idomoo

Videos That Get Personal

VIDEO GENERATION

API Documentation

TABLE OF CONTENTS

Copyright © Idomoo 2013. All Rights Reserved. Reproduction or distribution other than for intended purposes is prohibited, without the prior written consent of Idomoo. These materials are confidential, and shall be returned to Idomoo upon request.

1. OVERVIEW

1.1. Introduction

Idomoo is the leader in personalized video, enabling customer communication and direct marketing using automatically generated video.

A short video can effectively communicate a complex message, create emotions and drive customers to action. Whether your goal is customer satisfaction, sales or cost reduction, video is the most effective way to personalize communication with impact.

Combining quality video with personalization increases the visibility and appeal of your message, maximizes customer engagement, and improves responsiveness.

Idomoo's Video as a Service (VaaS) platform receives customer data from standard CRM systems or databases and generates millions of tailor-made, high quality, professional video clips.

By automatically integrating personal customer information within compelling videos, and distributing these videos via a range of convenient channels from email to web to mobile, Idomoo enables companies to communicate in a fun and personal manner.

Following successful deployments with customers worldwide, Idomoo has proven that its solution lowers service costs, improves customer satisfaction and increases average revenue per customer.

1.2. Purpose

This document depicts the workflow of operating Idomoo's video platform API for generating On-The-Fly or On-Demand Video content.

2. API INTERFACE

The following paragraphs describe the various aspects involved in interfacing with Idomoo's API.

2.1. API End-Points

Idomoo's API currently hosts several End-Points.

Environment	Supports HTTPs	FQDN
Test	Yes	http://api-dev.idomoo.com
Production – EU	Yes	http://api-eu.idomoo.com
Production – US	Yes	http://api-us.idomoo.com

The Test environment is currently only used for testing against Idomoo's demo template in order to test the entire video generation roundtrip. The Production End-Points will be used to serve functional (processed via Full or Self Service) templates only

2.2. API Video Generation Round Trip Test

In order to test the integration of your/your client's platform with Idomoo's API prior to having your video template ready on Idomoo's platform, you can generate test videos with Idomoo's test video template (accessed by specifying the ***pid*** parameter with a value of ***147*** when testing against the Test end point). All of the examples in this document are live working examples that generate videos using the above data.

3. API REQUEST SPECIFICATION

Idomoo's API supports various request types in order to fulfil different functionalities.

3.1. Clip Generation Action

This action is used in order to trigger video generation according to data passed in the request itself. Data posted is composed of the following sections:

3.1.1. Action HTTP Path

http://api-dev.idomoo.com/ext/cg.action

3.1.2. Action Request Formats

There are several formats to send via POST request to Idomoo's API. The platform will differentiate between these formats according to the "Content-Type" HTTP header set in the request. Failure to set the "Content-Type" in accordance with the table below will result in a request validation error:

Content	Content-Type Header	POST Body Example(Formatted)
JSON Document	application/json	*{* *"clips":[* *{* *"pid":147,* *"da":[* *{* *"key":"TEST1",* *"val":"Text1"* *},* *{* *"key":"TEST2",* *"val":"Text2"* *},* *{* *"key":"TEST3",* *"val":"Text3"* *},* *{* *"key":"TEST4",* *"val":"Text4"* *},* *{* *"key":"TEST5",* *"val":"Text5"* *},* *{* *"key":"TEST6",* *"val":"Text6"* *},* *{* *"key":"TEST7",* *"val":"Text7"*

		<pre>{
		},
		{
		"key":"TEST8",
		"val":"TextT8"
		}
		],
		"of":[
		{
		"nam":"HLS_VB_500"
		}
		],
		"kc":8
		}
		],
		"cc":1,
		"rf":"json"
		}</pre>

