

Table Of Contents

backend_api - backend_api	4
backend_api.account_billing - account_billing	4
backend_api.account_general - account_general	4
backend_api.account_plan - account_plan	4
backend_api.account_security - account_security	4
backend_api.add_tag - add_tag	4
backend_api.cdn_change_file - cdn_change_file	5
backend_api.cdn_check_folder - cdn_check_folder	5
backend_api.cdn_copy_file - cdn_copy_file	5
backend_api.cdn_create_folder - cdn_create_folder	5
backend_api.cdn_delete_file - cdn_delete_file	6
backend_api.cdn_delete_folder - cdn_delete_folder	6
backend_api.cdn_upload - cdn_upload	6
backend_api.check_can_share - check_can_share	6
backend_api.check_duplicate_folder - check_duplicate_folder	6
backend_api.check_folder_access - check_folder_access	7
backend_api.check_folder_owner - check_folder_owner	7
backend_api.check_hash - check_hash	7
backend_api.create_file - create_file	7
backend_api.create_folder - create_folder	7
backend_api.create_lead - create_lead	8
backend_api.default_folder - default_folder	8
backend_api.del_tag - del_tag	8
backend_api.delete - delete	8
backend_api.delete_file - delete_file	8
backend_api.delete_folder - delete_folder	9
backend_api.delete_share - delete_share	9
backend_api.documentation - Download API Documentation	9
backend_api.file_folder - return folder_id	9
backend_api.files - files	9
backend_api.fill_share - fill_share	10
backend_api.folder_tree - folder_tree	10
backend_api.format_bytes - format_bytes	10
backend_api.get_default_folder - get_default_folder	10
backend_api.get_parent - get_parent	11
backend_api.get_uid - get_uid	11
backend_api.get_user - get_user	11
backend_api.listing - listing	11
backend_api.move - move	12
backend_api.move_file - move_file	12
backend_api.move_folder - move_folder	12
backend_api.paginator - paginator	12
backend_api.plans_listing - plans_listing	12
backend_api.rabbit_task - rabbit_task	13
backend_api.rabbit_worker - rabbit_worker	13
backend_api.recursive_down - recursive_down	13
backend_api.recursive_folder_size - recursive_folder_size	13
backend_api.recursive_up - recursive_up	13
backend_api.refusal - refusal	14
backend_api.rename_file - rename_file	14
backend_api.rename_folder - rename_folder	14
backend_api.save - save	14
backend_api.save_file - save_file	15
backend_api.search - search	15
backend_api.send - send message	15
backend_api.share_folder - share_folder	15
backend_api.shares - shares	16
backend_api.sign_links - sign_links	16
backend_api.stop_sharing - share_folder	16
backend_api.subfolders - subfolders	16
backend_api.update_file - update_file	17
backend_api.update_folder - update_folder	17

backend_api.update_user - update_user	17
backend_api.used_storage - used_storage	18
backend_api.user_tokens - user_tokens	18

backend_api - backend_api

Functions of a general nature

backend_api.account_billing - account_billing

This method is used to get billing profile for user.

Parameter	Name	Required	Default	Description
uid	User id	Y		User id

backend_api.account_general - account_general

This method is used to get main user info.

Parameter	Name	Required	Default	Description
uid	User id	Y		User id

backend_api.account_plan - account_plan

This method is used to get user plan.

Parameter	Name	Required	Default	Description
uid	User id	Y		User id

backend_api.account_security - account_security

This method is used to get security user info.

Parameter	Name	Required	Default	Description
uid	User id	Y		User id

backend_api.add_tag - add_tag

This method is used to add tag(s) to file. Allow tags separated by space or ","

Parameter	Name	Required	Default	Description
file_id	file_id	Y		file id
tags	tags	Y		tag or multiple tags

Example Output

```
Return method 'listing' for current file folder;
```

backend_api.cdn_change_file - cdn_change_file

This method is used to change file in CDN storage.

Parameter	Name	Required	Default	Description
uid	User id			User id
sha1	File SHA1			File SHA1 hash
autodownload	File autodownload			File autodownload attribute
is_viewable	File is_viewable			File is_viewable attribute
is_hidden	File is_hidden			File is_hidden attribute
filename	Filename			CDN filename
cdn_file_id	CDN file_id	Y		CDN file id

backend_api.cdn_check_folder - cdn_check_folder

This method is used to check CDN folder.

Parameter	Name	Required	Default	Description
foldername	foldername			foldername

backend_api.cdn_copy_file - cdn_copy_file

This method is used to make copy file by sha1 hash in CDN storage.

