

The Network Services

Multiposting Integration Guide

2020-08-10

Contents

1	Definitions	4
2	Introduction	5
2.1	Purpose	5
2.2	Tool core functionalities	5
2.3	Understanding integration	6
3	Quickstart	7
3.1	Prerequisites	7
3.2	Create new listing	7
4	Alternative and optional flows	10
4.1	Error handling	10
4.2	Salary mapping	10
5	Security	11
5.1	Credentials	11
6	Provider services	12
6.1	Create listing	12
6.2	Update listing	13
6.3	Delete listing	14
6.4	Test create message service	15
6.5	Test update message listing	16
6.6	Categories	17
6.7	Job Boards	18
6.8	Request status	19
6.9	Credit status	23
6.10	Software Development Kit (SDK)	24
7	Job Board services	25
7.1	JobSON	25
7.2	Confirmation	27
7.3	Statistical data	29
A	Standard Response Codes	31
B	Job Board Validations	32
C	JobSON validations	33

1 Definitions

Keyword	Description
Customer	the company who posts jobs to job board
Job board	the web site to which the listings are delivered
Job board credentials	a set of information necessary to authenticate the customer's listings on a given job board
Listing	a definition of a given job offer
Listing duration	the number of days in which the listing should stay online on the job board
Provider	a third-party ATS responsible for delivery of customer's listing
SRESTI	Standard Representational State Transfer (REST) Integration
JobSON	api defined to be implemented by job boards that want to integrate with Multiposting

2 Introduction

2.1 Purpose

This document contains information about Multiposting process and tool.

The first chapter describes process from business point of view. There is general information:

- **WHAT** you can do with Multiposting tool?
- **HOW** Multiposting Tool works?
- **WHAT** can you achieve using it?

Following chapters are more technical. After reading them you will know:

- **HOW** to use Multiposting tool?

The technical chapters are divided into four parts:

1. Quick Start (shiny day scenario)
2. Alternative and optional use cases
3. Security
4. Services (structure, examples, possible error codes, etc.)

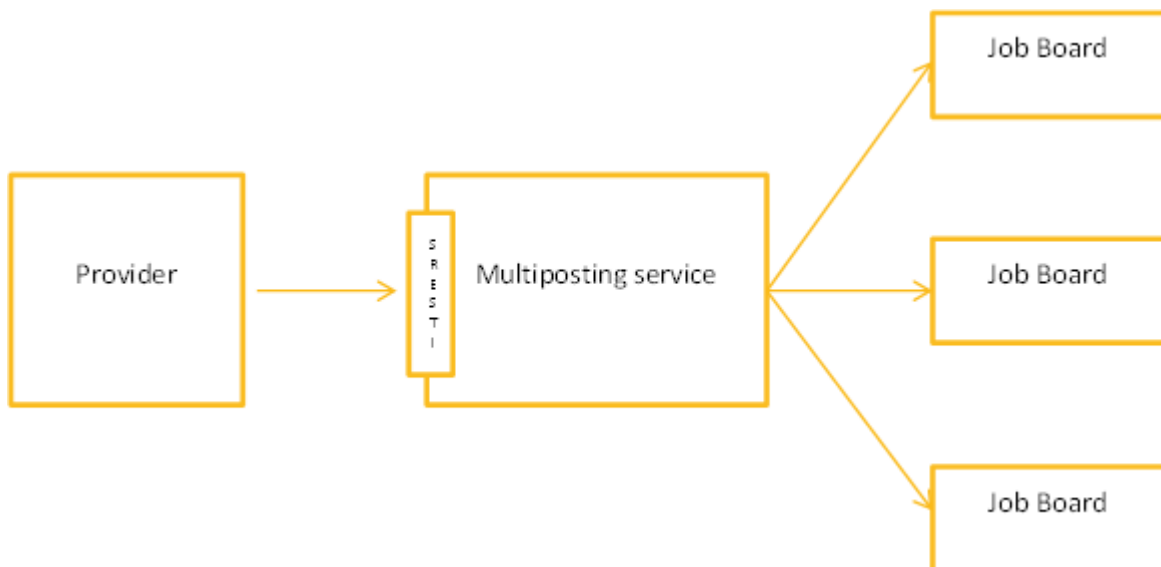
Who should read this document? Anyone, looking for answers for above questions.

