|  |
| --- |
| United Financial Corporation |
| Merchant Integration Technical Manual |

Содержание

[**1.1.1 Generate Transaction ID** 3](#_Toc470618111)

[**1.1.2 Readdressing The Client** 5](#_Toc470618112)

[**1.1.3 Transaction Result** 6](#_Toc470618113)

[**1.1.4 End of Business day** 7](#_Toc470618114)

[2**.1.1 Registering Transaction** 7](#_Toc470618115)

[**2.1.2    Registering Pre-Authorization** 8](#_Toc470618116)

[**2.1.3 Executing financial transaction on Pre-Authorization** 9](#_Toc470618117)

[**2.1. 4    Transaction Reversal** 10](#_Toc470618118)

[**2.1.5 Transaction Refund** 12](#_Toc470618119)

[**2.2.1 Regular Payment** 12](#_Toc470618120)

[**2.2.2 Registration** 13](#_Toc470618121)

[**2.2.3 Execution** 13](#_Toc470618122)

Процесс регистрации мерчанта состоит из следующих этапов:

1. Для интеграции мерчанта нам понадобится IP мерчанта, с которого мерчант попадет в нашу систему, доступ на IP откроется на нашем firewall-е.
2. OK и FAIL URL   (Линк переадресации клиента после завершения операции)
3. Мы предоставим вам сгенерированный на нашей стороне сертификат, который вы должны использовать при выполнении операции (внизу приведена детальная инструкция)
4. Рассмотрим проведение операции

## **1.1.1 Generate Transaction ID**

\*\* ID транзакции является уникальным идентификатором для каждой транзакции, который желательно, чтоб хранился и на стороне мерчанта, поскольку по данному значению можно идентифицировать транзакцию.

[https://ecommerce.ufc.ge:18443/ecomm2/MerchantHandler](https://securepay.ufc.ge:18443/ecomm2/MerchantHandler)    На это url   POST методом высылаете следующие параметры

command=v&amount=<amount>&currency=<currency>&client\_ip\_addr=<ip>&description=<desc>&language=<language>&msg\_type=SMS(&<property\_name>=<property\_value>)\*

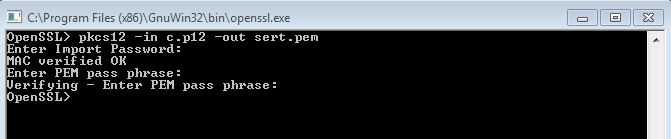
(В примере показаны параметры финансовой транзакции)

Для установления connect-а необходимо указать физическое расположение высланного нами сертификата.

Вы можете сертификат преобразовать в файл с расширением pem использовав openssl библиотеку, следующей командой:

 openssl pkcs12 -in \*.p12 -out \*.pem , где будут объединены в одном файле все части сертификата.

Команда которая сертификат преобразовывает в pem выглядет так:



pass phrase-ой ввел то же, что и в Password.

На сервер параметры вы должны передать использовав cURL библиотеку PHP и при connect-е укажите выше упомянутый pem файл.

Пример:

<?

$curl = curl\_init();

curl\_setopt($curl, CURLOPT\_SSLCERT, cert\_path);

curl\_setopt($curl, CURLOPT\_SSLKEY, cert\_path);

curl\_setopt($curl, CURLOPT\_SSLKEYPASSWD, sert\_password);

curl\_setopt($ch, CURLOPT\_SSL\_VERIFYPEER, 0);

curl\_setopt($ch, CURLOPT\_SSL\_VERIFYHOST, 0);

curl\_close($curl);

?>

Так же для наглядности, если ID транзакции сгенерируется.

<?php

$curl = curl\_init();

$post\_fields = "command=v&amount=500&currency=981&client\_ip\_addr=127.0.0.1&language=EN&description=UFCTEST&msg\_type=SMS";

$submit\_url = "https://ecommerce.ufc.ge:18443/ecomm2/MerchantHandler";

Curl\_setopt($curl, CURLOPT\_SSLVERSION, 0);

curl\_setopt($curl, CURLOPT\_POSTFIELDS, $post\_fields);

curl\_setopt($curl, CURLOPT\_VERBOSE, '1');

curl\_setopt($curl, CURLOPT\_SSL\_VERIFYHOST, '0');

curl\_setopt($curl, CURLOPT\_SSL\_VERIFYPEER, '0');

curl\_setopt($curl, CURLOPT\_RETURNTRANSFER, 1);

curl\_setopt($curl, CURLOPT\_TIMEOUT, 120);

curl\_setopt($curl, CURLOPT\_SSLCERT, getcwd().'/sertificate1.pem');

curl\_setopt($curl, CURLOPT\_SSLKEYPASSWD, 'AAAaaa123456');

curl\_setopt($curl, CURLOPT\_URL, $submit\_url);

$result = curl\_exec($curl);

$info = curl\_getinfo($curl);

if(curl\_errno($curl))

{

echo 'curl error:' . curl\_error($curl)."<BR>";

}

curl\_close($curl);

echo $result; // данный параметр лишь для наглядности , чтоб выдать trans ID.

