

Socneto Api

Overview

Socneto task api - no users yet

Version information

Version : v1

Paths

GET /api/charts/{jobId}

Parameters

Type	Name	Schema
Path	jobId <i>required</i>	string (uuid)

Responses

HTTP Code	Description	Schema
200	Success	< ChartDefinitionDt o > array

Produces

- `text/plain`
- `application/json`
- `text/json`

Tags

- Charts

POST /api/charts/{jobId}/create

Parameters

Type	Name	Schema
Path	jobId <i>required</i>	string (uuid)
Body	request <i>optional</i>	CreateChartDefinitionRequest

Responses

HTTP Code	Description	Schema
200	Success	ChartDefinitionDto

Consumes

- [application/json-patch+json](#)
- [application/json](#)
- [text/json](#)
- [application/*+json](#)

Produces

- [text/plain](#)
- [application/json](#)
- [text/json](#)

Tags

- Charts

GET /api/charts/{jobId}/{chartId}/remove

Parameters

Type	Name	Schema
Path	chartId <i>required</i>	string (uuid)
Path	jobId <i>required</i>	string (uuid)

Responses

HTTP Code	Description	Schema
200	Success	< ChartDefinitionDto > array

Produces

- [text/plain](#)
- [application/json](#)
- [text/json](#)

Tags

- Charts

GET /api/components/acquirers

Responses

HTTP Code	Description	Schema
200	Success	< AcquirerDto > array

Produces

- [text/plain](#)
- [application/json](#)
- [text/json](#)

Tags

- Components

GET /api/components/analysers

Responses

HTTP Code	Description	Schema
200	Success	< AnalyserDto > array

Produces

- [text/plain](#)
- [application/json](#)
- [text/json](#)

Tags

- Components

GET /api/heart-beat

Responses

HTTP Code	Description	Schema
200	Success	string

Produces

- [text/plain](#)
- [application/json](#)
- [text/json](#)

Tags

- Management

GET /api/job/all

Responses

HTTP Code	Description	Schema
200	Success	< JobDto > array

Produces

- `text/plain`
- `application/json`
- `text/json`

Tags

- Job

POST /api/job/create

Parameters

Type	Name	Schema
Body	request <i>optional</i>	JobSubmitRequest

Responses

HTTP Code	Description	Schema
200	Success	JobStatusResponse

Consumes

- `application/json-patch+json`
- `application/json`
- `text/json`
- `application/*+json`

Produces

- `text/plain`
- `application/json`
- `text/json`

Tags

- Job

POST /api/job/{jobId}/aggregation_analysis

Parameters

Type	Name	Schema
Path	jobId <i>required</i>	string (uuid)
Body	analysisRequest <i>optional</i>	GetAggregationAnalysisRequest

Responses

HTTP Code	Description	Schema
200	Success	AggregationAnalysisResponse

Consumes

- `application/json-patch+json`
- `application/json`
- `text/json`
- `application/*+json`

Produces

- `text/plain`
- `application/json`
- `text/json`

Tags

- Job

POST /api/job/{jobId}/array_analysis

Parameters

Type	Name	Schema
Path	jobId <i>required</i>	string (uuid)
Body	analysisRequest <i>optional</i>	GetArrayAnalysisRequest

Responses

HTTP Code	Description	Schema
200	Success	ArrayAnalysisResponse

Consumes

- [application/json-patch+json](#)
- [application/json](#)
- [text/json](#)
- [application/*+json](#)

Produces

- [text/plain](#)
- [application/json](#)
- [text/json](#)

Tags

- Job

GET /api/job/{jobId}/author_frequency

Parameters

Type	Name	Schema
Path	jobId <i>required</i>	string (uuid)

Responses

HTTP Code	Description	Schema
200	Success	AggregationAnalysisResponse

Produces

- [text/plain](#)
- [application/json](#)

- `text/json`

Tags

- Job

GET /api/job/{jobId}/language_frequency

Parameters

Type	Name	Schema
Path	jobId <i>required</i>	string (uuid)

Responses

HTTP Code	Description	Schema
200	Success	AggregationAnalysisResponse

Produces

- `text/plain`
- `application/json`
- `text/json`

Tags

- Job

GET /api/job/{jobId}/posts

Parameters

Type	Name	Schema	Default
Path	jobId <i>required</i>	string (uuid)	
Query	contains_words <i>optional</i>	< string > array(multi)	

Type	Name	Schema	Default
Query	exclude_words <i>optional</i>	< string > array(multi)	
Query	from <i>optional</i>	string (date-time)	
Query	page <i>optional</i>	integer (int32)	1
Query	page_size <i>optional</i>	integer (int32)	20
Query	to <i>optional</i>	string (date-time)	

Responses

HTTP Code	Description	Schema
200	Success	Paginated[AnalyzedPostDto]

Produces

- `text/plain`
- `application/json`
- `text/json`

Tags

- Job

GET /api/job/{jobId}/posts/export

Parameters

Type	Name	Schema
Path	jobId <i>required</i>	string (uuid)
Query	contains_words <i>optional</i>	< string > array(multi)

Type	Name	Schema
Query	exclude_words <i>optional</i>	< string > array(multi)
Query	from <i>optional</i>	string (date-time)
Query	to <i>optional</i>	string (date-time)

Responses

HTTP Code	Description	Schema
200	Success	No Content

Produces

- `text/csv`

Tags

- Job

GET /api/job/{jobId}/posts_frequency

Parameters

Type	Name	Schema
Path	jobId <i>required</i>	string (uuid)

