D2.3
Platform architecture and specification final

Document type	Report
Deliverable Nº	D2.3
Work package	WP2
Leading partner	ONE
Author(s)	Luis Cordeiro, Bruno Sousa
Dissemination level	Private
Status	Final
Date	30 December 2016
Version	1.0

Project	Enabling Multichannel PArticipation Through ICT Adaptations
Acronym	EMPATIA
Grant nº	687920
Duration	01/01/2016 to 30/12/2017

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 687920

List of Authors and Reviewers

Authors	Partner
Luis Cordeiro	ONE
Vitor Fonseca	ONE
Bruno Sousa	ONE
Pedro Valente	ONE

Reviewers	Partner

Versioning and contribution history

#	Description	Contributors
0.1	Table of contents	Luis Cordeiro
0.2	First content version	Bruno Sousa
0.3	General contributions	Vitor Fonseca
0.4	General contributions	Pedro Valente
0.5	2nd cycle contributions	Giulia Bertone
0.6	General contributions	Bruno Sousa
1.0	Final version	Luis Cordeiro

Executive Summary

This document presents the final version of the EMPATIA Platform architecture, initially specified in Deliverable D2.1[1]. The document starts by updating the initial state of the art (SoA) analysis, identifying a complete set of existing tools related with participation. The EMPATIA Platform technical requirements were updated to reflect the revised SoA analysis, EMPATIA use cases and scientific requirements, reported in Deliverable D1.1 [2] and on the developments activities of the EMPATIA Platform, reported in Deliverable D2.2 [3]. For each of the identified requirement, a priority is set based on EMPATIA pilots and use cases' needs.

A second section is devoted to the EMPATIA Platform architecture, where all the components and respective APIs are documented in detail. Emphasis on the major changes from the preliminary specification are also provided, changes which were performed to accommodate new functionalities or even to enhance the performance of the platform.

The lessons learn with the first implementations of the EMPATIA Platform are presented, considering the pilots' integration and developments activities conducted in some pilots.

In summary, the specification, implementation and testing activities have been executed according to the defined plans (and earlier in most cases), having implemented and tested most of the functionalities of the first EMPATIA Platform version released at M10.

Table of Contents

Table of Figures

Table of Tables

Acronyms

Acronyms	Description
API	Application Programming Interface
CB	Community Building
CM	Content Management
CRUD	Create Read Update Delete
CSV	Comma Separated Values
D-CENT	Decentralized Citizens Engagement Technologies
DCN	Deliberative Community Networks
DigiWhist	Digital Whistle-blower
EU	European Union
JSON	JavaScript Object Notation
JWT	JSON Web Token
IT	Information Technology
KPI	Key Performance Indicator
MP	Multichannel Participation
N/A	Not Available
PB	Participatory Budget
Q	Questionnaire
REST	Representational State Transfer
SAS	Statistical Analysis System
SMS	Short Message Service
SPSS	Statistical Package for the Social Sciences, IBM SPSS Statistics
SQL	Structured Query Language
WUI	Web User Interface
SoA	State of the Art
SaaS	Software as a Service
EMPATIAaaS	EMPATIA as a Service

1 Introduction

This document presents the final version of the EMPATIA Platform architecture, detailing its components and the respective APIs. The architecture and each component have been designed to be scalable, redundant to support fault tolerance and different deployment models. In addition, the architecture specification derives from the functional and non-functional requirements that have updated the initial requirements documented in Deliverable D2.1 [1] due to the development and integration activities, as documented in Deliverable D2.2. Such requirements are derived from the five EMPATIA use cases reported in Deliverable D1.1 [2]. Key aspects considered for the components and EMPATIA architecture include the support of different deployment models: all-in-one deployment (e.g. deployment in a single server); cloud deployment (e.g. public clouds like Amazon web services, Windows Azure, Openstack, others); and virtualization based on VMWare, Xen Citrix and docker containers. The concept a Software as a Service (SaaS), called EMPATIA as a Service (EMPATIAaaS), was also considered important for commercial exploitation. The deployment and usage of EMPATIA Platform should not require deep technical expertise from an end-user, no matter the selected deployment model.

The technological options of the preliminary architecture have been validated in the development and integration of the preliminary EMPATIA architecture. The achieved results validated the initial key design goals of the architecture namely in terms of scalability and fault tolerance support as well as the flexibility in the deployment models. Thus, some components have been merged into new components to accommodate new functionalities, such as the Community Building that in the final architecture corresponds to the PAD component. The APIs of the components have been enhanced as well, to allow the release of the platform as open source and to support the release of data in Open Data repositories, such as CKAN [4].

The SCRUM agile methodologies allowed a continuous delivery of the platform with enough flexibility to introduce new functionalities. The final implementation plans are delineated in this deliverable towards the final prototype of EMPATIA.

2 State of the art

This section describes the state of the art (SoA) of existing participation platforms (including participatory processes). This study focused in analysing existing solutions to identify possible platforms/tools/features to integrate within EMPATIA, identify requirements and identify limitations and issues to avoid.

