Формат	
Success	Флаг успешности выполнения запроса	True — заказ найден и статус получен False — не удалось выполнить запрос статуса	M

OrderId	Идентификатор платежа в системе Продавца Соответствует переданному в запросе	Строка (максимум 50 символов)	M
State	Статус платежа	PendingPay – операция оплаты ещё не получила финальный статус на момент выполнения запроса Charged – денежные средства списаны с баланса Покупателя Rejected – неуспешный платеж	M
Forwarded	Признак перенаправления платежа на другой Терминал	Boolean	M
Amount	Сумма платежа в копейках Передается, если «Success=True»	Цифры	O
MerchantContract	§ Наименование платежного Терминала Соответствует переданному в запросе. Передается, если «Success=True»	Строка (максимум 50 символов)	O
FinalTerminal	§ Конечный Терминал, на котором был выполнен платеж Передается, если «Forwarded=True»	Строка	O
AddInfo	Дополнительные параметры транзакции	Элемент AddInfo	O
ErrCode	Код ошибки Передается, если «Success=False»	см. коды ошибок	O

Пример успешного ответа api/GetState

```
<GetState Success="True" OrderId="865f86a3-d692-b544-4f0d-ae567fca9a67" State="PendingPay" Forwarded="False" MerchantContract="Merchant" Amount="12378" />
```

- метод **api/GetState** аналогичен карточному методу **api/GetState**, его описание доступно [в открытой документации Payture](#)
- если операция оплаты ещё не получила финальный статус на момент выполнения запроса **api/GetState**, то в ответе метода будет возвращено значение **Success="True"** и **State="PendingPay"**
- опрос шлюза Payture методом **api/GetState** рекомендуется выполнять с интервалами
 - 5 секунд
 - 10 секунд

- 30 секунд
- 60 секунд (5 попыток)
- 300 секунд (3 попытки)
- 3600 секунд (1 попытка)

5 Тестирование интеграции

Тестирование интеграции необходимо выполнять, используя параметры тестового или продуктивного доступа (при согласовании со стороны Оператора и Payture).

Для оценки результата операций end-to-end-тестирования на продуктивном окружении необходимо обратиться по адресу support@payturetech.com или в чат технической поддержки Payture.