Notice: *there are some subsections (like "understanding integration" and "prerequisites") containing some very basic information. If you believe you are familiar with that kind of topics, you can skip them.*

2.2 Tool core functionalities

Multi Posting Tool enables Providers to easily post listings into single or multiple Job Boards.

Once Provider's request is successfully validated, the MP Tool takes responsibility for delivery of the listing for a set of desired Job Boards. During listing delivery and in a post-delivery stage Providers can check status of their request fulfilment.



Following chapters contains details how to perform such posting, and what events can occur during posting.

2.3 Understanding integration

SRESTI - stands for Standard Representational State Transfer (**REST**) Integration.

There's available a set of web services, which you can access over **HTTP** protocol.

Each web service represents one kind of resource, and can be accessed via defined **URL**. Each kind of resource got set of allowed methods/verbs. For usage, possible combinations, and examples - please refer to chapter [3](#).

Data layer - currently supported format is **JSON**, with **UTF-8** encoding.

3 Quickstart

3.1 Prerequisites

Using the Multi Posting Tool is easy, however there are some prerequisites required to perform integration:

1. **Provider's Credentials**

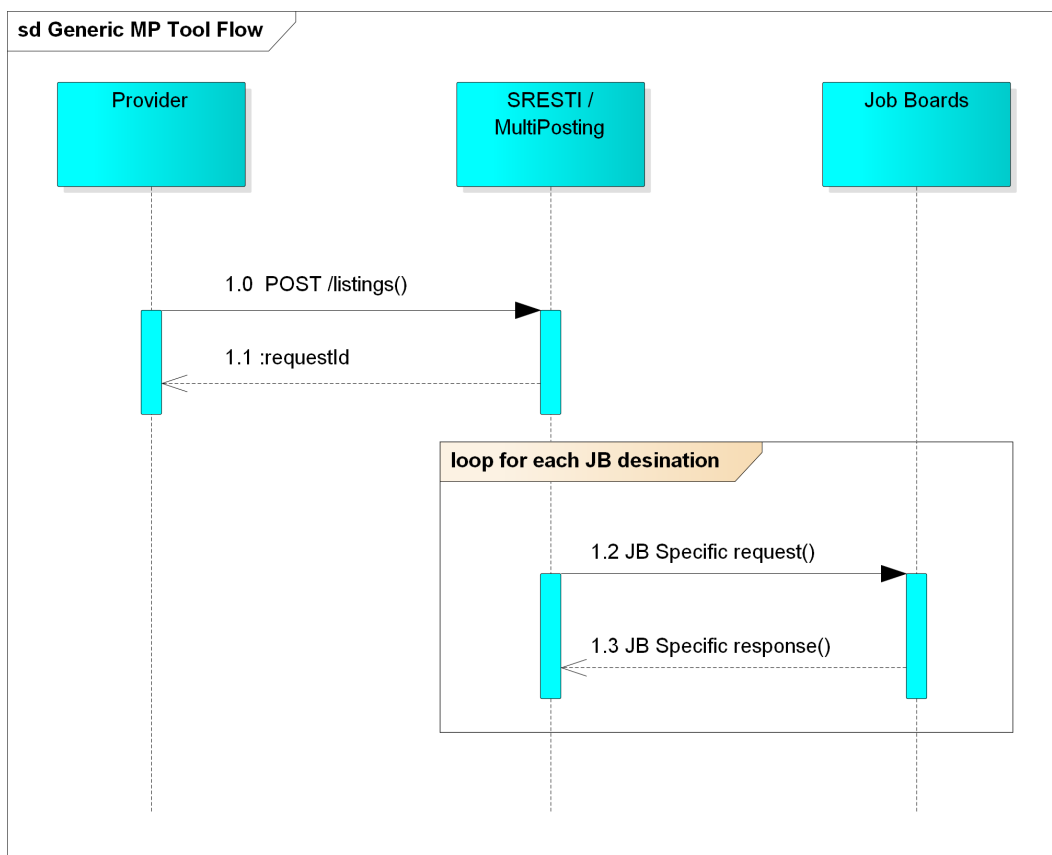
For more details please check the Security section link.

2. **Network connection**

Make sure you can establish clear connection (in terms of non-blocked http port traffic) between Provider and SRESTI hosts.

3.2 Create new listing

Posting new listing is possible as follows:



1. Provider's posting request (<https://flashpost.the-network.com/listings>)
2. SRESTI sends back request reference id
3. For each identified Job Board destination MPT sends:
 - (a) JobBoard specific request and receives
 - (b) JobBoard specific response (if applies)

sectionDebug email integration Multiposting tool has special debug integration which can be used to check the connection from the Provider's point of view.