The analysis of the state of the art has been performed according to the following methodology:

- Analysis of tools, platforms according to the requirements identified in the EMPATIA use cases (documented in D1.1) and identified in D2.1;
- Analysis of monitoring capabilities of the platforms, in terms of supporting static, dynamic and qualitative monitoring of the evolution of projects, including the possibility to check if progress is according to established deadlines, respecting the budget planning, among other aspects;
- Identification and deep analysis of most relevant PB platforms, in terms of functional, security and open access aspects.

2.1 Tools, platforms and projects

This section analyses diverse tools, platforms and projects that in some aspect foster participation. More concretely the deep analysis aimed to achieve the following goals:

1. Extensively identify the functional aspects of platforms, including supported deployment models, the required technical skills to use and deploy the platform;
2. The standards and license models associated, which can introduce limitations in participatory budget schemes;
3. The privacy and data protection mechanisms to assure that all the EU legislation regarding privacy and protection of data is accomplished;
4. Support for open access data, assuring the interoperability with other platforms and respecting ethical principles.

The template for the analysis of the state of the art regarding tools and platforms is summarized in Table 1. All the projects are reviewed according to this template, nonetheless, the most relevant projects have further details due to a rigorous and complete analysis. The deep analysis of the most relevant projects included the following projects: Your priorities [5], Loomio [7], the CitizenBudget [9], the LiberOpinion [66], the Paris Budget Participatif [10], the consider.it [67] the participare [13] and the consul [16] projects.

Table 1 – Template for Platform analysis

Functional aspects	Description
Deployment model	SaaS – Software as a Service All-in-one – (Local install, cloud install, virtualization) Single/Chosen components
Technical skills required for deployment/configuration for U-user, A-admin and M-manager spaces.	Basic Web knowledge Supports customization

Supported technologies/frameworks	Which frameworks are supported
Supported OS	All, Linux, Windows, MacOS, BSD
Main functionalities	List of functionalities in comparison to the EMPATIA components
Standards & License	Description
License	Type of license included (BSD, Apache2)
Available License	Where License is available
Third Party software	Uses 3rd party software, such as Google analytics
Privacy & Data Protection	Description
Information Sheets regarding collected and mined data	Does the platform/tool has a written policy regarding privacy and data protection
Information Consent Request	Is there a written information consent request? The collected metadata and standards are presented to the user
Privacy & personal data protection security	Description
Authentication	Supported authentication method
Encryption	Supported encryption methods
Password recovery	Allows user to recover its password
OAuth	Supports OAuth and how
Open Access	Description
Export of Data	Supports exporting data
Social media interaction	Interacts with social sites such as Twitter, Facebook, Google Plus
API availability	Has available an API that can be used by other tools, platforms?
Ethical principles	Addresses ethical issues

Next is presented the analyses to each of the tools, platforms and projects. Each presented analysis has a specific goal based on the characteristics of the analysed tools, platforms and projects. Because some tools, platforms and projects don’t have, or is not available data/features, some of the analysis have incomplete information.

2.1.1 Your Priorities

Your Priorities [5] is an application for the participation of citizens, developed by the non-profit Citizens Foundation of Iceland [6]. The analysis of this platform is summarized in Table 2.

Table 2 – Your Priorities Platform analysis

Functional aspects	Description
Deployment model	SaaS and All-in-One
Technical skills required for deployment/configuration	U-A-M: basic web knowledge
Supported technologies/frameworks	Cowbow framework, Polymer framework, google analytics, Erlang, Node.js
Supported OS	All, where Node.js is supported

List of functionalities	Authentication Community Building Content Management Events Notify Vote
Standards & License	Description
License	BSD
Available License	https://github.com/rbjarnason/your-priorities-app/blob/master/LICENSE.md
Third Party software	Google Analytics
Privacy & Data Protection	Description
Information Sheets regarding collected and mined data	No
Information Consent Request	No
Privacy & personal data protection security	Description
Authentication	Double-step authentication
Encryption	Supports HTTPS
Password recovery	Yes, by email
OAuth	Yes
Open Access	Description
Export of Data	No information is provided
Social media interaction	Yes, Facebook
API availability	No
Ethical principles	Yes

2.1.2 Loomio

Loomio [7] is presented as an application to allow groups to make decisions together. The platform has been used in New Zealand, Wales, among other places, to engage the participation of users in the process of improving the municipality where they live. The analysis of this platform is summarized in Table 3.

