

Digital Bees

REST-API Service Documentation

VERSION 1.0



SOMMARIO

1	Introduction.....	3
2	Domains and platforms	3
3	VIDEO API	4
3.1	Server-Server requests	4
3.1.1	Authentication	4
3.1.1	LIST requests	4
3.1.2	SEARCH requests.....	7
3.1.3	VIEWS requests	10
3.1.1	MATCH requests	13
3.1.1	Errors code	14
3.2	Client-Server requests	15
3.2.1	Public Methods	17



1 INTRODUCTION

Digital Bees API (**DBAPI** from now on) is a restricted REST service developed to serve Digital Bees (**DB** from now on) **artists** and **video** information to its partners.

The service is based on server side connection between partners server and DBAPI servers and between clients and DBAPI servers.

2 DOMAINS AND PLATFORMS

In order to guarantee the best service to all our customers, we offers 2 separate environments for our API: Stage and Production.

Digital Bees provide to every customer stage and production credentials. **The credentials of stage and production are different.**

Customers can use the staging environment for testing, while they may use the production environment for their own production environment.

Stage and Production Digital Bees API environments have different domains, so both calls server-server and client-server must respect these domains.

Stage Environment: **stage.api.digitalbees.it**

Production Environment: **api.digitalbees.it**



3 VIDEO API

3.1 SERVER-SERVER REQUESTS

Server-Server request to fetch all the available data from DB video repository.

The requests use the HTTP communication protocol on default port 80.

3.1.1 AUTHENTICATION

In order to access this service any partner need to own an authentication pair ("apikey", "apisecret"), those keys will be manually provided you by DigitalBees.

Those parameters have to be appended on every request as query string:

```
&apikey=< value >&apisecret=< value >
```

3.1.1 LIST REQUESTS

Response to an index request is an array listing the matching objects.

List of Author	http://api.digitalbees.it/list/author
List of Videos	http://api.digitalbees.it/list/video

GET **/list/author**

HTTP
200



/list/author?page=1&perPage=100&sortProperty=creationDate&sortDirection=DESC&apikey=<value>&apisecret=<value>

perPage	Optional	integer	perPage is the number of items returned. Default and max value is 100	1
page	Optional	integer	is the page number. The first page is 1. Default page value is 1	100
sortProperty	Optional	enum	sortProperty accept a fields as property for sorting. You can sort by one of these fields : <ul style="list-style-type: none"> • id • title • creationDate • latModifiedDate • publishedDate Default field value is id	id
sortDirection	Optional	enum	use sortDirection to obtain ascending (ASC) or descending (DSC) order. Default value is DESC	DESC

GET **/list/video** **HTTP 200**

/list/video?page=1&perPage=100&sortProperty=creationDate&sortDirection=DESC&apikey=<value>&apisecret=<value>

perPage	Optional	integer	perPage is the number of items returned. Default and max value is 100	1
page	Optional	integer	is the page number. The first page is 1. Default page value is 1	100
sortProperty	Optional	enum	sortProperty accept a fields as property for sorting.	id



You can sort by one of these fields :

- id
- title
- creationDate
- latModifiedDate
- publishedDate

Default field value is id

sortDirection	Optional	enum	use sortDirection to obtain ascending (ASC) or descending (DSC) order. Default value is DESC	DESC
----------------------	----------	-------------	---	------



3.1.2 SEARCH REQUESTS

Response to an index request is an array listing the matching objects.

Search Video	http://api.digitalbees.it/search/video
Search Author	http://api.digitalbees.it/search/author

GET	/search/video	HTTP 200
/search/video?q=@name%20call%me%maybe&apikey=<value>&apisecret=<value>		
q	Required	string @<field><space><string to search> allowed fields: <ul style="list-style-type: none"> • name • title • metatag eg: <a href="http://api.digitalbees.it/search/video?q=@metatag%20vasco%20rossi&apikey=<value>&apisecret=<value>">http://api.digitalbees.it/search/video?q=@metatag%20vasco%20rossi&apikey=<value>&apisecret=<value>
perPage	Optional	integer perPage is the number of items returned. Default and max value is 100
page	Optional	integer is the page number. The first page is 1. Default page value is 1
sortProperty	Optional	enum sortProperty accept a fields as property for sorting. You can sort by one of these fields : <ul style="list-style-type: none"> • id • title • creationDate • latModifiedDate