Parameter	Name	Required	Default	Description
uid	User id	Y		User id
sha1	File SHA1			File SHA1 hash
autodownload	File autodownload		1	File autodownload attribute
is_viewable	File is_viewable		1	File is_viewable attribute
is_hidden	File is_hidden		1	File is_hidden attribute
filename	Filename	Y		filename
cdn_folder_id	CDN folder_id	Y		CDN folder id

backend_api.cdn_create_folder - cdn_create_folder

This method is used to create folder in CDN storage.

Parameter	Name	Required	Default	Description
uid	User id	Y		User id
foldername	Foldername	Y		foldername
folder_id	Parent folder id			parent folder_id

backend_api.cdn_delete_file - cdn_delete_file

This method is used to delete file from CDN storage.

Parameter	Name	Required	Default	Description
uid	User id	Y		User id
cdn_file_id	CDN File id			cdn_file_id

backend_api.cdn_delete_folder - cdn_delete_folder

This method is used to delete folder in CDN storage.

Parameter	Name	Required	Default	Description
uid	User id	Y		User id
cdn_folder_id	CDN Folder id	Y		cdn_folder_id

backend_api.cdn_upload - cdn_upload

This method is used to upload local file to CDN storage.

Parameter	Name	Required	Default	Description
uid	User id	Y		User id
filepath	Filepath	Y		local path to file
filename	Filename	Y		filename
cdn_folder_id	cdn_folder_id	Y		CDN folder id

backend_api.check_can_share - check_can_share

This method is used to check can user share folders.

Parameter	Name	Required	Default	Description
uid	User id	Y		User id
folder_id	folder_id	Y		folder_id

backend_api.check_duplicate_folder - check_duplicate_folder

This method is used to check exist folder with this name in parent folder.

Parameter	Name	Required	Default	Description
uid	User id	Y		User id
folder_id	Parent folder_id			Parent folder_id
foldername	Folder name	Y		Folder name

backend_api.check_folder_access - check_folder_access

This method is used to check folder access.

Parameter	Name	Required	Default	Description
uid	User id	Y		User id
folder_id	folder_id	Y		folder_id

backend_api.check_folder_owner - check_folder_owner

This method is used to check folder owner.

Parameter	Name	Required	Default	Description
uid	User id	Y		User id
folder_id	folder_id	Y		folder_id

backend_api.check_hash - check_hash

This method is used to check hash and ip.

Parameter	Name	Required	Default	Description
hash	hash	Y		recieved user HASH
ip	ip address	Y		ip address

backend_api.create_file - create_file

This method is used to create local file record.

Parameter	Name	Required	Default	Description
filename	filename	Y		filename
size	size	Y		File size
folder_id	folder_id	Y		folder_id
sha1folder	sha1folder			sha1folder
active	active		0	active
cdn_ready	cdn_ready		0	cdn_ready
dup	dup		1	create_duplicate

backend_api.create_folder - create_folder

This method is used to create new folder.

Parameter	Name	Required	Default	Description
uid	User id	Y		User id
folder_id	Parent folder_id			Parent folder_id
foldername	Folder name	Y		Folder name
return	need return		listing	Type of return

backend_api.create_lead - create_lead

This method is used to create lead (future user).

Parameter	Name	Required	Default	Description
email	User email	Y		User email

backend_api.default_folder - default_folder

This method is used to set default_folder for uploads.

Parameter	Name	Required	Default	Description
uid	User id	Y		User id
foldername	Folder name			Folder name

backend_api.del_tag - del_tag

This method is used to del all tags from file (file_id)

Parameter	Name	Required	Default	Description
file_id	file_id	Y		file id

backend_api.delete - delete

This method is used to delete [... file ...] from database.

Parameter	Name	Required	Default	Description
table	table	Y		mysql table name
id	record id	Y		record id to delete
fieldname	field name	Y		fied name

backend_api.delete_file - delete_file

This method is used to delete file.

Parameter	Name	Required	Default	Description
uid	User id	Y		User id
hash	hash			user hash for send answer or errors
file_id	file_id	Y		file_id
filename	file name			file name

backend_api.delete_folder - delete_folder

This method is used to delete folder.

Parameter	Name	Required	Default	Description
uid	User id	Y		User id
folder_id	folder_id	Y		folder_id
foldername	Folder name			Folder name

backend_api.delete_share - delete_share

This method is used to delete (stop) share folder

Parameter	Name	Required	Default	Description
uid	User id	Y		User id
email	User email	Y		User email
folder_id	folder_id	Y		folder_id

backend_api.documentation - Download API Documentation

This method is used to get a document with details of all available API methods.