Table 3 – Loomio Platform analysis

Functional aspects	
Deployment model	SaaS and All-in-One
Technical skills required for deployment/configuration	U-A-M: basic web knowledge
Supported technologies/frameworks	CloudFlare (as CDN), AngulasJS, Nginx, Intercom, Moment.js, New Relic, jQuery
Supported OS	All, where Ngnix is supported
List of functionalities	Analytics Authentication Community Building Content Management Events

	Notify Questionnaire Vote
Standards & License	Description
License	Open Source GNU Affero General Public Licens (AGPL)
Available License	https://github.com/loomio/loomio/blob/master/LICENSE.txt
Third Party software	New Recli, Intercom, Google TAG Manager, Gravatar
Privacy & Data Protection	Description
Information Sheets regarding collected and mined data	Yes, in the Loomio Manual [8]
Information Consent Request	No
Privacy & personal data protection security	Description
Authentication	Double-step authentication
Encryption	Supports HTTPS
Password recovery	Yes, by email
OAuth	Yes
Open Access	Description
Export of Data	Not in the free version, only in paid models
Social media interaction	Yes
API availability	Yes
Ethical principles	Yes, more information available at: https://www.loomio.org/about

2.1.3 CitizenBugdet

CitizenBudget [9] is a tool to build better community relationships and to educate users regarding their budget consultation. The analysis of this platform is summarized in Table 4.

Table 4 – CitizenBugdet Platform analysis

Functional aspects	
Deployment model	SaaS
Technical skills required for deployment/configuration	U: basic web knowledge
Supported technologies/frameworks	D3, Google Analytics, Ngnix, RackCache, Ruby on Rails, jQuery, jQuery UI, Mustache Framework
Supported OS	All, where Ngnix is supported
List of functionalities	Analytics Authentication Community Building Content Management Events Notify Questionnaire Vote 2ND Cycle (partly for budget control on http://victoria.citizenbudget.com/

Standards & License	Description
License	Closed Source
Available License	Not published
Third Party software	Google Analytics, Facebook Connect
Privacy & Data Protection	Description
Information Sheets regarding collected and mined data	No
Information Consent Request	No
Privacy & personal data protection security	Description
Authentication	Single-step authentication
Encryption	No support for HTTPS
Password recovery	N/A
OAuth	N/A
Open Access	Description
Export of Data	N/A
Social media interaction	Yes, Facebook and Twitter
API availability	No
Ethical principles	Partial, within the explanation of PB processes reminded in this page http://www.opennorth.ca/2015/10/13/open-norths-citizen-budget-simulator-boosts-pioneering-citizen-engagement-project.html of the mother company website http://www.participatorybudgeting.org/about-participatory-budgeting/where-has-it-worked/

2.1.4 LiberOpinion

LiberOpinion [66] is a closed source and proprietary platform developed by Libertrium for the communication between municipalities and citizens supporting, among other features, participatory budget. It is currently being used by many municipalities in Portugal. A summary of the analysis performed to the platform is presented in Table 5.

Table 5 – LiberOpinion Platform analysis

Functional aspects	
Deployment model	SaaS
Technical skills required for deployment/configuration	U: basic web knowledge
Supported technologies/frameworks	D3, Google Maps, Ngnix, Facebook Widget, Twitter Bootstrap, reCAPTCHA, AngularJS, CKEditor, Modernizr, NVD3, jQuery, jQuery UI, Socket.io, Node.js
Supported OS	All, where Ngnix is supported
List of functionalities	Analytics Authentication Community Building Content Management Events Notify

	Questionnaire Vote 2ND Cycle (partly, external 3rd party) http://lisboaparticipa.pt/projetos-vencedores Issues reporting Transparent Execution of projects: Users can receive notifications of project progress Users can follow a project Project evolution updated with documents and images Monitoring of winning projects of the PB process: Static Monitoring including title, proposal description, GPS location, date of submission, theme, images Dynamic Monitoring including implementation status, implementation by municipality Qualitative Monitoring including events/milestones, meetings of groups of stakeholders involved in the proposal Data visualization: maps of proposals Infographics of aggregated project data (themes, progress status, budget)
Standards & License	Description
License	Closed Source
Available License	Not published
Third Party software	Google Analytics, Facebook Connect
Privacy & Data Protection	Description
Information Sheets regarding collected and mined data	No
Information Consent Request	No
Privacy & personal data protection security	Description
Authentication	Single-step authentication
Encryption	No support for HTTPS
Password recovery	Yes, by email.
OAuth	N/A
Open Access	Description
Export of Data	N/A
Social media interaction	Yes, Facebook, Twitter, Vimeo and Google Plus
API availability	No
Ethical principles	Partial, within the explanation of PB processes http://lisboaparticipa.pt/o-que-e-o-op

2.1.5 Paris Budget Participatif

Paris Budget Participatif [10] is a platform to enable participatory budget in the municipality of Paris. The main language of the platform is French. The analysis of this platform is summarized in Table 6.