Responses

HTTP Code	Description	Schema
200	Success	AggregationAnalysisResponse

Produces

- `text/plain`

- `application/json`
- `text/json`

Tags

- Job

GET /api/job/{jobId}/status

Parameters

Type	Name	Schema
Path	jobId <i>required</i>	string (uuid)

Responses

HTTP Code	Description	Schema
200	Success	JobDto

Produces

- `text/plain`
- `application/json`
- `text/json`

Tags

- Job

GET /api/job/{jobId}/stop

Parameters

Type	Name	Schema
Path	jobId <i>required</i>	string (uuid)

Responses

HTTP Code	Description	Schema
200	Success	JobStatusResponse

Produces

- [text/plain](#)
- [application/json](#)
- [text/json](#)

Tags

- Job

GET /api/platform_status

Responses

HTTP Code	Description	Schema
200	Success	SocnetoComponentsStatus

Produces

- [text/plain](#)
- [application/json](#)
- [text/json](#)

Tags

- Management

POST /api/produce

Parameters

Type	Name	Schema
Body	request <i>optional</i>	ProduceRequest

Responses

HTTP Code	Description	Schema
200	Success	No Content

Consumes

- `application/json-patch+json`
- `application/json`
- `text/json`
- `application/*+json`

Tags

- Management

POST /api/user/login

Parameters

Type	Name	Schema
Body	login <i>optional</i>	LoginRequest

Responses

HTTP Code	Description	Schema
200	Success	LoginResponse

Consumes

- `application/json-patch+json`
- `application/json`
- `text/json`
- `application/*+json`

Produces

- `text/plain`
- `application/json`
- `text/json`

Tags

- User

Definitions

AcquirerDto

Name	Schema
component_type <i>optional</i>	enum (DATA_ANALYSER, DATA_ACQUIRER)
identifier <i>optional</i>	string

AggregationAnalysisResponse

Name	Schema
aggregations <i>optional</i>	< string, object > map

AnalyserDto

Name	Schema
analysis_properties <i>optional</i>	< string, enum (numberValue, textValue, numberListValue, textListValue, numberMapValue, textMapValue) > map
component_type <i>optional</i>	enum (DATA_ANALYSER, DATA_ACQUIRER)
identifier <i>optional</i>	string

AnalysisDataPath

Name	Schema
analyser_id <i>optional</i>	string

Name	Schema
property <i>optional</i>	string

AnalysisDataPathRequest

Name	Schema
analyser_component_id <i>optional</i>	string
analyser_property <i>optional</i>	string

AnalyzedPostDto

Name	Schema
analyses <i>optional</i>	object
job_id <i>optional</i>	string (uuid)
post <i>optional</i>	PostDto

ArrayAnalysisResponse

Name	Schema
data <i>optional</i>	< object > array
total_count <i>optional</i>	integer (int32)

ChartDefinitionDto

Name	Schema
analysis_data_paths <i>optional</i>	< AnalysisDataPath > array

Name	Schema
chart_type <i>optional</i>	string
id <i>optional</i>	string (uuid)
is_x_datetime <i>optional</i>	boolean
title <i>optional</i>	string

CreateChartDefinitionRequest

Name	Schema
analysis_data_paths <i>optional</i>	< AnalysisDataPathRequest > array
chart_type <i>optional</i>	integer (int32)
is_x_post_datetime <i>optional</i>	boolean
title <i>optional</i>	string

GetAggregationAnalysisRequest

Name	Schema
analyser_id <i>optional</i>	string
analysis_property <i>optional</i>	string

GetArrayAnalysisRequest

Name	Schema
analyser_id <i>optional</i>	string
analysis_properties <i>optional</i>	< string > array
is_x_post_date <i>optional</i>	boolean
page <i>optional</i>	integer (int32)
page_size <i>optional</i>	integer (int32)

JobDto

Name	Schema
finished_at <i>optional</i>	string (date-time)
job_id <i>optional</i>	string (uuid)
job_name <i>optional</i>	string
started_at <i>optional</i>	string (date-time)
status <i>optional</i>	enum (Running, Stopped)
topic_query <i>optional</i>	string

JobStatusResponse

Name	Schema
job_id <i>optional</i>	string (uuid)
status <i>optional</i>	enum (Running, Stopped)

JobSubmitRequest

Name	Schema
attributes <i>optional</i>	< string, < string, string > map > map
job_name <i>optional</i>	string
selected_acquirers <i>optional</i>	< string > array
selected_analysers <i>optional</i>	< string > array
topic_query <i>optional</i>	string

LoginRequest

Name	Schema
password <i>optional</i>	string
username <i>optional</i>	string

LoginResponse

Name	Schema
username <i>optional</i>	string

Paginated[AnalyzedPostDto]

Name	Schema
data <i>optional</i>	< AnalyzedPostDto > array
pagination <i>optional</i>	Pagination

Pagination

Name	Schema
page <i>optional</i>	integer (int32)
page_size <i>optional</i>	integer (int32)
total_size <i>optional</i>	integer (int32)

PostDto

Name	Schema
author_id <i>optional</i>	string
language <i>optional</i>	string
original_id <i>optional</i>	string
original_text <i>optional</i>	string
posted_at <i>optional</i>	string (date-time)
text <i>optional</i>	string

ProduceRequest

Name	Schema
kafkaTopic <i>optional</i>	string
message <i>optional</i>	string
serverAddress <i>optional</i>	string

SocnetoComponentsStatus

Name	Schema
acquirers <i>optional</i>	< string, enum (STOPPED, STARTING, RUNNING) > map
analysers <i>optional</i>	< string, enum (STOPPED, STARTING, RUNNING) > map
jms <i>optional</i>	enum (STOPPED, STARTING, RUNNING)
storage <i>optional</i>	enum (STOPPED, STARTING, RUNNING)