3.2.1 Step 1: Create new listing

To create new listing you can post the following JSON message to <https://flashpost.the-network.com/listings>:

↔ Payload

The following changes should be made to the above JSON:

1. `customerId` should be changed to the ID provided by The Network
2. `jobBoards / jobBoardId` should be changed to the ID of Debug Email integration; The ID is available through service described in section 6.7
3. `companyDetails / contactEmail` should be changed to your email

In case of successful posting, the system returns `RequestId` and sends email to address received in `companyDetails / contactEmail`.

```
{
  "description" : "JSON file validated, pending contract check and job board: 12345 processing",
  "warnings" : [ {
    "field" : "jobDetails.categories.geography",
    "description" : "More than one geography was selected. Only geography id 200000 will be used."
  }, {
    "description" : "Warning about whole message"
  }, {
    "field" : "jobDetails.categories.workfield",
    "description" : "More than one workfield was selected. Only workfield id 1002000 will be used."
  } ],
  "requestId" : 100,
  "trackingLink" : "https://flashpost.the-network.com/tracking/link/d59b84a4-b627-4959-a547-188c41c2c9d0",
  "resultCode" : 0
}
```

For more information on validation, please see appendix B.

3.2.2 Step 2: Check the status

To check the status of the created request, `RequestId` from Step 1 must be used. The GET request should be sent to <https://flashpost.the-network.com/status/requestId>.

```
{
  "messages" : [ {
    "action" : "CREATE",
    "status" : "ACCEPTED",
    "timeReceived" : "2020-08-10",
    "statusDescription" : "The listing has been accepted by MP",
    "referenceId" : "123/456/789",
    "author" : "user1@example.org"
  }, {
    "action" : "UPDATE",
    "status" : "ERROR",
    "timeReceived" : "2020-08-10",
    "statusDescription" : "Unexpected job board response error",
    "referenceId" : "123/456/789",
    "author" : "user2@example.org"
  } ],
  "state" : "OFFLINE",
  "stateDescription" : "Job board is unable to publish",
  "jobBoardId" : 1,
  "jobBoardUrl" : "https://jobboard.example.com/job/12345",
  "jobBoardReferenceId" : "JB12345",
  "publicationTime" : "2020-08-10",
  "expirationTime" : "2020-09-09",
  "trackingLink" : "https://flashpost.the-network.com/tracking/link/d59b84a4-b627-4959-a547-188c41c2c9d0",
}
```

```
"resultCode" : 0  
}
```

The system answers with a JSON containing current status. More information in section [6.8](#).

4 Alternative and optional flows

4.1 Error handling

For every request to SRESTI, system returns response. For success response contains result code 0, and http 200 response code (RFC 2616). Sometimes an exceptional situation may occur. Usually it is caused by a malformed request. To make things easier, SRESTI uses dedicated error codes (negative numbers), which enable you to take a quick look what and why went wrong. You can find possible error codes, messages and description in [Appendix 1 - Error codes link](#). If you don't need that level of granularity, you can check whether an error occurred or not by http response code.

4.1.1 Status checking

Once Provider has sent proper request to create new listing, it is possible to check status of request fulfilment, based on returned Request Id. Notice: we are working hard to provide accurate posting status, however it is based on Job Board's behavior, therefore there is possibility of delays and some misleading information.

4.1.2 Ignored updates

If an update request is the same as a previous create or update message then the request will be ignored. The server's response will contain a warning if such thing happens.

4.2 Salary mapping

4.2.1 Currency and period transformation

If Job Board accepts salary values only in single currency and single period (for example only in EUR/YEAR) then Multiposting transforms provided values by multiplying them by value based on currency rates and/or period length.

4.2.2 Limited salary range options

Some Job Boards accepts only limited set of salary range options. In that case Multiposting chooses first salary range that contains average value of beginning and end of salary range sent to Multiposting. For example Job Board allows only salary ranges 1000-2000, 2000-3000, 3000-4000 and so on. Multiposting receives request with salary range 1500-3500. In that case range 2000-3000 is chosen because it contains average value $\frac{1500+3500}{2} = 2500$. Transformations mentioned in previous point may also be performed if needed.