XML Document	application/xml	<pre><request>
		<cc>1</cc>
		<clips>
		<clip>
		<da>
		<dae key="TEST1" val="Text1"/>
		<dae key="TEST2" val="Text2"/>
		<dae key="TEST3" val="Text3"/>
		<dae key="TEST4" val="Text4"/>
		<dae key="TEST5" val="Text5"/>
		<dae key="TEST6" val="Text6"/>
		<dae key="TEST7" val="Text7"/>
		<dae key="TEST8" val="Text8"/>
		</da>
		<kc>8</kc>
		<of>
		<ofe nam="HLS_VB_500"/>
		</of>
		<pid>147</pid>
		</clip>
		</clips>
		<rf>json</rf>
		</request></pre>

3.1.3. Action Response Formats

Each request sent to Idomoo's API has its response format set according to the value of the "rf" parameter, regardless of which input format was chosen by the request originator. The following table describes the various output formats provided by Idomoo's API:

"rf" parameter value	Returned content	Example of successful response	Failure response format	Example of failure response
rvd	An HTTP 302 redirect to the media file generated by the request (M3U8 manifest for HLS, mp4 file for MP4,etc...)	http://ap-dev.idomoo.com/vd/140/1da51b8d8ff98f6a48f01ae79fe3ca6c26e1abb7b7d125259255d6d2b875ea08/manifest.m3u8	xml	<response> <status>ALL_CLIPS_GENERATION_FAILED</status> <statusDescription> 'kc' param pased must be equal to the number of key-value pairs passed inside 'da' </statusDescription> <projectId>140</projectId> </response>

json	A JSON formatted document containing generation information regarding the requested video/s	*{
		"status":"ALL_CLIPS_GENERATION_SUCCEEDED",
		"projectId":140,
		"clips":[
		{
		"outputs":[
		{
		"format":"HLS_VB_500",
		"length":"401",
		"url":"http://api-dev.idomoo.com/vd/140/8241649609f88ccd2a0a5b233a07a538ec313ff6adf695aa44a969dbca39f67d/manifest.m3u8",
		"redirectable":true
		}
		],
		"internalCode":0,
		"internalId":1123
		}
		]
		}*
			json	*{
				"status": "ALL_CLIPS_GENERATION_FAILED",
				"statusDescription": "'kc' param pased must be equal to the number of key-value pairs passed inside 'da'",
				"projectId": 140
				}*

Xml	An XML formatted document containing generation information regarding the request video/s	*<response> <status>ALL_CLIPS_GENERATION_SUCCEEDED</status> <projectId>140</projectId> <clips> <clip> <internalCode>0</internalCode> <internalId>12429</internalId> <outputs> <output> <format>HLS_VB_500</format> <length>401</length> <url> http://api-dev.idomoo.com/vd/140/6e4001871c0cf27c7634ef1dc478408f642410fd3a444e2a88e301f5c4a35a4d/manifest.m3u8 </url> </output> </outputs> </clip> </clips> </response>*	xml	*<response> <status>ALL_CLIPS_GENERATION_FAILED</status> <statusDescription> 'kc' param pased must be equal to the number of key-value pairs passed inside 'da' </statusDescription> <projectId>140</projectId> </response>*

3.1.4. Action parameters

3.1.4.1. Request Scope parameters

This section refers to parameters present in the scope of the entire request.

API Param Name	Description	Possible Values	Default Value	Mandatory	Dependencies/ Restrictions
cc	Contains the clip count to be generated.	Any positive integer	N/A	Only when sending POST requests in JSON/XML format	Value must match the Number of “Clip Data Nodes” sent via the POST request

jp	Add JSONP support and pad response with the value of this parameter	Any ASCII String containing alpha-numeric characters	N/A	No	Only when “json” is used for the “rf” parameter
rf	Response format. Determines the formatting and names of which the clip generated will be returned to the request originator	rvd,json,xml See “Response Table” below for detailed description of the each possible value	Xml	Yes	Using or “rvd” is restricted to a single clip in a generation request , meaning “cc” parameter value must be to 1 when sending JSON or XML formatted POST requests
clips	Structured Clip Generation Data -	Structured data passed in accordance to matching Input format (JSON or XML): **JSON Example :** *clips : [{ “pid” : “110” , : “rf : ”json” , “kc” : 7 , “of” : [{“nam” : “HLS_VB_500”}] , “da” : [{ “key” : “a” , “val” : “b”} , {“key” : “c” , “val” : “2”}] }]* **XML Example:** *<clips><clip><pid>110</pid> <rf >json</rf><kc>2</kc> <of><ofe nam=”HLS_VB_500”/></of> <da><dae key=“a” val=“b”/><dae key=“c” val=“2”/> </da></clip></clips>*	N/A	Yes	Amount of “Clip” nodes specified here must match the value of the ***cc*** parameter