Example Output

This method will output PDF data containing the API documentation.

backend_api.file_folder - return folder_id

This method is used to get folder id for exist file_id.

Parameter	Name	Required	Default	Description
file_id	file_id	Y		Local file_id

backend_api.files - files

This method is used to list files in selected folder.

Parameter	Name	Required	Default	Description
uid	User id	Y		User id
folder_id	folder_id			folder_id
ip	ip	Y		ip
sort	sorting		name	sorting
order	order		asc	order
offset	offset		0	offset
page	page		1	page
limit	limit		20	limit

backend_api.fill_share - fill_share

This method is used to fill files to folder.

Parameter	Name	Required	Default	Description
folder_id	folder_id	Y		folder_id
content_type	content_type	Y	movie	content_type
year	year			year
create_tags	create_tags			create_tags
source	source			source
translate_id	translate_id			translate_id
limit	limit			limit
dup	duplicate		1	create duplicate
uid	uid	Y		uid

backend_api.folder_tree - folder_tree

This method is used to get folder_tree from current folder.

Parameter	Name	Required	Default	Description
uid	User id	Y		User id
folder_id	folder_id			folder_id

backend_api.format_bytes - format_bytes

This method is used to format bytes to readable size.

Parameter	Name	Required	Default	Description
bytes	bytes	Y		bytes
round	round		2	round

backend_api.get_default_folder - get_default_folder

This method is used to get default_folder for uploads.

Parameter	Name	Required	Default	Description
uid	User id	Y		User id

backend_api.get_parent - get_parent

This method is used to get parent_id for current folder.

Parameter	Name	Required	Default	Description
folder_id	folder_id	Y		folder_id

backend_api.get_uid - get_uid

This method is used to get user id.

Parameter	Name	Required	Default	Description
email	User email	Y		User email

backend_api.get_user - get_user

This method is used to get all user info.

Parameter	Name	Required	Default	Description
uid	User id	Y		User id
params	params			params from user DB

backend_api.listing - listing

This method is used to listing folder for sub folders and files.

Parameter	Name	Required	Default	Description
uid	User id	Y		User id
hash	hash			user hash for send answer or errors
folder_id	folder_id			folder_id
ip	ip	Y		ip
sort	sorting		name	sorting
order	order		asc	order
offset	offset		0	offset
page	page		1	page
limit	limit		20	limit
send	send now		0	send listing to client

backend_api.move - move

This method is used to move folders or files.

Parameter	Name	Required	Default	Description
uid	User id	Y		User id
folder_id	folder_id			folder_id
file_id	file_id			file_id
target_folder_id	target_folder_id	Y		target_folder_id

backend_api.move_file - move_file

This method is used to move_file.

Parameter	Name	Required	Default	Description
uid	User id	Y		User id
file_id	file_id			file_id

backend_api.move_folder - move_folder

This method is used to get first act of move_folder.

Parameter	Name	Required	Default	Description
uid	User id	Y		User id
folder_id	folder_id			folder_id

backend_api.paginator - paginator

This method is used to make page navigation.

Parameter	Name	Required	Default	Description
total	total	Y	0	total
per_page	per_page	Y		per_page
current	current	Y		current
folder_id	folder_id	Y		folder_id

backend_api.plans_listing - plans_listing

This method is used to list available plans.

Parameter	Name	Required	Default	Description
active	active	Y	1	active or not active plans

backend_api.rabbit_task - rabbit_task

This method is used to create new rabbitMQ task.

Parameter	Name	Required	Default	Description
task_name	task_name	Y		task_name
task	task	Y		task params

backend_api.rabbit_worker - rabbit_worker

This method is used to call for rabbit worker.

Parameter	Name	Required	Default	Description
cmd	cmd	Y		cmd
message	message	Y		message

backend_api.recursive_down - recursive_down

This method is used to get all child folders from current.

Parameter	Name	Required	Default	Description
uid	User id	Y		User id
hash	User token			User token (hash)
folder_id	folder_id			folder_id

backend_api.recursive_folder_size - recursive_folder_size

This method is used to get count subfolders and sum of size of all files in all subfolders

Parameter	Name	Required	Default	Description
folder_id	folder_id	Y		folder_id

backend_api.recursive_up - recursive_up

This method is used to get all folders from current to up.