4.2.3 Salary range narrowing

If Job Board add some constraints to size of range, and range received is wider than possible then it is be changed by adding to min and subtracting from max the same value needed to fulfill constraint. For example Keljob job board requires that $min \leq max \leq 1.5 * min$. If Multiposting receives 100-160 it changes it to 104-156 when sending it to Keljob.

5 Security

Posting listings to the Job Boards are taken seriously from the security point of view. To prevent unauthorized access, mistakes, and intentional (or unintentional) abuse of SRESTI service - we require Providers to identify themselves (authenticate) so we can grant them (authorize) posting to assigned Job Boards. Good security is based on as simple as possible effective rules. To do so, we use HTTP Basic Authentication. Authenticated Provider automatically gets access to all granted SRESTI functionalities. It is possible to check which Job Boards are available for the Provider using <https://flashpost.the-network.com/jobBoards> service.

5.1 Credentials

Providers are recognized by Credentials (a pair of login and password). Please keep them secret and do not share with others. You should have already received them - if not - please contact our representative.

To check whether your credentials work correctly you can try to access list of available JobBoards (do not forget to change YOUR_CUSTOMER_ID into proper number):

https://flashpost.the-network.com/jobBoards?customerId=YOUR_CUSTOMER_ID

6 Provider services

6.1 Create listing

URL	https://flashpost.the-network.com/listings
Method	POST
Encoding	UTF-8
Content type	application/json
URL Parameters	source – String, duration – Integer

6.1.1 Payload

↔ Payload

The exact list of fields is Job Board dependent. Appendix B contains the validations per Job Board.

6.1.2 Success response

```
{
  "description" : "JSON file validated, pending contract check and job board: 12345 processing",
  "warnings" : [ {
    "field" : "jobDetails.categories.geography",
    "description" : "More than one geography was selected. Only geography id 200000 will be used."
  }, {
    "description" : "Warning about whole message"
  }, {
    "field" : "jobDetails.categories.workfield",
    "description" : "More than one workfield was selected. Only workfield id 1002000 will be used."
  } ],
  "requestId" : 100,
  "trackingLink" : "https://flashpost.the-network.com/tracking/link/d59b84a4-b627-4959-a547-188c41c2c9d0",
  "resultCode" : 0
}
```

6.1.3 Error response

```
{
  "description" : "Request validation errors: Application email is required; [customerId] cannot be null",
  "errors" : [ {
    "description" : "Application email is required"
  }, {
    "field" : "customerId",
    "description" : "cannot be null"
  } ],
  "warnings" : [ {
    "field" : "jobDetails.categories.workfield",
    "description" : "More than one workfield was selected. Only workfield id 200000 will be used."
  } ],
  "requestId" : 0,
  "resultCode" : -100
}
```

6.2 Update listing

URL	https://flashpost.the-network.com/listings/{requestId}
Method	PUT
Encoding	UTF-8
Content type	application/json
URL Parameters	source – String, duration – Integer

6.2.1 Payload

↪ [Payload](#)

The exact list of fields is Job Board dependent. Appendix [B](#) contains the validations per Job Board.

6.2.2 Success response

```
{
  "description": "JSON file validated, pending contract check and job board: 12345 processing",
  "warnings": [ {
    "field": "jobDetails.categories.geography",
    "description": "More than one geography was selected. Only geography id 200000 will be used."
  }, {
    "description": "Warning about whole message"
  }, {
    "field": "jobDetails.categories.workfield",
    "description": "More than one workfield was selected. Only workfield id 1002000 will be used."
  } ],
  "requestId": 100,
  "trackingLink": "https://flashpost.the-network.com/tracking/link/d59b84a4-b627-4959-a547-188c41c2c9d0",
  "resultCode": 0
}
```

6.2.3 Error response

```
{
  "description": "Request validation errors: Application email is required; [customerId] cannot be null",
  "errors": [ {
    "description": "Application email is required"
  }, {
    "field": "customerId",
    "description": "cannot be null"
  } ],
  "warnings": [ {
    "field": "jobDetails.categories.workfield",
    "description": "More than one workfield was selected. Only workfield id 200000 will be used."
  } ],
  "requestId": 0,
  "resultCode": -100
}
```

6.2.4 Update constraints

Due to job board specific rules some value modifications are not allowed in UPDATE action as performing them results in consumption of another credit on target job board as the modified listing is considered as a new one. Multiposting tool does not allow such actions and validates updates checking if UPDATE constraints are not violated. Please see [B](#) for detailed description of constraints that are in place for each field of the request.