$curl = curl\_init();

?>

В случае успешных действий вернется сгенерированный на нашей стороне trans ID (уникальный идентификатор транзакции). Дальше trans ID  опять POST методом вышлите на url [https://ecommerce.ufc.ge/ecomm2/ClientHandler](https://securepay.ufc.ge/ecomm2/ClientHandler)   , в ответ вернется линк, после редиректа которого, пользователь попадет на страницу платежей, существующую на стороне UFC (Банка), где пользователь должен заполнить данные карточки.

## **1.1.2 Readdressing The Client**

Например:

<html>

<head>

<title>Merchant example post template to ECOMM</title>

<script type='text/javascript' language='javascript'>

function redirect() {

  document.returnform.submit();

}

</script>

</head>

<body onLoad='<javascript:redirect()>'>

<form name='returnform' action='[https://ecommerce.ufc.ge/ecomm2/ClientHandler](https://securepay.ufc.ge/ecomm2/ClientHandler)' method='POST'>

  <input type='hidden' name='trans\_id' value='уникальный идентификатор транзакции'>

<noscript>

    <center>Please click the submit button below.<br>

    <input type='submit' name='submit' value='Submit'></center>

</noscript>

</form>

</body>

</html>

После нажатия кнопки „Submit“ , по документации, в любом случае , т.е. при введении верных / неверных данных карточки , вернетесь на OK URL, и лишь во время системной ошибки – на FAIL URL.

**returnOkUrl** – the client will be readdressed to this address after 3D Secure authentication and transaction are complete **(regardless of the result).**

**returnFailUrl** – the client will be readdressed to this address should **a technical failure** occur in the ECOMM system

Во время переадресации на OK / FAIL URL в параметре $\_REQUEST["trans\_id"] будет прописан Trsans uniq ID. При помощи идентификатора вы можете узнать статус конкретной транзакции и опираясь на это выдавать соответствующий ответ пользователю.

Для получения статуса инициатором запроса должна быть ваша сторона. Если вы не вышлете эту команду, не будете знать статуса транзакции (Успешная / неудачная).

По документации при помощи данного идентификатора можете выслать команду для результата транзакции:

## **1.1.3** **Transaction Result**

**Command line parameters:**

-c identifies a request for transaction result

Command=c&trans\_id=<trans\_id>&client\_ip\_addr=<ip>(&<property\_name>=<property\_value>)\*

**!!! Внимание!!!** Для того, чтоб сумма финансово отобразилась на вашем счету, **необходимо** технически выполнить операцию закрытия дня.

## 

## **1.1.4 End of Business day**

Как правило эта операция выполняется автоматически раз в 24 часа.

Business day ends on closing the last opened batch for a particular merchant.

**Command line parameters:**

-b identifies a request for the end of business day

Http post parameters:  command=b(&<property\_name>=<property\_value>)\*

Example of the result:

RESULT: OK

RESULT\_CODE: 500 – код ответа закрытия дня

FLD\_075: 12

FLD\_076: 31

FLD\_087: 3201

FLD\_088: 10099

**Перечень существующего функционала, определение и инструкция выполнения:**

## 2**.1.1 Registering Transaction**

Это финансовая транзакция, в это время у владельца карточки с карточного счета происходит операция взымания денег (блокирование суммы) и после произведения выше упомянутого закрытия дня произойдет зачисление суммы на счет мерчанта.

Обязательные параметры:

-v identifies a request for transaction registration

amount transaction amount in fractional units, mandatory (up to 12 digits)

currency transaction currency code (ISO 4217), mandatory, (3 digits)

client\_ip\_addr client’s IP address, mandatory (15 characters)

description transaction details, optional (up to 125 characters)

language authorization language identifier, optional (up to 32 characters)

post parameters:

command=v&amount=<amount>&currency=<currency>&client\_ip\_addr=<ip>&desc=<desc>&language=<language>&msg\_type=SMS(&<property\_name>=<property\_value>)\*

Result:

TRANSACTION\_ID: <trans\_id>

trans\_id transaction identifier (28 characters in base64 encoding)

In case of an error, the returned string of symbols begins with ‘error:‘.