3.1.4.2. Clip Scope parameters

This section refers to parameters present in the scope of the clip data. Several clip sections can be present in a single request, each requires specifying the parameters in accordance to the following table

API Param Name	Description	Possible Values	Default Value	Mandatory	Dependencies/ Restrictions
pid	Project/Template Id , as given by Idomoo's support team or via the IWS platform	Any positive integer	N/A	Yes	Must match the Project/Template ID , as handed by Idomoo's support team or via the IWS platform
uid	Account Id , as given by Idomoo's support team or via the IWS platform	Any positive integer	N/A	No	Must match the User/Account ID , as handed by Idomoo's support team or via the IWS platform
da	Structured Clip Data - a structured array/collection of Key/Value pairs	Structured data passed in accordance to matching Input format (JSON or XML) :	N/A	Only when rda is not used	Key names and their corresponding values must be passed and will be treated as strings.

	holding all of the required data needed to generate the clip	**JSON Example :** *da : [{ "key" : "a" , : "val" : "b"} , { "key" : "c" , : "val" : "2"}]* **XML Example:** *<da><dae key="a" val="b"/> <dae key="c" val="2"/></da>*			Pair count must match the value specified using the ***kc.*** This parameter is mutually exclusive with the ***rda*** parameter Specifying at least one is mandatory
rda	Raw Clip Data – a UTF-8 encoded String of raw data delivered by the user	Any UTF-8 Encoded String	N/A	Only when da is not used. **This parameter cannot be used without prior coordination with idomoo's support teams**	Key names and their corresponding values must be passed and will be treated as strings. Pair count must match the value specified using the ***kc.*** This parameter is mutually exclusive with the ***da*** parameter. Specifying at least one is mandatory. This parameter is only supported for Full-Service customers. Idomoo favors the usage of the ***da*** parameter in all cases.
of	Output Format - contains a structured array/collection of Output Format specifications	Structured data passed in accordance to matching Input format (JSON or XML) . **JSON Example :** *of : [{ "nam" : "HLS_VB_500" }]* **XML Example:** *<of><ofe nam=" HLS_VB_500" /></of>* Certain formats , such as image formats , require additional arguments passed in the following manner : **JSON Example :** *of : [* *{* *"nam" :* *"JPEG_MAIN_THUMBNAIL" ,* *"eps" : [* *{* *"ep" : "00:00:04" ,* *"ix" : 1* *}* *]* *}* *]* **XML Example:** *<of>* *<ofe nam="* *JPEG_MAIN_THUMBNAIL" >* *<eps>* *<epse ep="00:00:04"*	N/A	Yes	Output Format names (specified via the ***nam*** key) used must be specified according to Output Format Specification section

		ix="1"/> *</eps>* *</ofe>* *</of>*			
kc	Contains the Key/Value pairs Count expected inside the ***da*** parameter	Any positive integer	N/A	Only when the ***da*** is parameter is used. If the ***rda*** parameter is used instead this parameter must be omitted.	Value must match the number of Key/Value pairs found in the ***da*** parameter
exs	Use External Storage. This parameter indicate whether video content generated should be uploaded to external storage services such as Amazon S3 or reside on the API Platform storage	Boolean Value(true/false) True implies the video content generated should reside on the API Platform Storage False implies the video content generated should be uploaded to external storage service after being generated successfully	false	No **This parameter cannot be used without prior coordination with idomoo's support teams**	Using external storage requires manual configuration on Idomoo's backend and is available to Full-Service customers only
eid	Clip "External ID". This Identifier will be used to as part of the URL identifier of the clip to be generated. Omitting this parameter will cause a self-generated "hashed" external ID to be used	Any string of UTF-8 encoded string	N/A	No	ID given will be used "as-is" , with no additional encryption/Hashing
eidl	Clip "External ID" length.	0 - Do not trim the Clip External ID (will be of 32 bytes in length) 7-31 - Trim the clip external ID to be in matching length to the value of this parameter	0	No	Passing an value larger than 0 and smaller then 7 (non-inclusive) would result in an error
eidh	Clip "External ID" Hash	Boolean Value (true/false) True implies "External ID" provided should be hashed prior to incorporating it into the video link URL False implies "External ID" provided should be used as is when incorporating it into the video link URL	false	No	This parameter is only meaningful when eid params is used