6.3 Delete listing

URL	https://flashpost.the-network.com/listings/{requestId}
Method	DELETE
Encoding	UTF-8
Content type	application/json
URL Parameters	source – String

6.3.1 Payload

```
[101]
```

6.3.2 Success response

```
{
  "description" : "JSON file validated, pending contract check and job board: 12345 processing",
  "errors" : [ {
    "description" : "JSON file validated, pending contract check and job board: 12345 processing"
  } ],
  "resultCode" : -104
}
```

6.3.3 Error response

```
{
  "description" : "listing was already deleted",
  "errors" : [ {
    "description" : "listing was already deleted"
  } ],
  "resultCode" : -104
}
```

6.4 Test create message service

These service allow testing correctness of the create listing JSON payload. They will not publish or update existing listings. There is UI for this service that allows to input JSON content as a text directly - <https://flashpost.the-network.com/listingTest.html>

URL	<code>https://flashpost.the-network.com/listingsTesting</code>
Method	POST
Encoding	UTF-8
Content type	application/json
URL Parameters	source – String

6.4.1 Payload

This service accepts the same messages as 6.1.

6.4.2 Success response

Responses have the same structure as in 6.1. The only difference is requestId and trackingLink will never be returned (because listing is not being created) and permission errors will be returned as warnings (so it is possible to test correctness even if account is not set up yet).

```
{
  "description" : "CREATE content correct for job board 12345",
  "warnings" : [ {
    "field" : "jobDetails.categories.geography",
    "description" : "More than one geography was selected. Only geography id 200000 will be used."
  }, {
    "description" : "Customer 54321 is not allowed to post on jobboards [12345]"
  }, {
    "field" : "jobDetails.categories.workfield",
    "description" : "More than one workfield was selected. Only workfield id 1002000 will be used."
  } ],
  "requestId" : 0,
  "resultCode" : 0
}
```

6.4.3 Error response

```
{
  "description" : "Request validation errors: Application email is required; [customerId] cannot be null",
  "errors" : [ {
    "description" : "Application email is required"
  }, {
    "field" : "customerId",
    "description" : "cannot be null"
  } ],
  "warnings" : [ {
    "field" : "jobDetails.categories.workfield",
    "description" : "More than one workfield was selected. Only workfield id 200000 will be used."
  } ],
  "requestId" : 0,
  "resultCode" : -100
}
```

6.5 Test update message listing

URL	<code>https://flashpost.the-network.com/listingsTesting/{requestId}</code>
Method	PUT
Encoding	UTF-8
Content type	application/json
URL Parameters	source – String

6.5.1 Payload and response

This service accepts and returns the same messages as [6.2](#).

6.6 Categories

Provider can access the current list of categories. Categories are grouped into the following types: geography, workfield, sector, contractType, experience, education, workSchedule.

Providers are required to use common MPT Categories regardless of the target Job Board. During posting MPT takes care of proper Category mapping, per each target Job Board.

There is possibility to use multiple categories (where applicable), for example:

```
{
  "geography": ["200713", "200804"]
}
```

URL	https://flashpost.the-network.com/categories
Method	GET
Encoding	UTF-8
Content type	application/json
URL Parameters	jobBoardId – Long (required), categoryType – String

6.6.1 Payload

There's no payload for this service.

6.6.2 Success response

```
{
  "resultCode": 0,
  "description": "",
  "categories": {
    "workfield": [
      {
        "id": 1007000,
        "description": "Helpdesk & Technical Support"
      },
      {
        "id": 5002000,
        "description": "Payroll"
      }
    ]
  }
}
```


6.7 Job Boards

Provider can access the current list of job boards that a customer is allowed to post on. The ids of job boards are necessary to create listing.

If customerId parameter is not provided service will return full list of job boards that can be set up on Multiposting by our customer support.

URL	https://flashpost.the-network.com/jobBoards
Method	GET
Encoding	UTF-8
Content type	application/json
URL Parameters	customerId – Long

6.7.1 Payload

There's no payload for this service.

6.7.2 Success response

```
{
  "description" : "",
  "jobBoards" : [ {
    "jobBoardId" : 1,
    "jobBoardName" : "Job Board Name 1"
  }, {
    "jobBoardId" : 2,
    "jobBoardName" : "Job Board Name 2"
  } ],
  "resultCode" : 0
}
```

6.8 Request status

Provider can check the status of listings that he tried to post. The response is generated, based on RequestId returned by Create New Listing method.

