Moneybookers Ltd.

Moneybookers Automated Payments Interface

Merchant Integration Manual

Version: <2.1>

1 Contents

2   Introduction

The Moneybookers Automated Payments Interface is a useful tool where merchants can execute requests to the Moneybookers servers to:

- Make 'send money' transactions to their customers

- Make 'Pay-on-demand' transactions

- Check the status of transactions, recurring payments, etc.

- Download account histories and repost status reports

These functionalities are described in detail in the following sections.

We strongly advise that you call the Moneybookers URLs by hostname when making requests rather than hardcoding the static IP of our server since Moneybookers may need to change it.

For all merchant support, please contact our Merchant Service Department:

Email: merchantservices@moneybookers.com

Phone: +44 870 383 0762 (Mon-Fri, 8am until 5pm UK time)

3   Important security requirement for making requests to the API

All requests to the API are standard HTTPS requests.

Although previous versions of this manual do not mention this and the HTTP/1.1 specification states that the **'User-Agent' HTTP request header** is not mandatory, it **is now required by Moneybookers**.

It is a means of verification of the program on the client host and if the client does not send this string, we cannot verify nor log it and the response from our servers will be HTTP 404 Forbidden.

Sending the 'User-Agent' is one of the principle rules of our network security and is usually a simple setting in client programs. If you are against sending the header for tracking reasons, you may be interested to know that this is used as a loophole by potential attackers.

4   Send Money per HTTPS Request

Moneybookers transfers are usually performed by customers through the website www.moneybookers.com in the "Send Money" section.

Merchants can execute the same kind of payments using Moneybookers' Automated Payments Interface (API) via simple HTTPS requests. For Merchants wishing to receive payments, please refer to our Merchant Gateway Manual.

The process is divided into two steps:

1.   **Authorisation and preparation of the payment**

2.   **Execution of the transfer**

After each step the Moneybookers' servers return a XML response that contains the result of the performed action. You will probably require test accounts for which you need to open accounts online via our website and inform us of the email addresses used. The accounts will then be made test and funds uploaded. Test accounts operate in the production environment but funds cannot be sent from a test account to a normal account. Before you start testing, please read the security restrictions detailed in Section 6.

4.1 Specification of the Send Money Interface

The merchant must provide the following parameters within HTTPS requests to this URL: https://www.moneybookers.com/app/pay.pl. There are two interaction steps:

4.2 Authorisation and Preparation of the Payment

4.2.1 Request Parameters

Field	Description	Required?	Example value
action	define the first step of the iteration	Y	action=prepare
email	email address of the merchant	Y	moneybookers@merchant.com
password	MD5 of the merchants password	Y	9f535b6ae672f627e4a5f79f2b7c63fe
amount	amount to be transferred	Y	10.95
currency	currency of the amount	Y	EUR
bnf_email	beneficiary's email address	Y	customer@host.com
subject	subject of the notification email	Y	Your order is ready
note	note to be included in the notification email	Y	Details are available at our site…
frn_trn_id	reference id supplied by the merchant (must be unique if submitted)	N	A1234

4.2.2 XML Server Response

The resulting XML will contain a 'response' tag that has one of the following elements:

- If the authorisation and payment preparation were successful, a 'sid' element containing a session identifier is included that is to be submitted at the next step.

- If an error occurs, an 'error' element is included with a 'error_msg' child that contains the error message.

4.2.3 Example

Successful Authorisation:

Request:

```
GET https://www.moneybookers.com/app/pay.pl?action=prepare
&email=merchant@host.com&password=6b4c1ba48880bcd3341dbaeb68b2647f
&amount=1.2&currency=EUR
&bnf_email=beneficiary@domain.com&subject=some_subject&note=some_note
&frn_trn_id=111
```

Response:

```
<?xml version="1.0" encoding="UTF-8"?>
<response>
        <sid>5e281d1376d92ba789ca7f0583e045d4</sid>
</response>
```

Incorrect request (missing 'amount'):

Request:

```
GET https://www.moneybookers.com/app/pay.pl?action=prepare
&email=merchant@host.com&password=6b4c1ba48880bcd3341dbaeb68b2647f
& currency=EUR
&bnf_email=beneficiary@domain.com&subject=some_subject&note=some_note
&frn_trn_id=111
```

Response:

```
<?xml version="1.0" encoding="UTF-8"?>
<response>
    <error>
        <error_msg>MISSING_AMOUNT</error_msg>
    </error>
</response>
```