Example of the result:

TRANSACTION\_ID: bAt6JLX52DUbibbzD9gDFl5Ppr4=

Прошу учесть, что после этой команды необходимо выполнить процедуру **1.1.3 Transaction result,** для выяснения результата.

## **2.1.2    Registering Pre-Authorization**

Это финансовое действие, во время которого у владельца карточки блокируется сумма, но не снимается пока мерчант не решить снять сумму со счета (максимальный срок сохранения блока – 30 дней)

Например: при получении услуги в гостинице если нужно заблокировать X сумму, но точно не известно на какую сумму должна быть оплата, или во время оказания любой услуги , когда сумма оплаты может измениться.

Обязательные параметры:

Command line parameters:

-a identifies a request for transaction registration

amount transaction amount in fractional units, mandatory (up to 12 digits)

currency transaction currency code (ISO 4217), mandatory, (3 digits)

client\_ip\_addr client’s IP address, mandatory (15 characters)

description transaction details, optional (up to 125 characters)

language authorization language identifier, optional (up to 32 characters)

post parameters:

command=a&amount=<amount>&currency=<currency>&client\_ip\_addr=<ip>&desc=<desc>&language=<language>&msg\_type=DMS(&<property\_name>=<property\_value>)\*

Result:

TRANSACTION\_ID: <trans\_id>

trans\_id transaction identifier (28 characters in base64 encoding)

In case of an error, the returned string of symbols begins with ‘error:‘

Example of the result:

TRANSACTION\_ID: bAt6JLX52DUbibbzD9gDFl5Ppr4=

Прошу учесть, что после этой команды необходимо выполнить процедуру **1.1.3 Transaction result,** для выяснения результата.

## **2.1.3 Executing financial transaction on Pre-Authorization**

Финансовое подтверждение: чтобы случилось снятие денег с чета у которого сумма сидит в блоке, мерчант должен выслать command – t.

Если сумма в финансовом подтверждении отличается от суммы пре-авторизации (command –a), разница обратно зачислится владельцу карточки.

Command line parameters:

-t identifies a request for transaction registration

auth\_id identifies authorization of a financial transaction

amount transaction amount in fractional units, mandatory (up to 12 digits)

currency transaction currency code (ISO 4217), mandatory, (3 digits)

client\_ip\_addr client’s IP address, mandatory (15 characters)

description transaction details, optional (up to 125 characters)

Http post parameters:

command=t&trans\_id=<auth\_id>&amount=<amount>&currency=<currency>&client\_ip\_addr=<ip>&desc=<desc>&language=<language>&msg\_type=DMS(&<property\_name>=<property\_value>)\*

Results:

RESULT: <result>

RESULT\_CODE: <result\_code>

RRN: <rrn>

APPROVAL\_CODE: <app\_code>

CARD\_NUMBER <pan>

result – transaction results: OK – успешная транзакция, FAILED – неудачная транзакция

result\_code –код ответа для каждой транзакции (3 digits) // (см. result\_code.docx файл)

rrn – идентификатор транзакции на стороне Банка (12 characters)

app\_code – идентификатор транзакции на стороне Банка (max 6 characters)

pan – Маскированный номер карточки

RESULT\_CODE является информационным значением. Транзакция была успешной или нет можно определить только по значению RESULT\_CODE. Поля rrn и app\_code генерируются лишь для технически успешных транзакций.

**Когда транзакция / пре-авторизация успешны, возвращается значение 000** **RESULT\_CODE-а, все остальные значения соответствуют неудачным!**

* Кроме кода 500 , означающего успешное закрытие дня.
* И кроме кода 400 , означающего успешный реверсал.

Example of the result:

RESULT: OK

RESULT\_CODE: 000

RRN: 123456789012

APPROVAL\_CODE: 123456

CARD\_NUMBER: 9\*\*\*\*\*\*\*\*\*\*\*9999

## **2.1. 4    Transaction Reversal**

Функционал реверсала автоматически включен для всех мерчантов.

Необходимо учесть следующее:

* Реверсал на одну транзакцию / авторизацию можно выполнить лишь один раз.
* Возможно выполнить частичный реверсал. Сумма реверсала зачислится владельцу карточки, остальная сумма – мерчанту, после закрытия дня.
* Частичный реверсал нельзя выполнить если сделана пре-авторизация (command –a ). В таком случае, если необходимо вернуть часть суммы, нужно выполнить command –t на ту сумму, которая должна зачислиться мерчанту, а разница зачислиться владельцу карточки (см. 2.1.3 Executing financial transaction on Pre-authorization).
* Сумма реверсала не должна превышать сумму оригинальной транзакции.
* Реверсал можно выполнить лишь до закрытия дня (1.1.4 End of business day)

Например: Если день закрывается в 17:00 PM , а транзакция выполнена в 13:00 PM , сделать реверсал этой транзакции можно до 17:00 PM (включительно). Поскольку реверсал нельзя выполнить после закрытия дня в таком случае нужно сделать рефанд, (см. 2.1.5 Transaction refund).