URL	<code>https://flashpost.the-network.com/api/status/v2/{requestId}</code>
Method	GET
Encoding	UTF-8
Content type	application/json

6.8.1 Payload

There's no payload for this service.

6.8.2 Success response

```
{
  "messages" : [ {
    "action" : "CREATE",
    "status" : "ACCEPTED",
    "timeReceived" : "2020-08-10",
    "statusDescription" : "The listing has been accepted by MP",
    "referenceId" : "123/456/789",
    "author" : "user1@example.org"
  }, {
    "action" : "UPDATE",
    "status" : "ERROR",
    "timeReceived" : "2020-08-10",
    "statusDescription" : "Unexpected job board response error",
    "referenceId" : "123/456/789",
    "author" : "user2@example.org"
  } ],
  "state" : "OFFLINE",
  "stateDescription" : "Job board is unable to publish",
  "jobBoardId" : 1,
  "jobBoardUrl" : "https://jobboard.example.com/job/12345",
  "jobBoardReferenceId" : "JB12345",
  "publicationTime" : "2020-08-10",
  "expirationTime" : "2020-09-09",
  "trackingLink" : "https://flashpost.the-network.com/tracking/link/d59b84a4-b627-4959-a547-188c41c2c9d0",
  "resultCode" : 0
}
```

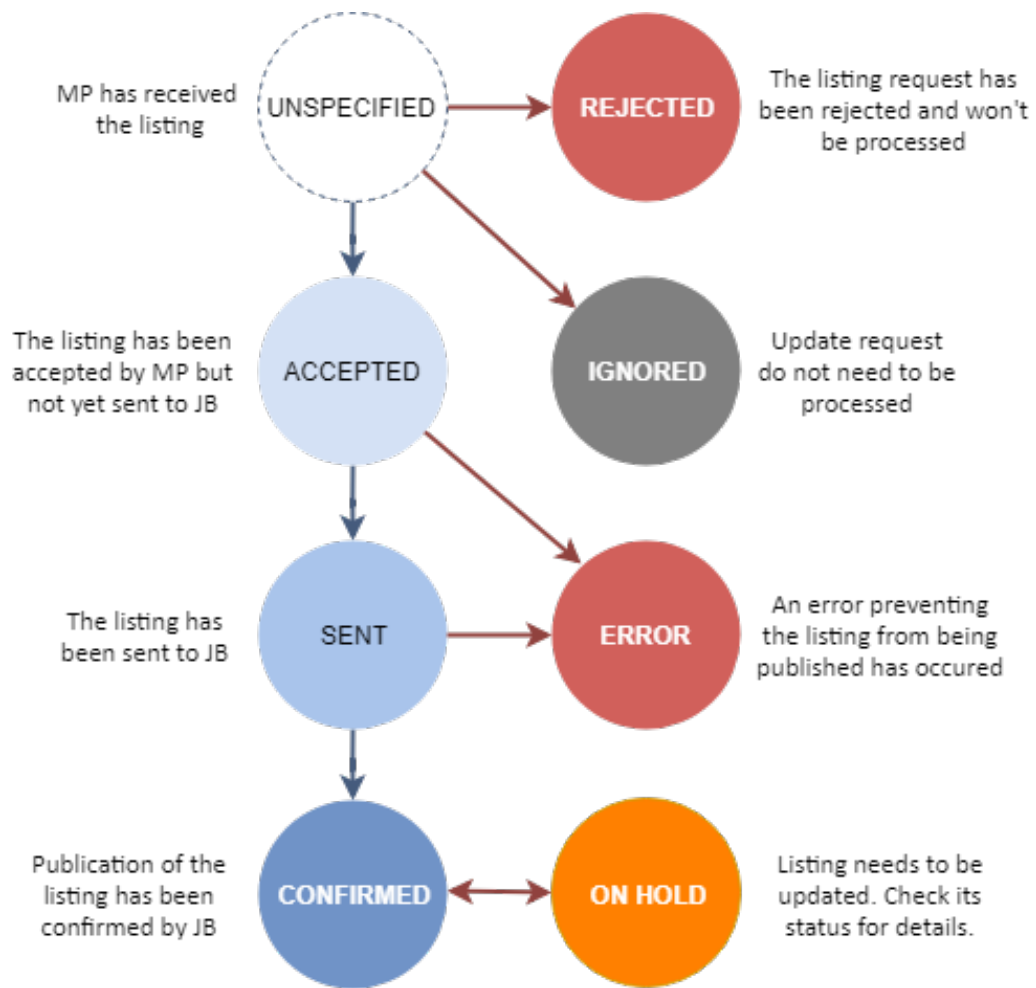
Field name	Type	Optional?
messages.action	String	
messages.status	String	
messages.timeReceived	Date	
messages.statusDescription	String	
messages.referenceId	String	yes
messages.author	String	
state	String	
stateDescription	String	yes
jobBoardId	number	
jobBoardUrl	String	yes
jobBoardReferenceId	String	yes
expirationTime	Date	yes
publicationTime	Date	yes
trackingLink	String	yes
description	String	yes
resultCode	number	