- publishedDate

Default field value is id

sortDirection	Optional	enum	use sortDirection to obtain ascending (ASC) or descending (DESC) order. Default value is DESC	DESC
---------------	----------	------	--	------

GET	/search/author			HTTP 200
/search/author?q=@name%20carl%20rae%20jepson&apikey=<value>&apisecret=<value>				
q	Required	string	@<field><space><string to search> allowed fields: <ul style="list-style-type: none">name eg: http://api.digitalbees.it/search/video?q=@name%20vasco%20rossi&apikey=<value>&apisecret=<value>	
perPage	Optional	integer	perPage is the number of items returned. Default and max value is 100	1
page	Optional	integer	is the page number. The first page is 1. Default page value is 1	100
sortProperty	Optional	enum	sortProperty accept a fields as property for sorting. You can sort by one of these fields : <ul style="list-style-type: none">idnamecreationDatelatModifiedDatepublishedDate Default field value is id	id



sortDirection	Optional	enum	use sortDirection to obtain ascending (ASC) or descending (DESC) order. Default value is DESC	DESC
----------------------	----------	-------------	--	------



3.1.3 VIEWS REQUESTS

Match Video by Author	http://api.digitalbees.it/view/video-by-author?id=[int]
Match Video from date	http://api.digitalbees.it/view/video-from-date?from=53463453&to=90045432523

GET	/view/video-by-author?id=n			HTTP 200
/view/video-by-author?id=1567&apikey=<value>&apisecret=<value>				
id	Required	integer		1564
perPage	Optional	integer	perPage is the number of items returned. Default and max value is 100	100
page	Optional	integer	is the page number. The first page is 1. Default page value is 1	1
sortProperty	Optional	enum	sortProperty accept a fields as property for sorting. You can sort by one of these fields : <ul style="list-style-type: none">• id• title• creationDate• latModifiedDate• publishedDate Default field value is id	id
sortDirection	Optional	enum	use sortDirection to obtain ascending (ASC) or descending (DSC) order. Default value is DESC	DESC

GET	/view/video-by-date?from<start-timecode>to<end-timecode>	HTTP 200
<a href="/view/video-by-date?from<timecode>to<timecode>&apikey=<value>&apisecret=<value>">/view/video-by-date?from<timecode>to<timecode>&apikey=<value>&apisecret=<value>		



start-timecode	Required	integer	It indicates the start date for the video search data range. The correct response (HTTP code 200) return all videos created in the specified date range (from start to end). <u>The timecode format is Unix timestamp GMT.</u> <u>The start timecode value is required.</u> If you need further information about unix timestamp, please check http://www.unixtimestamp.com	1380603600
end-timecode	Optional	integer	It indicates the end date for the video search data range. The correct response (HTTP code 200) return all videos created in the specified date range (from start to end). <u>The timecode format is Unix timestamp GMT.</u> The end timecode is optional, if omitted the response return all video created from start timecode to the current timecode. If you need further information about unix timestamp, please check http://www.unixtimestamp.com	1382371200
perPage	Optional	integer	perPage is the number of items returned. Default and max value is 100	100
page	Optional	integer	is the page number. The first page is 1. Default page value is 1	1
sortProperty	Optional	enum	sortProperty accept a fields as property for sorting. You can sort by one of these fields : <ul style="list-style-type: none"> • id • title • creationDate • latModifiedDate 	id



- publishedDate

Default field value is id

sortDirection	Optional	enum	use sortDirection to obtain ascending (ASC) or descending (DESC) order. Default value is DESC	DESC
----------------------	----------	-------------	--	------



3.1.1 MATCH REQUESTS

Match Video by id	http://api.digitalbees.it/get/video/id=[int]
Match Author by id	http://api.digitalbees.it/get/author/id=[int]

GET	/get/video/<id>	HTTP 200
get/video/<id>?apikey=<value>&apisecret=<value>		
id	Required	Number id must be a valid video id 400100

GET	/get/author/<id>	HTTP 200
get/author/<id>?apikey=<value>&apisecret=<value>		
id	Required	Number id must be a valid author id 203



3.1.1 ERRORS CODE

ERROR CODE	ERROR MESSAGE
400	Bad request
403	Forbidden
404	Not found
500	Internal Server Error
502	Bad Gateway



3.2 CLIENT-SERVER REQUESTS

In order to show a video player inside a web page you need to call our js library (DigitalBees SDK) from your web page (client), so this is client-server request to the digitalbees api system.