Command line parameters:

-r identifies a request for transaction reversal

trans\_id transaction identifier, mandatory (28 characters)

Http post parameters:

command=r&trans\_id=<trans\_id>&amount=<amount>

Result:

RESULT: <result>

RESULT\_CODE: <result\_code>

result – reversal results:

OK – successful reversal transaction

FAILED – failed reversal transaction

Example of the result:

RESULT: OK

RESULT\_CODE: 400

**Значение RESULT\_CODE-а реверсала должно быть 400 , чтоб реверсал был успешным.**

**Во всех других случаях реверсал буден неудачным.**

## **2.1.5 Transaction Refund**

Функционал рефанда не включен ни для одного мерчанта, и включается лишь в соглашении с Банком.

Необходимо учесть следующее:

* Рефанд можно выполнить лишь на ту транзакцию, после выполнения которой закрылся день (1.1.4 End of business day).

Например: Если день закрывается в 17:00 PM, транзакция выполнена в 13:00 PM , и эту сумму вернуть стало необходимо после 17:00 PM, то нужно выполнить рефанд.

* Возможно выполнить частичный рефанд
* Возможно выполнить повторно частичный рефанд
* Сумма рефанда не должна превышать сумму оригинальной транзакции
* Если в команде не будет указан параметр amount – произойдет возврат всей суммы.

Command line parameters:

-k identifies a request for transaction refund

trans\_id - transaction identifier, mandatory (28 characters), full original transaction amount is always refunded

Http post parameters:

command=k&trans\_id=<trans\_id>&amount=<amount>

Result:

RESULT: <result>

RESULT\_CODE: <result\_code>

result – Refund results:

OK successful refund transaction

FAILED failed refund transaction

В случае успешного рефанда RESULT\_CODE должен быть 000, в других случаях рефанд будет неудачным.

## **2.2.1 Regular Payment**

Регулярный платеж является функционалом, во время которого пользователь только однажды переходит на страницу платежей, во время каждого последующего снятия денег мерчант может высылать команду, описанную ниже и сумма снимется без перехода на страницу платежей.

Регулярный платеж не включен ни у одного мерчанта, и происходит лишь после соглашения с Банком.

## **2.2.2 Registration**

Command line parameters for registration along with the first payment:

-z request for SMS transaction

amount transaction amount in fractional units, mandatory (up to 12 digits)

currency transaction currency code (ISO 4217), mandatory, (3 digits)

client\_ip\_addr client’s IP address, mandatory (15 characters)

description transaction details

rec\_pmnt\_id merchant-selected regular payment identifier

expiry preferred deadline for a regular payment MMYY

Command line parameters for registration with an authorization for a certain amount, without the first payment:

-p request for authorization and registration

currency transaction currency code (ISO 4217), mandatory, (3 digits)

client\_ip\_addr client’s IP address, mandatory (15 characters)

description transaction details

rec\_pmnt\_id merchant-selected regular payment identifier

expiry preferred deadline for a regular payment MMYY

Http post parameters:

command=z&amount=<amount>&currency=<currency>&client\_ip\_addr=<ip>&desc=<desc>&language=<language>&msg\_type=SMS&biller\_client\_id=<recc\_pmnt\_id>&perspayee\_expiry=<expiry>&perspayee\_gen=1(&<property\_name>=<property\_value>)\*

command=p&amount=0&currency=<currency>&client\_ip\_addr=<ip>&desc=<desc>&language=<language>&msg\_type=AUTH&biller\_client\_id=<recc\_pmnt\_id>&perspayee\_expiry=<expiry>&perspayee\_gen=1(&<property\_name>=<property\_value>)\*

## **2.2.3 Execution**

Command line parameters:

-e request for SMS transaction registration

amount transaction amount in fractional units, mandatory (up to 12 digits)

currency transaction currency code (ISO 4217), mandatory, (3 digits)

client\_ip\_addr client’s IP address, mandatory (15 characters)

description transaction details

rec\_pmnt\_id merchant-selected regular payment identifier

Http post parameters:

command=e&amount=<amount>&currency=<currency>&client\_ip\_addr=<ip>&desc=<desc>&language=<language>&biller\_client\_id=<recc\_pmnt\_id>(&<property\_name>=<property\_value>)\*

Result:

TRANSACTION\_ID: <trans\_id>

RESULT\_CODE:<result\_code>

RRN:<rrn>

APPROVAL\_CODE:<appr\_code>