

Подключение электронной торговой площадки к процессинговому центру PayOnline System

Приложение 2

Программный интерфейс Rebill

В данном приложении описан программный интерфейс Rebill, позволяющий выполнять повторные транзакции без непосредственного участия плательщика. Как и остальные программные интерфейсы, Rebill работает по защищенному протоколу https с использованием шифрования SSL, что исключает возможность перехвата информации и нарушения целостности данных третьими лицами. Повторные транзакции могут выполнять только мерчанты с включенной опцией «Rebills».

RebillAnchor

Если у мерчанта включена опция «Rebills», то после завершения оплаты адрес обратной связи (CallBackUrl) вызывается с дополнительным параметром RebillAnchor. Параметр RebillAnchor должен быть сохранен на стороне мерчанта для использования при совершении повторной транзакции.

Повторная транзакция

Повторную транзакцию можно совершить, выполнив HTTP запрос к сервису по адресу <https://secure.payonlinesystem.com/payment/transaction/rebill/>

Сервис создает новую транзакцию с типом Purchase или PreAuth, в зависимости от настроек мерчанта. При создании транзакции используется та же платежная информация, которая была указана плательщиком при совершении первоначальной оплаты. Для вызова метода необходимо передать следующие параметры методом GET или POST:

№	Название	Описание	Тип данных
1	MerchantId	Идентификатор, который вы получаете при активации.	Целое число.
2	RebillAnchor	Ключ повторной транзакции, полученный в параметре RebillAnchor при вызове адреса обратной связи (CallBackUrl) после совершения первоначальной оплаты.	Строка.
3	OrderId	Номер заказа.	Строка.
4	Amount	Сумма.	Число с фиксированной точкой (Decimal), два знака после точки, например 1500.99
5	Currency	Валюта.	Строка, три символа. Возможные варианты: RUB – российский рубль USD – американский доллар EUR – евро
6	SecurityKey	Открытый ключ, подтверждающий целостность параметров запроса.	Строка, 32 символа в нижнем регистре.

7	ContentType	Формат вывода результата – текст или xml. Необязательный параметр, по умолчанию text.	Строка, возможные варианты: text – результат в текстовом формате xml – результат в формате XML.
---	-------------	--	---

SecurityKey должен формироваться по алгоритму, описанному в разделе «Параметры платежной формы» документа «Подключение электронной торговой площадки к процессинговому центру PayOnline System (схема Standart)», в качестве аргумента к функции вычисления хеша подставляется:

MerchantId={MerchantId}&RebillAnchor={RebillAnchor}&OrderId={OrderId}&Amount={Amount}&Currency={Currency}&PrivateSecurityKey={PrivateSecurityKey}

В ответ на запрос с корректными значениями параметров, служба возвращает следующие данные:

№	Название	Описание	Тип данных
1	TransactionId	Уникальный идентификатор транзакции.	Целочисленное число (Long)
2	Operation	Название операции, Rebill.	Строка.
3	Result	Результат выполнения. Обязательный элемент.	Строка, возможные значения: Ok – платеж выполнен; Error – платеж не выполнен.
4	Code	Код результата. Обязательный элемент.	Число.
5	Status	Статус транзакции. Обязательный элемент.	Строка, возможные варианты: Pending – платеж выполнен; PreAuthorized – платеж выполнен, требуется подтверждение; Declined – платеж не выполнен.
6	ErrorCode	Код ошибки. Только, если платеж не выполнен.	Число, возможные значения: 1 - возникла техническая ошибка, повторите попытку через 10 мин. 2 - транзакция отклонена фильтрами, повторите через СУТКИ но не более 5 попыток, после требуется повторное прохождение плательщика через форму; 3 - транзакция отклоняется банком-эмитентом. Возможен

повтор попыток не более пяти
раз в сутки в течение 3 дней;
4 - транзакция отклоняется
банком-эмитентом. Следует
прекратить дальнейшие
операции Rebill с данным
RebillAnchor.

Если указан текстовый формат, то результат будет выглядеть следующим образом:

```
TransactionId={TransactionId}&Operation=Rebill&Result={Result}&Status={Status}&Code={Code}&ErrorCode={Error  
rCode}
```

Если XML:

```
<transaction>  
<id>{id}</id>  
<operation>Rebill</operation>  
<result>{result}</result>  
<status>{status}</status>  
<code>{code}</code>  
<errorCode>{errorCode}</errorCode>  
</transaction>
```