Table 6 – Paris Budget Participatif Platform analysis

Functional aspects	
Deployment model	SaaS

Technical skills required for deployment/configuration	U-A-M: basic web knowledge
Supported technologies/frameworks	Font Awesome, Java, Lightbox, Ngnix, Piwik, Twitter Bootstrap, jQuery, jQuery UI
Supported OS	All, where Ngnix is supported
List of functionalities	Analytics using Piwik Authentication Community Building Content Management Events Notify Questionnaire Vote 2ND Cycle [11]
Standards & License	Description
License	Closed Source
Available License	Not published
Third Party software	Piwik Analytics
Privacy & Data Protection	Description
Information Sheets regarding collected and mined data	Yes
Information Consent Request	No
Privacy & personal data protection security	Description
Authentication	Double-step authentication
Encryption	Support for HTTPS
Password recovery	Yes, by email.
OAuth	No
Open Access	Description
Export of Data	N/A
Social media interaction	No
API availability	No
Ethical principles	Yes, https://budgetparticipatif.paris.fr/bp/le-budget-participatif-.html

2.1.6 Consider.it

Consider.it [67] is a proprietary and closed source platform to collect feedback, make a choice, survey a group, focus a dialogue, organize a meeting among other features. The analysis of this platform is summarized in Table 7.

Table 7 – Consider.it Platform analysis

Functional aspects	
Deployment model	SaaS
Technical skills required for deployment/configuration	U-A-M: basic web knowledge

Supported technologies/frameworks	Font Awesome, Ngnix, Phusion Passenger, React, Ruby on Rails, D3, HeadJS, Modernizr
Supported OS	All, where Ngnix is supported
List of functionalities	Analytics with Google Analytics (fro paid versions) Authentication Community Building Content Management Events Notify Questionnaire Vote
Standards & License	Description
License	Closed Source
Available License	Not published
Third Party software	Google Analytics
Privacy & Data Protection	Description
Information Sheets regarding collected and mined data	No
Information Consent Request	No
Privacy & personal data protection security	Description
Authentication	Double-step authentication
Encryption	Support HTTPS
Password recovery	Yes, by email
OAuth	N/A
Open Access	Description
Export of Data	No (only in paid versions)
Social media interaction	No
API availability	No
Ethical principles	Partial, within the explanation of PB processes http://lisboaparticipa.pt/o-que-e-o-op

2.1.7 Participare

Participare [13] is a proprietary and closed source platform for participatory budget. Supporting the most common tasks, such as proposal creation and voting, Participare is a cloud based platform available in pay-per-use without the need to self-hosting it. They offer four plans, one free for small groups and three payed with different limits for the number of users and budgets, providing also online support. A summary of the analysis perform is presented Table 8.

Table 8 – Participare Platform analysis

Functional aspects	
Deployment model	SaaS
Technical skills required for deployment/configuration	U-A-M: basic web knowledge

Supported technologies/frameworks	AngularJS, jQuery, D3, Laravel, PHP, Twitter Bootstrap, other JavaScript frameworks
Supported OS	All, where Ngnix is supported
List of functionalities	Anti-Fraud Mechanisms Authentication Community Building Content Management Events Notify Questionnaire Vote Multilingual support
Standards & License	Description
License	Closed Source
Available License	Published but not easily accessible
Third Party software	Yes, https://uservoice.com
Privacy & Data Protection	Description
Information Sheets regarding collected and mined data	Yes, stated in Terms of Service (TOS)
Information Consent Request	Yes, in registration
Privacy & personal data protection security	Description
Authentication	Double-step authentication
Encryption	Support HTTPS
Password recovery	Yes, by email
OAuth	No
Open Access	Description
Export of Data	Under work [14]
Social media interaction	No
API availability	With limited functionalities (e.g. budgets, proposals, languages) [15]
Ethical principles	No

2.1.8 Consul

Consul [16] is a proprietary and closed source platform for participatory budget. The analysis of this platform is summarized in Table 9.

Table 9 – Consul Platform analysis

Functional aspects	
Deployment model	All-in-One
Technical skills required for deployment/configuration	U-A: basic web knowledge M: requires technical skills for server deployment
Supported technologies/frameworks	Ruby on Rails, Git, PostgreSQL, PhantomJS, Zurb Foundation Web Framework
Supported OS	Linux, Unix-like, BSD, Apache

List of functionalities	Authentication Community Building Content Management Events Notify Questionnaire Vote
Standards & License	Description
License	AFFERO GPL v3
Available License	https://github.com/consul/consul/blob/master/LICENSE-AGPLv3.txt
Third Party software	Yes, Front Page
Privacy & Data Protection	Description
Information Sheets regarding collected and mined data	Yes
Information Consent Request	Yes
Privacy & personal data protection security	Description
Authentication	Double-step authentication
Encryption	Support HTTPS
Password recovery	Yes, by email
OAuth	Yes, Twitter, Facebook and Google
Open Access	Description
Export of Data	Some metadata can be exported
Social media interaction	Not provided, but can be implemented using Oauth.
API availability	N/A
Ethical principles	Yes, published in [17]

2.1.9 AppCivist

AppCivist [22], [23] is a platform for democratic assembly and collective action that has been developed by Inria and the Social Apps Laboratory, available under an Apache 2 license. The platform is based on modular components, which can be selected by the user of the platform. The components that compose the platform are: Proposal making, Visualization, Drawing, Mapping, Denouncing, Alerts, Voting, Deliberation, Versioning, Forum making, Monitoring and Mobilizing. The analysis of this platform is summarized in Table 10.