3.1.5. Action Usage Examples

The following examples describe various use cases of the Clip Generation Action.

3.1.5.1. Single “On-The-Fly” Video Generation Request

Below is a single video generation request via XML POST, with the response format set to be JSON, Output Format will be HLS with a Video Bit Rate of 500 Kbps

```
POST /ext/cg.action HTTP/1.1
User-Agent: curl/7.27.0
Host: api-dev.idomoo.com
Content-Type:application/xml
Content-Length: 384

<request><cc>1</cc><clips><clip><da><dae key="TEST1" val="Text1"
/><dae key="TEST2" val="Text2" /><dae key="TEST3" val="Text3" /><dae
key="TEST4" val="Text4" /><dae key="TEST5" val="Text5" /><dae
key="TEST6" val="Text6" /><dae key="TEST7" val="Text7" /><dae
key="TEST8" val="Text8" /></da><kc>8</kc><of><ofe
nam="HLS_VB_500"
/></of><pid>147</pid></clip></clips><rf>json</rf></request>
```

3.1.5.2. Multiple “On-Demand” Video Generation Request

Below is a multiple videos generation request via JSON POST, with the response format set to be XML. The 1st video will be generated as an MP4 with a video bit-rate of 500 Kbps. The 2nd video will be generated as an MP4 formatted video in HD Ready resolution with a video bit-rate of of 800 kbps

3.1.5.3. Single “On-Demand” Video Generation Request with Multiple Output Formats

```
POST /ext/cg.action HTTP/1.1
User-Agent: curl/7.27.0
Host: api-dev.idomoo.com
Content-Type:application/json
Content-Length: 664

{"clips":[{"pid":147,"da":[{"key":"TEST1","val":"Text1"},{"key":"TEST2","val":"Text2"}
,{"key":"TEST3","val":"Text3"},{"key":"TEST4","val":"Text4"},{"key":"TEST5","val":"T
ext5"},{"key":"TEST6","val":"Text6"},{"key":"TEST7","val":"Text7"},{"key":"TEST8","v
al":"TextT8"}],"of":[{"nam":"MP4_VB_500"}],"kc":8},{"pid":147,"da":[{"key":"TEST9","
val":"Text9"},{"key":"TEST10","val":"Text10"},{"key":"TEST11","val":"Text11"},{"key"
:"TEST12","val":"Text12"},{"key":"TEST13","val":"Text13"},{"key":"TEST14","val":"Te
xt14"},{"key":"TEST15","val":"Text15"},{"key":"TEST16","val":"TextT16"}],"of":[{"nam
":"MP4_HD_READY_VB_800"}],"kc":8}],"cc":2 , "rf" : "xml"}
```