Parameter	Name	Required	Default	Description
uid	uid	Y		uid
hash	User token			User token (hash)
folder_id	folder_id	Y		folder_id

backend_api.refusal - refusal

This method is used to refusal from shared folder

Parameter	Name	Required	Default	Description
uid	User id	Y		User id
hash	User token			User token (hash)
folder_id	folder_id	Y		folder_id

backend_api.rename_file - rename_file

This method is used to rename file.

Parameter	Name	Required	Default	Description
uid	User id	Y		User id
hash	User token			User token (hash)
file_id	file_id	Y		file_id
filename	file name			file name

backend_api.rename_folder - rename_folder

This method is used to rename folder.

Parameter	Name	Required	Default	Description
uid	User id	Y		User id
hash	User token			User token (hash)
folder_id	folder_id	Y		folder_id
foldername	Folder name	Y		Folder name

backend_api.save - save

This method is used to save files.

Parameter	Name	Required	Default	Description
uid	User id	Y		User id
hash	User token			User token (hash)
file_id	file_id	Y		file_id
target_folder_id	target_folder_id	Y		target_folder_id

backend_api.save_file - save_file

This method is used to save_file.

Parameter	Name	Required	Default	Description
uid	User id	Y		User id
hash	User token			User token (hash)
file_id	file_id			file_id

backend_api.search - search

This method is used to search files in all user folders.

Parameter	Name	Required	Default	Description
need	need	Y		need
uid	User id	Y		User id
folder_id	folder id			folder id
hash	User token			User token (hash)
sort	sorting		name	sorting
order	order		asc	order
offset	offset		0	offset
limit	limit		20	limit

backend_api.send - send message

This method is used to send message to user via sockets.

Parameter	Name	Required	Default	Description
uid	User id			User id, send to all device of user
hash	User hash			User hash, send only to this hash
message	message	Y		Message for send
type	type of message		info	Type of message

backend_api.share_folder - share_folder

This method is used to share (reshare) folder

Parameter	Name	Required	Default	Description
uid	User id	Y		User id
hash	User token			User token (hash)
email	User email	Y		User email
folder_id	folder_id	Y		folder_id
reshare	reshare		0	allow re-sharing

backend_api.shares - shares

This method is used to list shares.

Parameter	Name	Required	Default	Description
uid	User id	Y		User id

backend_api.sign_links - sign_links

This method is used for sign cdn files.

Parameter	Name	Required	Default	Description
url	url	Y		url for sign
ip	ip			ip
tag	tag			custom tag
sign_type	sign_type		notuse	string sign_type (ip or cookie or notuse)
cdn_key	cdn_key		iUZrolmDExUi	cdn_key
time	time		6	time in hours

backend_api.stop_sharing - share_folder

This method is used to share (reshare) folder

Parameter	Name	Required	Default	Description
uid	User id	Y		User id
email	User email	Y		User email
folder_id	folder_id	Y		folder_id
reshare	reshare		0	allow re-sharing

backend_api.subfolders - subfolders

This method is used to listing folder subfolders.

Parameter	Name	Required	Default	Description
uid	User id	Y		User id
folder_id	folder_id		0	folder_id
exclude_folder_id	exclude_folder_id			exclude folder_id
sort	sorting		name	sorting
order	order		asc	order
offset	offset		0	offset
limit	limit		50	limit

backend_api.update_file - update_file

This method is used to update local file record.

Parameter	Name	Required	Default	Description
file_id	file_id	Y		File ID
filename	filename			filename
type	type			File type (ext)
size	size			File size
sha1	sha1			File sha1
created	created			File created date
folder_id	folder_id			folder_id
cdn_folder_id	cdn_folder_id			CDN folder_id
cdn_file_id	cdn_file_id			CDN file_id
cdn_path	cdn_path			CDN path
local_path	local_path			Local path
active	active			active
cdn_ready	cdn_ready			cdn_ready

backend_api.update_folder - update_folder

This method is used to update local folder record.

Parameter	Name	Required	Default	Description
folder_id	folder_id	Y		Folder ID
foldername	foldername			foldername
parent_id	parent_id			parent_id
active	active			active

backend_api.update_user - update_user

This method is used to update user.

Parameter	Name	Required	Default	Description
uid	User id	Y		User id
params	params	Y		params to update

backend_api.used_storage - used_storage

This method is used to get how many used user storage.

Parameter	Name	Required	Default	Description
uid	User id	Y		User id
hash	hash			user hash for send answer or errors
send	send now		0	send data to client

backend_api.user_tokens - user_tokens

This method is used to return all active tokens for uid, if present ip, receive only token for this uid and this ip.

Parameter	Name	Required	Default	Description
uid	User id	Y		User id
ip	ip address			user ip address