Table 10 – AppCivist Platform analysis

Functional aspects	
Deployment model	SaaS and All-in-One
Technical skills required for deployment/configuration	U-A-M: basic web knowledge
Supported technologies/frameworks	N/A
Supported OS	N/A

List of functionalities	Authentication Community Building: Collaboration between citizens creating proposals Content Management Events Notify Data Visualization: Custom fields on graphics and reports Support for maps
Standards & License	Description
License	N/A
Available License	N/A
Third Party software	N/A
Privacy & Data Protection	Description
Information Sheets regarding collected and mined data	N/A
Information Consent Request	N/A
Privacy & personal data protection security	Description
Authentication	N/A
Encryption	N/A
Password recovery	N/A
OAuth	N/A
Open Access	Description
Export of Data	No information is provided
Social media interaction	N/A
API availability	N/A
Ethical principles	N/A

2.1.10 D-CENT

The Decentralized Citizens Engagement Technologies (D-CENT) is a Europe-wide project aiming to develop open source, distributed and privacy-aware tools for direct democracy and economic empowerment. The developed tools allow the creation of proposals/ideas, with or without collaboration, and voting on the created proposals. The D-CENT platform relies on a set of tools (Your Priorities, OpenAhjo API [25], Notifications), leading to some complexity regarding their integration to support a participatory budget project. The different tools that compose the platform are available under an AFFERO GPL v3 license. A summary of the platform analysis is presented in Table 11.

Table 11 – D-CENT Platform analysis

Functional aspects	
Deployment model	N/A
Technical skills required for deployment/configuration	N/A

Supported technologies/frameworks	N/A
Supported OS	N/A
List of functionalities	Authentication Community Building Content Management Events Notify (email) Vote Social Media (Twitter, Facebook) Multi-language support Open data – all the details about the city council are available to citizens User account verification by ID card and residence
Standards & License	Description
License	AFFEROGPL v3
Available License	N/A
Third Party software	N/A
Privacy & Data Protection	Description
Information Sheets regarding collected and mined data	N/A
Information Consent Request	N/A
Privacy & personal data protection security	Description
Authentication	N/A
Encryption	N/A
Password recovery	N/A
OAuth	N/A
Open Access	Description
Export of Data	N/A
Social media interaction	N/A
API availability	N/A
Ethical principles	N/A

2.1.11 Bilancio Partecipativo -- Bipart

The Bilancio Partecipativo – BiPart [26][26] is a project for Participatory Democracy from the Centro Studi Democrazia Partecipativa. The platform is based on OpenDCN platform (described in section 2.1.12) with specific adaptations and configurations. This platform allows to manage and monitor different steps of a participatory process, aiming at reducing the costs to the municipalities or entities using it. The summarized analysis of the platform is shown in Table 12.

Table 12 – Bipart Platform analysis

Functional aspects	
Deployment model	N/A

Technical skills required for deployment/configuration	U: basic web knowledge
Supported technologies/frameworks	N/A
Supported OS	N/A
List of functionalities	Authentication Community Building Content Management Events Notify Questionnaire Vote User Account Verification by: Social security number Identity card By SMS Monitoring of winning projects of the PB process: Static Monitoring including title, proposal description, GPS location, date of submission, theme, images Dynamic Monitoring including implementation status, implementation progress and implementation by municipality Qualitative Monitoring including milestones and implementation by municipality Data visualization: Aggregated table with projects
Standards & License	Description
License	N/A
Available License	N/A
Third Party software	N/A
Privacy & Data Protection	Description
Information Sheets regarding collected and mined data	N/A
Information Consent Request	N/A
Privacy & personal data protection security	Description
Authentication	N/A
Encryption	N/A
Password recovery	N/A
OAuth	N/A
Open Access	Description
Export of Data	N/A
Social media interaction	N/A
API availability	N/A
Ethical principles	N/A

2.1.12 OpenDCN

OpenDCN [27], is an open source software environment composed of many different tools, such as Informed Discussion and Brainstorming, aimed at supporting participative and deliberative processes.

Developed by the Foundation RCM [28] and the Civic Informatics Laboratory [29]. This set of tools is aimed at both municipalities and private organizations. The development platform was released under a MIT license while the application is available under a GPL license. The analysis performed to the platform is summarized in Table 13.