Below is a multiple videos generation request via JSON POST, with the response format set to be JSON. The 1st Output Format will an MP4 with a video bit-rate of 500 Kbps. The 2nd Output Format will be a JPEG Thumbnail, extracted from the 4th second of the video.

```
POST /ext/cg.action HTTP/1.1
User-Agent: curl/7.27.0
Host: api-dev.idomoo.com
Content-Type:application/json
Content-Length: 664

{"clips":[{"pid":147,"da":[{"key":"TEST1","val":"Text1"},{"key":"TEST2","val":"Text2"}
,{"key":"TEST3","val":"Text3"},{"key":"TEST4","val":"Text4"},{"key":"TEST5","val":"T
ext5"},{"key":"TEST6","val":"Text6"},{"key":"TEST7","val":"Text7"},{"key":"TEST8","v
al":"TextT8"}],"of":[{"nam":"MP4_VB_500"}, { "nam" : "JPEG_MAIN_THUMBNAIL" ,
"eps" : [{"ep" : "00:00:04" , "ix" : 1 }] ],"kc":8}],"cc":1 , "rf" : "json"}
```

3.2. Clip Seeding Action

This action is used in order to seed video generation data passed without actually generating the video. The caller then receives a URL which he could later use in order to trigger the video generation for the data seeded

3.2.1. Action HTTP Path

```
http://api-dev.idomoo.com/ext/cs.action
```

3.2.2. Action Request Formats

As this action differs from Clip Generation Action in the actual result of the action, it derives its request formats from it. For fully detailed information, click here

3.2.3. Action Response Formats

Each seeding request sent to Idomoo's API has its response format set according to the value of “srf” parameter, regardless of what input format was chosen by the request originator. The following table describes the various output formats provided by Idomoo's API:

"srf" parameter value	Returned Content	Example of Successful Response	Failure Response Format	Example of Failure Response
json	A JSON formatted document containing seeding information regarding the requested video/s	*{ "link":"http://api-dev.idomoo.com/seeds/ci.action?hrid=a6e446f1cde0b2caeefd5d8a6e6210bf", "status":"ALL_CLIPS_SEEDING_SUCCEEDED" }*	json	*{ "status":"ALL_CLIPS_SEEDING_FAILED", "statusDescription":"'kc' param pased must be equal to the number of key-value pairs passed inside 'da'" }*
Xml	An XML formatted document containing generation information regarding the request video/s	*<response> <status>ALL_CLIPS_SEEDING_SUCCEEDED</status> <link>http://api-dev.idomoo.com/seeds/ci.action?hrid=a6e446f1cde0b2caeefd5d8a6e6210bf</link> </response>*	xml	*<response> <status>ALL_CLIPS_SEEDING_FAILED</status> <statusDescription>'kc' param pased must be equal to the number of key-value pairs passed inside 'da'</statusDescription> </response>*

3.2.4. Action Parameters

This action uses the same data for structure and parameters used for the Clip Generation Action in order to seed the video generation data. For the fully detailed list, click here. Aside from these parameter, the following parameters are also used.

3.2.4.1. Request Scope parameters

This section refers to parameters present in the scope of the entire request.

API Param Name	Description	Possible Values	Default Value	Mandatory	Dependencies/ Restrictions
Srf	Seeding Response format. Determines the format in which the Clip Generation	json,xml See "Response Formats Table" for detailed description of each possible value	xml	Yes	Note that this refers only to the response format to be used for the seed request itself. The response format for the video generation request is still specified by the "***rf***" parameter

3.2.5. Action Usage Examples

The following examples describe a sample use case of the Clip Seeding Action.

3.2.5.1. Single Video Seeding Request

Below is a single video seeding request via JSON, with the response format set to be JSON , Seeding response format , Output Format will be HLS with a Video Bit Rate of 500 Kbps

```
POST /ext/cs.action HTTP/1.1
User-Agent: curl/7.27.0
Host: api-dev.idomoo.com
Content-Type:application/json
Content-Length: 346

{"clips":[{"pid":147,"da":[{"key":"TEST1","val":"Text1"},{"key":"TEST2","val":"Tex
t2"},{"key":"TEST3","val":"Text3"},{"key":"TEST4","val":"Text4"},{"key":"TEST5","
val":"Text5"},{"key":"TEST6","val":"Text6"},{"key":"TEST7","val":"Text7"},{"key":"
TEST8","val":"TextT8"}],"of":[{"nam":"HLS_VB_500"}],"kc":8}],"cc":1 , "srf" :
"json" , "rf" : "json"}
```

3.3. Clip Inflation Action

This action is used in order to initiate (“inflate”) video generation data previously seeded according to the given hash ID specified in the request URL. The video content generation data and response format are pre-determined according to the “seeded” data.