4.3 Execution of the Transfer

4.3.1 Request Parameters

Field	Description	Required ?	Example value
action	define second step of interaction	Y	action=transfer
sid	session identifier from the previous step	Y	5e281d1376d92ba789ca7f0583e045d4

4.3.2 XML Server Response

The correct XML will contain a 'response' tag that includes the following elements:

- If the payment was successful, the response contains a 'transaction' element with the following children:

Field	Description
Amount	Amount paid in the currency of the merchant's account
Currency	Currency of the merchant's account
Id	Transaction ID
Status	Numeric value of the transaction status: 1 – scheduled (if beneficiary is not yet registered at moneybookers) 2 - processed (if beneficiary is registered)
status_msg	Text value of the transaction status

- If an error occurs, an 'error' element is included with a 'error_msg' child.

4.3.3 Example

Successful Execution of Payment:

```
GET    https://www.moneybookers.com/app/pay.pl?
action=transfer&sid=5e281d1376d92ba789ca7f0583e045d4
```

Response:

```
<?xml version="1.0" encoding="UTF-8"?>
<response>
    <transaction>
        <amount>1.20</amount>
        <currency>EUR</currency>
        <id>497029</id>
        <status>2</status>
        <status_msg>processed</status_msg>
    </transaction>
</response>
```

4.3.4 Reposting the 'transfer' HTTPS request

In case of a communication error during the process of a transfer, merchants are advised to resend the 'transfer' payment request again within 15 minutes of the last 'transfer' request ('transfer' request sessions expire after 15 minutes). This will not cause a second transfer to be executed since one session identifier (sid) will be executed only once by Moneybookers.

There are three possible results:

1. In case this 'transfer' request has been successful already as a result of an older request, Moneybookers will simply return the result of the previous payment.

2. If this 'transfer' request has not yet been executed but a request had already been received before, Moneybookers will respond with EXECUTION_PENDING.

3. If the 'transfer' request has been received for the first time, Moneybookers will respond as described above in section 3.1.2.

5 Merchant Query Interface

The Merchant Query Interface allows merchants to query the Moneybookers database for the current status of transactions performed by them. The Query interface is accessible by posting specific parameters at https://www.moneybookers.com/app/query.pl. The interface requires three general parameters -- email, password, action -- and a number of parameters specific to the requested action (actions are explained below).

General query parameters:

Field Name	Description	Required?	Example value
email	The email address of the merchant's Moneybookers account.	Yes	merchant@merchant.com
password	The hex MD5 of the merchant's password.	Yes	9f535b6ae672f627e4e5f79f2 b7c63fe
action	The required action.	Yes	repost

5.1 Actions

The following actions are defined:

5.1.1 Repost

action=repost

Repost allows merchants to request that a notification of the status of a given transaction be posted to their status_url. In response, the Moneybookers system posts a status report (the format is explained in 3.2.5 of the Merchant Gateway Manual).

The following action-specific parameters are required

Field Name	Description	Required?	Example value
trn_id	Merchant transaction ID.	Yes/No	500123
mb_trn_id	Moneybookers transaction ID.	Yes/No	4585262
status_url	Where to post the notification	No	https://www.merchant.com/mb_notifications.asp

Either trn_id or mb_trn_id must be supplied. If both are given, trn_id will be used. If status_url is not provided, the status_url given at the time the transaction was created will be used. Upon success, HTTP 200 with a body of "OK" is returned.

5.1.2 Transaction status

action=status_trn

This action allows merchants to request the details about a transaction received via the Merchant Gateway. The following parameters are expected:

Field Name	Description	Required?	Example value
trn_id	Merchant transaction ID.	Yes/No	500123
mb_trn_id	Moneybookers transaction ID.	Yes/No	4585262

As with Repost, either trn_id or mb_trn_id must be supplied and if both are given, trn_id will be used. If a transaction with the given ID is found, the response will contain the transaction details encoded as application/x-www-form-urlencoded.

5.1.3 Account history

action=history

Through the 'Account history' action, the merchant may request a list with the details of all transactions performed by them during a given period. The following parameters are expected:

Field Name	Description	Required?	Example value
start_date	The start date in DD-MM-YYYY format	Yes	29-05-2002
end_date	The end date in DD-MM-YYYY format	No	30-06-2002

Upon success, returns the complete account history for the specified period in CSV (comma separated values) form. If the end_date parameter is not specified, we use today's date.