Table 13 – openDCN Platform analysis

Functional aspects	
Deployment model	N/A
Technical skills required for deployment/configuration	U-A-M: basic web knowledge
Supported technologies/frameworks	N/A
Supported OS	N/A
List of functionalities	Authentication Community Building Content Management Events Notify Certified Citizen Consultation On-Line Ordered Meeting Informed Discussion Vote Negative Positive
Standards & License	Description
License	MIT
Available License	http://opendcn.org/index.php/en/licenza-opendcn
Third Party software	N/A
Privacy & Data Protection	Description
Information Sheets regarding collected and mined data	N/A
Information Consent Request	N/A
Privacy & personal data protection security	Description
Authentication	N/A
Encryption	N/A
Password recovery	N/A
OAuth	N/A
Open Access	Description
Export of Data	N/A
Social media interaction	N/A
API availability	N/A
Ethical principles	N/A

2.1.13 Parelon

Parelon [30] is an international non-profit organization developing a platform with the same name aimed at involving the citizens in decisions relating with public affairs. The platform under development is based on the already existent open source LiquidFeedback [31]. Although it is still being developed, it is possible to test part of the platform. A summary of the analysis performed is shown in Table 14.

Table 14 – Parelon analysis

Functional aspects	
Deployment model	N/A
Technical skills required for deployment/configuration	U-A-M: basic web knowledge
Supported technologies/frameworks	N/A
Supported OS	N/A
List of functionalities	Authentication Community Building Content Management Events Notify Vote Multi-language support Proposals Management Transparent Results and budgeting Citizens Certification
Standards & License	Description
License	N/A
Available License	N/A
Third Party software	N/A
Privacy & Data Protection	Description
Information Sheets regarding collected and mined data	N/A
Information Consent Request	N/A
Privacy & personal data protection security	Description
Authentication	Security Token Authentication
Encryption	N/A
Password recovery	N/A
OAuth	N/A
Open Access	Description
Export of Data	N/A
Social media interaction	N/A
API availability	N/A
Ethical principles	N/A

2.1.14 wmOP

The wmOP [18] is a platform for participatory budget used in several Portuguese municipalities (Leiria, Ponta Delgada, Marinha Grande, among others) being a closed source and proprietary solution. The platform supports the management of the entire participatory budget cycle, from the proposals submissions to the voting phase, while also allowing to follow up on the winning projects during its implementation. The analysis performed to the platform is summarized in Table 15.

Table 15 – wmOP Platform analysis

Functional aspects	
Deployment model	All-in-One
Technical skills required for deployment/configuration	U-A-M: basic web knowledge
Supported technologies/frameworks	N/A
Supported OS	N/A
List of functionalities	Authentication (supporting also citizen card) Community Building Content Management Events Notify Questionnaire Vote (supporting web, phone tablet, SMS and offline) PB and participants Management
Standards & License	Description
License	N/A
Available License	N/A
Third Party software	N/A
Privacy & Data Protection	Description
Information Sheets regarding collected and mined data	Yes in TOS
Information Consent Request	N/A
Privacy & personal data protection security	Description
Authentication	N/A
Encryption	N/A
Password recovery	N/A
OAuth	N/A
Open Access	Description
Export of Data	N/A
Social media interaction	N/A
API availability	N/A
Ethical principles	N/A

2.1.15 Bürgerhaushalt Lichtenberg

Bürgerhaushalt Lichtenberg [19] is a proprietary and closed source participatory budget platform for the Lichtenberg district of Berlin developed by Binary Objects [20] and DEMOS [21]. The platform employs open source content management systems (e.g., Drupal) but is released as closed source. A summary of the analysis of the platform is presented in Table 16.

Table 16 – Bürgerhaushalt Lichtenberg Platform analysis

Functional aspects	
Deployment model	All-in-One
Technical skills required for deployment/configuration	U-A-M: basic web knowledge
Supported technologies/frameworks	N/A
Supported OS	N/A
List of functionalities	Authentication Creation of proposals by district, municipality, geolocation and subject to moderation Proposals, search on map, search by district, subject, year, evaluation, follow proposals, Community Building Content Management Events Vote PB and participants Management Social network share (Facebook, Twitter, Google+) Monitoring of PB proposals: Static Monitoring including title, proposal description, GPS location, date of submission, theme, images Dynamic Monitoring including implementation status, implementation by municipality Qualitative Monitoring including events/milestones, meetings of groups of stakeholders involved in the proposal Data visualization (maps of proposals)
Standards & License	Description
License	N/A
Available License	N/A
Third Party software	N/A
Privacy & Data Protection	Description
Information Sheets regarding collected and mined data	Yes in TOS
Information Consent Request	N/A
Privacy & personal data protection security	Description
Authentication	N/A
Encryption	N/A
Password recovery	N/A
OAuth	N/A
Open Access	Description
Export of Data	No

Social media interaction	N/A
API availability	N/A
Ethical principles	N/A

2.1.16 MyNeighbourhood

MyNeighbourhood [35] is a platform funded by the EU that allows people living close to each other – neighbours – to meet and share events and opinions inside their community. Although closed source, the platform provides an API so that its data can be used and displayed in other platforms. However, other platforms can't insert data in the platform. A summary of the analysis performed to the platform is presented in Table 17.