3.3.1. Action HTTP Path

```
http://api-dev.idomoo.com/ext/ci.action
```

3.3.2. Action Request Formats

This is action a Non-RESTful HTTP GET request, which implies the request body simply doesn’t exist. No other request format are supported.

3.3.3. Action Response Formats

As this action formats the response in accordance to the seeded data, it’s response formats are derived from those of the Clip Generation Action .For fully detailed information, click here

3.3.4. Action Parameters

As this action is called using HTTP GET method, it disallows complex structured data. Aside from key value parameters specified as HTTP query parameters .The following query parameters are currently supported:

API Param Name	Description	Possible Values	Default Value	Mandatory	Dependencies/ Restrictions
Hrid	Seed Hash Reference ID The value specified here should correspond to the seed ID provided when calling the Clip Seeding action	Any ASCIII Alpha-Numeric String	N/A	Yes	Specifying an invalid Seed ID will result in a 404 Failure

3.3.5. Action Usage Examples

The following examples describe a sample use case of the Clip Inflation Action.

3.3.5.1. Generic Video Inflation Request

Below is an inflation request meant to trigger video generation for Seed ID corresponding to "5f9535b9d310d3016e88e46c094f6f8b".

```
GET /seeds/ci.action?hrid=5f9535b9d310d3016e88e46c094f6f8b HTTP/1.1
User-Agent: Wget/1.14 (linux-gnu)
Accept: */*
Host: api-dev.idomoo.com
```

4. OUTPUT FORMAT SPECIFICATION

Idomoo's video generation API supports various video and image formats to be passed upon posting the video generation request.

4.1. Technical definitions

- Bit-rate - The maximum number of bits that can be used to represent data in each single second of the video. Increasing this number can drastically improve both visual and audible content quality, at the price of increasing storage volume and bandwidth required to store and stream each video. Idomoo's API currently supports passing several values for the video bit-rate , while the audio bit-rate remains constant.

- Video Codec - Encoding Library used to generate the visual content of the video. Idomoo currently uses H264 as its main video codec.

- Audio Codec - Encoding Library used to generate the audible content of the video. idomoo currently uses AAC as its main audio codec.

- Video/Image Format – The actual format of the generated video. Idomoo's API currently supports MP4 or HLS (m3u8 manifest file and mpegTS segments) as its video formats and JPEG as its Image format.

4.2. Technical Specification

The table below describes the output formats supported by Idomoo's API.

API ID	API Category	API Extra Parameters	Video/Image Format	Video Codec	Video Bit Rate	Video Dimensions	Audio Codec	Audio Bit Rate
HLS_HD_READY_VB_800	On-The-Fly	N/A	HLS	H264	800 kbps	HD Ready (1280x720)	AAC	128 kbps
HLS_VB_500	On-The-Fly	N/A	HLS	H264	500 kbps	Standard (640x360)	AAC	128 Kbps
MP4_HD_READY_VB_800	On-Demand	N/A	MP4	H264	800 kbps	HD Ready (1280x720)	AAC	128 kbps
MP4_VB_500	On-Demand	N/A	MP4	H264	500 kbps	Standard (640x360)	AAC	128 Kbps
JPEG_MAIN_THUMBNAIL_HD_READY	On-Demand	HH:MM:SS	JPEG	N/A	N/A	HD Ready (1280x720)	N/A	N/A
JPEG_MAIN_THUMBNAIL	On-Demand	HH:MM:SS	JPEG	N/A	N/A	Standard (640x360)	N/A	N/A

4.3. Technical Restrictions

- Output Formats from different categories cannot be used together in a single API request.

- Only a single “On-The-Fly” Output can be used in a single API request,

- Image formats require timestamp in the form HH:MM:SS in order to correctly choose the frame from which the main thumbnail image would be extracted

- Using any output format aside from ones listed below will result in video generation failure