Expiration date is based on the standard publishing length for selected job board and the time publishing was confirmed. It is set only if listing was published.

The action field can have the following values: CREATE, UPDATE, DELETE.

The status field can have the following values: see [6.8.3](#)

6.8.3 Statuses



6.9 Credit status

Service returning credits and durations for selected customer (query parameter customerId).

Field "credits" contains status of all credits available to the customer.

Field "durations" contains durations of listings posted on particular job board. This field contains only job boards customer can post to.

URL	https://flashpost.the-network.com/credits
Method	GET
Encoding	UTF-8
Content type	application/json
URL Parameters	customerId – long (required)

6.9.1 Paylod

There's no payload for this service.

6.9.2 Success response

```
{
  "resultCode": 0,
  "description": "",
  "credits": [
    {
      "productType": "Local stepstone.de",
      "remainingCredits": 100,
      "consumedCredits": 42
    },
    {
      "productType": "GFP2",
      "remainingCredits": 20,
      "consumedCredits": 21
    }
  ],
  "durations": [
    {
      "jobBoardId": 21,
      "jobBoardName": "stepstone.at",
      "durationInDays": 30
    },
    {
      "jobBoardId": 22,
      "jobBoardName": "stepstone.de",
      "durationInDays": 30
    }
  ]
}
```

6.10 Software Development Kit (SDK)

Provider services are available also through FlashPost Software Development Kit for Java. Please contact our support for jar library and documentation.

6.10.1 SDK Example

```
FlashPostClient client = FlashPostClient.builder()
    .credentials(username, password)
    .build();
ListingCreateRequest request = /* prepare request with all fields here */;
ListingRequestResponse response = client.getListingService().createListing(req);

if (response.getResultCode().isSuccess()) {
    System.out.println("Listing created, request id=" + response.getRequestId());
}
```

7 Job Board services

7.1 JobSON

JobSON is a standard integration for Multiposting Tool. The validations for JobSON are listed in appendix C. JobSON uses Multiposting standard categories. The list is available through the categories service – see section 6.6. More detailed examples of JobSON requests are listed in appendix ??

7.1.1 Security

JobSON use HTTP Basic Authentication and connection over SSL.

7.1.2 Create listing

The payload is send to given URL using HTTP POST.

7.1.2 Payload

↪ Payload

7.1.2 Success response

In case of success reponse, the Job Board must provide HTTP status code 200 and urlOnJobBoard parameter – which is the link to URL to listing on Job Board. Multiposting tool expects the following payload:

```
{
  status: "CONFIRMED",
  urlOnJobBoard: "http://jobboard.com/1235"
}
```

Status field can have the following values:

ACCEPTED - the listing has been accepted by job board but not published yet

CONFIRMED - the listing has been published; in such case urlOnJobBoard parameter is mandatory

7.1.2 Error response

In case of error response, the Job Board must provide HTTP status code 4xx or 5xx and human readable error description. Multiposting tool expects the following payload:

```
{
  errorDescription: "Human readable error description or empty."
}
```

7.1.3 Update listing

The payload is send to given URL using HTTP PUT.

7.1.3 Payload

↪ Payload

7.1.3 Success response

In case of success reponse, the Job Board must provide HTTP status code 200 and urlOnJobBoard parameter – which is the link to URL to listing on Job Board. Multiposting tool expects the following payload:

```
{
  status: "CONFIRMED",
  urlOnJobBoard: "http://jobboard.com/1235"
}
```


Status field can the following values:

ACCEPTED - the listing has been accepted by job board but not published yet

CONFIRMED - the listing has been published; in such case `urlOnJobBoard` parameter is mandatory

7.1.3 Error response

In case of error response, the Job Board must provide HTTP status code 4xx or 5xx and human readable error description. Multiposting tool expects the following payload:

```
{
  errorDescription: "Human readable error description or empty."
}
```

7.1.4 Delete listing

The payload is send to given URL using HTTP DELETE.