Table 17 – MyNeighbourhood Platform analysis

Functional aspects	
Deployment model	N/A
Technical skills required for deployment/configuration	N/A
Supported technologies/frameworks	N/A
Supported OS	N/A
List of functionalities	Authentication (through email, password, social accounts) Community Building Social media sharing Content Management Events Notify Questionnaire Vote (like/dislike)
Standards & License	Description
License	N/A
Available License	N/A
Third Party software	N/A
Privacy & Data Protection	Description
Information Sheets regarding collected and mined data	Yes in TOS
Information Consent Request	N/A
Privacy & personal data protection security	Description
Authentication	N/A
Encryption	N/A
Password recovery	N/A
OAuth	N/A
Open Access	Description
Export of Data	N/A
Social media interaction	N/A
API availability	N/A

Ethical principles	N/A

2.1.17 De Olho nas Metas

"De Olho nas Metas" (Eye-to-the-Goals) [36] is an initiative created and promoted by the São Paulo City civil society. The platform helps citizens and organizations from civil society monitor the achievement of São Paulo City Hall "goals" through the monitoring of the projects related in Table 18.

Table 18 – De Olho nas Metas Platform analysis

Functional aspects	
Deployment model	N/A
Technical skills required for deployment/configuration	U: basic web knowledge
Supported technologies/frameworks	N/A
Supported OS	N/A
List of functionalities	Authentication Content Management Notify (email, SMS) Follow and comment goals, projects and companies Monitoring of goals, projects, contracting companies, expenditures: Static Monitoring on goals include title, goal description, technical details, budget allocated, location, theme, area and related projects. On the project monitoring includes, name, location, are in city and related goals. On the contracting companies monitoring includes name, expenditures, contracts and total mount of budget received Dynamic Monitoring including status of sub-tasks and project implementation progress Qualitative Monitoring including milestones and advisers Data visualization: Progress bar Infographics to show goals implementation
Standards & License	Description
License	Creative Commons Attribution-ShareAlike 3.0
Available License	N/A
Third Party software	N/A
Privacy & Data Protection	Description
Information Sheets regarding collected and mined data	N/A
Information Consent Request	N/A
Privacy & personal data protection security	Description
Authentication	Yes
Encryption	N/A
Password recovery	N/A
OAuth	N/A
Open Access	Description
Export of Data	Yes (to CSV, XLS, XML, JSON)

Social media interaction	Facebook and Twitter for sharing
API availability	Yes
Ethical principles	N/A

2.1.18 Monitoraggio Opere Pubbliche

Monitoraggio Opere Pubbliche [37] is a platform developed by nuovole.org to help public administrations monitor the implementation of public works and services. The tool has been used by the Parma Municipality (Italy) during the administration of Major Vignali (2007 – 2011) but never opened to public use. The available features are documented in Table 19.

Table 19 – Monitoraggio Opere Pubbliche Platform analysis

Functional aspects	
Deployment model	N/A
Technical skills required for deployment/configuration	U: basic web knowledge
Supported technologies/frameworks	N/A
Supported OS	N/A
List of functionalities	Authentication Content Management Monitoring of public works: Static Monitoring including title, project information, person in charge, priority, allocated budget, location, expenditures committed and start/conclusion dates Dynamic Monitoring including project implementation progress, sub tasks status, days of delay on expected timeline Qualitative Monitoring including technical issues by municipality Data visualization: Maps Timelines
Standards & License	Description
License	N/A
Available License	N/A
Third Party software	N/A
Privacy & Data Protection	Description
Information Sheets regarding collected and mined data	N/A
Information Consent Request	N/A
Privacy & personal data protection security	Description
Authentication	N/A
Encryption	N/A
Password recovery	N/A
OAuth	N/A
Open Access	Description
Export of Data	N/A
Social media interaction	N/A

API availability	N/A
Ethical principles	N/A

2.1.19 Open Expo 2015

Open Expo – Data and transparency for Expo 2015 [38] is a digital environment aiming at ensuring total transparency of the exposition in 2015. The website allows the consultation of all information concerning the management, planning, organization and carrying out of the event. Open Expo intends to enhance the information that was released by Expo 2015 S.p.A. Within a transparent administration, the data on the economic management of the event (inflows and outflows, purchases, payments and related beneficiaries), the carried out works (worksites, description of the works, amounts envisaged for their carrying out) and the possible variants during the conduction of the event (number of visitors, mobility and public transport) is made available. The available features are documented in Table 19.