5.1.4 Cancel recurring payment

action= cancel_rec

This action allows merchants to cancel a recurring payment. The following action-specific parameters are required:

Field Name	Description	Required?	Example value
trn_id	Merchant transaction ID	Yes	500123

Upon success, HTTP 200 with a body of "OK" is returned.

5.1.5 Recurring payment status

action=status_rec

This action allows merchants to check the status of a recurring payment. The following action-specific parameters are required:

Field Name	Description	Required?	Example value
trn_id	Merchant transaction ID	Yes	500123

If a transaction with the given ID is found, the response will contain following parameters:

- Status: 0 active/ -1 cancelled/ -2 failed/ 1 finished

- Next payment date in dd-mm-yyyy format. This parameter is returned only if status is 'active' or 'failed'

- End date in dd-mm-yyyy format. This parameter is returned only if status is 'active' or 'failed'

5.1.6 Extending the end date of a recurring payment

action= extend_rec

This action allows merchants to extend the end date (rec_end_date)of a recurring payment. If you want to use this action, you have first to apply for it by contacting merchantservices@moneybookers.com. The following action-specific parameters are required:

Field Name	Description	Required?	Example value
trn_id	Merchant transaction ID	Yes	500123
rec_end_date	The recurrent end date in dd-mm-yyyy format	Yes	30-06-2007

Upon success, HTTP 200 with a body of "OK" is returned.

5.1.7 Cancel pay-on-demand payment

action= cancel_od

This action allows the merchant to cancel a pay-on-demand payment. The following action-specific parameter is required:

Field Name	Description	Required?	Example value
trn_id	Merchant transaction ID	Yes	500123

Upon success, HTTP 200 with a body of "OK" is returned.

5.1.8 Pay-on-demand status

action= status_od

This action allows the merchant to check the status of a pay-on-demand payment. The following action-specific parameter is required:

Field Name	Description	Required?	Example value
trn_id	Merchant transaction ID	Yes	500123

If a transaction with the given ID is found, the response will contain following parameters:

- Status: 0 active/ -1 cancelled

- Last execution date in dd-mm-yyyy format.

5.2 Error messages

The following errors may be returned by the Query interface:

Code	Explanation	Reason
401	Authorization declined.	Wrong email and/or password. Password in wrong format.
402	Unknown action.	Action not one of 'repost', 'status_trn', 'history'.
403	Transaction not found.	Wrong transaction ID.
404	Missing parameter.	A required parameter was not found
405	Illegal parameter value.	The value of a required parameter did not match the expected format.

6 Pay-on-demand Interface

The Pay-on-demand Request Interface allows merchants to debit the customers' Moneybookers account once pay-on-demand has been authorised (refer to our Merchant Gateway Manual, section 4.7). The interface is accessible by posting specific parameters through HTTPS at https://www.moneybookers.com/app/ondemand_request.pl.

You should contact merchantservices@moneybookers.com in order to be allowed to use this service.

6.1 Actions

The following actions are defined:

6.1.1 Prepare

action=prepare

This action prepares the transaction to be executed later on with the *request* action. The following action specific parameters are required:

Field Name	Description	Required?	Example value
email	The email address of the merchant's Moneybookers account	Yes	merchant@merchant.com
password	The hex MD5 of the merchant's password	Yes	9f535b6ae672f627e4e5f79f2b7c63fe
action	The required action	Yes	prepare
amount	Amount of the request for a debit transaction	Yes	10.50
currency	3-letter code of the currency of the maximum amount according to ISO 4217 (see Annex I for accepted currencies)	Yes	EUR
ondemand_note	Text that will be shown to the customer on the confirmation page as a reason for the pay-on-demand payment	No	Sample merchant will debit your account so that you can continue using our services
frn_trn_id	Merchant transaction ID used for the recurring payment	Yes/No	A205220

rec_payment_id	Recurring payment ID as sent to the merchant on the status_url and rec_status_url upon creating the pay-on-demand payment	Yes/No	200005

Either *frn_trn_id* or *rec_payment_id* must be provided.

If o*ndemand_note* is not provided, the one that is submitted when creating the pay-on-demand payment will be used.

A session identifier *sid* parameter is returned upon success.