7.1.4 Payload

↔ Payload

7.1.4 Success response

In case of success response, the Job Board must provide HTTP status code 200 – which is the link to URL to listing on Job Board. No payload is expected.

7.1.4 Error response

In case of error response, the Job Board must provide HTTP status code 4xx or 5xx and human readable error description. Multiposting tool expects the following payload:

```
{
  errorDescription: "Human readable error description or empty."
}
```

7.2 Confirmation

Job board can confirm successful posting or report an error for particular listing.

7.2.1 Success

In case of successful posting of listing, job board can provide information about the status.

URL	<code>https://flashpost.the-network.com/confirmation/success</code>
Method	POST
Encoding	UTF-8
Content type	application/json
URL Parameters	actionGuid – String (required)

7.2.1 Payload

```
{
  "referenceId": "123",
  "urlOnJobBoard": "http://somejobboard.com/101.html"
}
```

Field name	Type	Mandatory?	Content restrictions
referenceId	String		Job Board's reference id.
urlOnJobBoard	String	Yes	URL to the listing online.

7.2.1 Success response

```
{
  resultCode : 0,
  description : "Posting success confirmed."
}
```

7.2.1 Error response

In case of error response the JSON message looks the same as for the success response but the error code is one from the appendix A. The description contains human readable error description.

7.2.2 Error

In case of unsuccessful posting of listing, job board can provide information about the status.

URL	<code>https://flashpost.the-network.com/confirmation/error</code>
Method	POST
Encoding	UTF-8
Content type	application/json
URL Parameters	actionGuid – String (required)

7.2.2 Payload

```
{  
  "errorDescription": "Some error description"  
}
```

7.2.2 Success response

```
{  
  resultCode : 0,  
  description : "Posting error reported."  
}
```

7.2.2 Error response

In case of error response the JSON message looks the same as for the success response but the error code is one from the [appendix A](#). The description contains human readable error description.

7.3 Statistical data

Job board can insert own statistical data to improve quality of such data collected by Multiposting statistics service. In case of the data inconsistency, value delivered by a job board is taken only if it's greater than one gathered by Multiposting service. To access service basic auth needs to be used - credentials will be provided by The Network's support. Service will try to update as much entries as possible in case of a problem with single listing entry others will still be processed and statistics will be updated.

URL	https://flashpost.the-network.com/stats/update
Method	POST
Encoding	UTF-8
Content type	application/json

7.3.1 Payload

↔ Payload

Field name	Type	Mandatory?	Content restrictions
jobBoardId	number	Only if jobBoardReferenceld is provided	The value is provided by The Network.
listings	entry for single listing	yes	must contain only one of following fields: requestId, jobBoardReferenceld
requestId	number	Only if jobBoardReferenceld is not provided	Listing identifier in Multiposting system
jobBoardReferenceld	String	Only if requestId is not provided	Listing identifier in the job board.
applications	number		Number of times someone applied to the listing
clicks	number		Number of times someone started application process
views	number		Number of times someone viewed listing details

7.3.2 Success response

```
{
  "resultCode": 0,
  "description": "Statistics updated"
}
```

7.3.3 Error response

```
{
  "resultCode": 9,
  "description": "No statistics were updated"
  "errors" : [
    { "description" : "No listing with jobBoardReferenceId ABCD" }
    { "description" : "No listing with requestId 1234" }
  ]
}
```

The descriptions contains human readable error description.

A Standard Response Codes

The following list contains of all error codes responded by Multiposting tool.

0	OK
-1	UNKNOWN_ERROR
-100	REQUEST_VALIDATION_ERROR
-101	EMPTY_REQUEST_ERROR
-102	REQUEST_PARSING_ERROR
-103	USER_UNAUTHORIZED
-104	ACTION_NOT_ALLOWED_ERROR
-105	RESOURCE_NOT_FOUND
-200	CREDITS_NOT_FOUND
-300	ERRORS_IN_BATCH_OPERATION
-400	EMAIL_ALREADY_USED
-504	SERVICE_UNAVAILABLE

B Job Board Validations

↔ Job boards' validation rules

C JobSON validations

↪ JobSON validation rules