When invoked, the SDK application read at bootstrap the first element of "_sqrapiq" array that must be set to "init", as showed below:

```
_sqrapiq.push(['init', apiKey, callback]);
```

This method initializes the sdk reading the user configuration on the digitalbees platform (server-side), this means that the apiKey value must correspond to an existing USER on digitalbees.

The 'init' method replaces the array "_sqrapiq" with an object "ExternalInterface" that exposes the public methods of the SDK that you can invoke using the "push" method

example:

```
_sqrapiq.push ( [ 'setOption' , 'autoStart' , true ] );
```

Alternatively, you can directly retrieve the object ExternalInterface to invoke methods in the traditional way.

Example:

```
_sqrapiq.getApi ().searchVideo ('O Vient' , 1, function(data){console.log(data)});
```



Below you can find the digitalbees client-side code example to include in your webpage:

```
<script type="text/javascript">
  var _srapiq = _srapiq || [];
  _srapiq.push(['init', apiKey, <your_callback_method>]); //
Required
  _srapiq.push(['setOption', 'width', 600]); // Optional
  _srapiq.push(['setOption', 'height', 400]); // Optional
  _srapiq.push(['setOption', 'autoStart', true]); // Optional
  _srapiq.push(['setOption', 'advUrl', advURL]); // Optional
  _srapiq.push(['embedVideo', 'videoContainer', '1']); // Required

  (function() {
    var sr = document.createElement('script');
    sr.type = 'text/javascript'; sr.async = true;
    sr.src = '//sdk.digitalbees.it/latest.js';
    var s = document.getElementsByTagName('script')[0];
    s.parentNode.insertBefore(sr, s);
  }) ();
</script>
```

the example above contains comments showing id the method is optional or mandatory.



3.2.1 PUBLIC METHODS

IMPORTANT: Init method must be the first call to the SDK.

METHOD	PROPERTY	INPUT VALUE	DESCRIPTION
<code>init(apiKEY, <your callback>);</code>	apiKEY	string	your apikey
	<your callback>		you can specify your callback function to have a notification when the the player template is ready
<code>setOption(optKey, optValue);</code>			use this method to set the player options as height, width, autostart, advUrl example: _srapiq.push(['setOption', 'width', 600]);
	height	integer	
	width	integer	



	<code>autoStart</code>	boolean	
	<code>advUrl</code>	string	
<code>getVideoDuration();</code>			return the video duration of the last video player embedded in web page
<code>embedVideo(parentElement, videoAssetId);</code>			enables the Video Player in web page. If your SRAPI account is enabled from Digital Bees staff to manage Video, you must use this method to enableVideos in your web page
	<code>parentElement</code>	string	parentElement is the id or the instance of the dom element where you want append the video player
	<code>videoAssetId</code>	integer	a valid video id
<code>loadVideoById(videoAssetId, <your-callback>);</code>			use this method to change the video asset inside the last player embedded in web page
	<code>videoAssetId</code>	integer	set the width for the video player in your web page



<code><your-callback></code>		you can specify your callback function to have a notification when the the player template is ready
<code>playVideo() ;</code>		use this method to set play the last player embedded in web page
<code>stopVideo() ;</code>		use this method to set stop the last player embedded in web page
<code>pauseVideo() ;</code>		use this method to set pause the last player embedded in web page
<code>searchVideo(searchTxt,limit, <your-callback>) ;</code>		use this method to search a video by name or by author o metatag
<code>searchTxt</code>	string	the text string that you want search (author name, video title or metatag)
<code>limit</code>	integer	the max number of results you want obtain as response
<code><your-callback></code>		because the search is asynchronus, you can specify your callback function to have a notification when the the search results are ready