6.1.2 Request

action=request

This action executes the actual transaction. The following action specific parameters are required:

Field Name	Description	Required?	Example value
sid	Session identifier	Yes	7783bfa23641a627e4a5f79f2b7c6
action	The required action.	Yes	request

Upon success, returns the details of the transaction as an XML response. This response contains the following fields:

Field Name	Description	Example value
cust_amount	Amount of the transaction as requested by the merchant	10.50
cust_currency	3-letter code of the currency of the amount as requested by the merchant, according to ISO 4217 (see Annex I of Merchant Gateway Manual for accepted currencies)	EUR
trn_status	Pay-on-demand status: 0 – active, -1 – cancelled, -2 – failed	0
trn_id	Transaction ID	500123
rec_payment_id	Recurring payment ID	200005

rec_payment_type	Type of payment	on-demand

If a request fails, the merchant is not allowed to make more than two requests for a debit of a customer's account on demand per customer per 24 hours.

The customer is notified via email for every pay-on-demand request executed by a merchant.

7   Security Restrictions

Merchants are able to apply access restrictions to this service using the "Automated Payments Interface" section within the "Merchant tools" page. Please contact us if you cannot see the "Merchant tools" page in your "My account" section:

- Enable or disable the automated payments. By default "Automated Payments Interface" is disabled.

AUTOMATED PAYMENTS INTERFACE

☐ Enable service

Allow payments only from the following IP addresses (separated by spaces)

- Restrict the access to the interface by supplying an IP address (or a part of an address) or a list of IP addresses delimited by space that will be allowed to execute payments.

Example of using only one IP address:

AUTOMATED PAYMENTS INTERFACE

☑ Enable service

Allow payments only from the following IP addresses (separated by spaces)   192.168.15.1

Example of using multiple IP addresses:

AUTOMATED PAYMENTS INTERFACE

☑ Enable service

Allow payments only from the following IP addresses (separated by spaces)   192.168.15.1 192.168.15.2

ANNEX I – Error Messages

Errors when validating parameters

Error	Resolution
LOGIN_INVALID	Email address and/or password were not provided
MISSING_EMAIL	Provide registered email address of merchant account
MISSING_PASSWORD	Provide correct password of merchant account
MISSING_AMOUNT	Provide amount you wish to send
MISSING_CURRENCY	Provide currency you wish to send
MISSING_BNF_EMAIL	Provide email address of the beneficiary
MISSING_SUBJECT	Provide subject of the payment
MISSING_NOTE	Provide notes for the payment

Errors during log in

Error	Resolution
CANNOT_LOGIN	Email address and/or password are incorrect
PAYMENT_DENIED	Check in your account profile that the API is enabled and you are posting your requests from the IP address specified

Errors when validating payment details

Error	Resolution
INVALID_BNF_EMAIL	Check the format of the beneficiary email address
INVALID_SUBJECT	Check parameter length submitted
INVALID_NOTE	Check parameter length submitted
INVALID_FRN_TRN_ID	Check parameter length submitted

Error	Resolution
INVALID_AMOUNT	Check amount format
INVALID_CURRENCY	Check currency code
EXECUTION_PENDING	If you resend a transfer request with the same session identifier before the 'transfer' request was processed, this error will be returned
ALREADY_EXECUTED	If you have requested that the value for frn_trn_id must be unique for each transfer, this error will be returned when you try to submit the same value for more than one transfer
BALANCE_NOT_ENOUGH	Sending amount exceeds account balance
SINGLE_TRN_LIMIT_VIOLATED	Maximum amount per transaction = EUR 10,000
DISALLOWED_RECIPIENT	You are not permitted to send money to the recipient. E.g. Gaming merchants are not permitted to send or receive payments to/from US based customers
CHECK_FOR_VERIFIED_EMAIL	Your account email address needs to be verified
LL_NO_PAYMENT	Your account is locked for security reasons. Please contact us

Errors when making pay-on-demand requests

Error	Resolution
RECIPIENT_LOCKED	The customer's account is locked for outgoing payments
BALANCE_NOT_ENOUGH	The customer's account balance is insufficient
RECIPIENT_LIMIT_EXCEEDED	The customer's account limits are not sufficient
CARD_FAILED	The customer's credit or debit card failed
REQUEST_FAILED	Generic response for transaction failing for any other reason
ONDEMAND_CANCELLED	The customer has cancelled this on-demand payment
ONDEMAND_INVALID	The on-demand payment requested does not exist