Table 20 – Open Expo 2015 Platform analysis

Functional aspects	
Deployment model	N/A
Technical skills required for deployment/configuration	U: basic web knowledge
Supported technologies/frameworks	N/A
Supported OS	N/A
List of functionalities	Authentication Content Management Monitoring of public works: Static Monitoring including, work name, company, prices, subcontractors, variations in work progress, extensions, work progress Dynamic Monitoring including work status, time progress indicator, work progress indicator, potential delay indicator, financial index, time index Data visualization: Maps Dynamic dashboards Progress bars
Standards & License	Description
License	N/A
Available License	N/A
Third Party software	N/A
Privacy & Data Protection	Description
Information Sheets regarding collected and mined data	N/A
Information Consent Request	N/A
Privacy & personal data protection security	Description
Authentication	N/A
Encryption	N/A
Password recovery	N/A
OAuth	N/A

Open Access	Description
Export of Data	Yes, in CSV, XLS and JSON
Social media interaction	Yes, twitter
API availability	N/A
Ethical principles	N/A

2.1.20 Open Ricostruzione

Open Ricostruzione (Open Reconstruction) [39] allows citizens to monitor the implementation of works for the reconstruction of the sites destroyed or damaged by the earthquake which hit Emilia Romagna region (Italy) in 2012. Data is updated every three months and is taken by F.EN.I.C.E system, a web application used by Emilia Romagna Administrative Region to manage the reconstruction works. The available features are documented in Table 21.

Table 21 – Open Ricostruzione Platform analysis

Functional aspects	
Deployment model	N/A
Technical skills required for deployment/configuration	U: basic web knowledge
Supported technologies/frameworks	N/A
Supported OS	N/A
List of functionalities	Authentication Content Management Monitoring of public works: Static Monitoring including, work name, type of work, site owner, implementer, location, total price Dynamic Monitoring including work status, expenditures co-financing, expenditures paid with public funds Qualitative Monitoring including milestones Data visualization: Maps Dynamic dashboards Progress bars
Standards & License	Description
License	N/A
Available License	N/A
Third Party software	N/A
Privacy & Data Protection	Description
Information Sheets regarding collected and mined data	N/A
Information Consent Request	N/A
Privacy & personal data protection security	Description
Authentication	N/A
Encryption	N/A
Password recovery	N/A

OAuth	N/A
Open Access	Description
Export of Data	Yes, in CSV, XLS and JSON
Social media interaction	Yes, Twitter, Facebook, Google + (for sharing)
API availability	YEs
Ethical principles	N/A

2.1.21 OpenSpending

OpenSpending [40] is developed under the framework of OpenBudgets.eu project, a Europe H2020 funded project focused on the visualization and analysis of public budgets and spending data (see section 2.1.25). OpenSpending tracks and analyses public financial information globally. It is meant to be a resource for individuals and groups who wish to discuss and investigate public financial information, including journalists, academics, campaigners, and more. The platform allows community of users and contributors to share datasets on public financial information, including budgets, spending, balance sheets, procurement among others. Originally launched as Where Does My Money Go in 2009, OpenSpending joined forces with other projects in 2011 and expanded to become OpenSpending. Today, the original OpenSpending database has over 1000 datasets representing more than 28 million transactions. The available features are documented in Table 22.

Table 22 – OpenSpending Platform analysis

Functional aspects	
Deployment model	N/A
Technical skills required for deployment/configuration	U: basic web knowledge
Supported technologies/frameworks	D3.js, Raphael, Javascript
Supported OS	N/A
List of functionalities	Authentication Content Management Monitoring of public financial information: static Monitoring including information uploaded by the user, which is described by title of dataset, description the sources of the dataset, the measurement process and the dimensions of the dataset Data visualization can be configured per each dataset: BubbleTree, displays hierarchical spending data TreeMap to compare proportions among different items of spending Aggregation tables
Standards & License	Description
License	N/A
Available License	N/A
Third Party software	N/A
Privacy & Data Protection	Description
Information Sheets regarding collected and mined data	N/A
Information Consent Request	N/A

Privacy & personal data protection security	Description
Authentication	N/A
Encryption	N/A
Password recovery	N/A
OAuth	N/A
Open Access	Description
Export of Data	Yes, in CSV, JSON
Social media interaction	Yes, Twitter, Facebook, Google + (for sharing)
API availability	Yes
Ethical principles	N/A

2.1.22 Bilanciovico

Bilanciovico [41] is a platform to help citizens understand and monitor municipal budget of Vico Pisano Town (Italy). It offers data aggregations on incomes end expenditures and allows users to browse and search data. The available features are documented in Table 23.

Table 23 – Bilanciovico Platform analysis

Functional aspects	
Deployment model	N/A
Technical skills required for deployment/configuration	U: basic web knowledge
Supported technologies/frameworks	N/A
Supported OS	N/A
List of functionalities	Authentication Content Management Monitoring of public financial information: static monitoring including expenditure commitments, contracting company, administrative documents Dynamic Monitoring including expenditure progress Data visualization: Expenditure progress bar
Standards & License	Description
License	N/A
Available License	N/A
Third Party software	N/A
Privacy & Data Protection	Description
Information Sheets regarding collected and mined data	N/A
Information Consent Request	N/A
Privacy & personal data protection security	Description
Authentication	N/A
Encryption	N/A

Password recovery	N/A
OAuth	N/A
Open Access	Description
Export of Data	N/A
Social media interaction	N/A
API availability	N/A
Ethical principles	N/A

2.1.23 Dirolamia

Dirolamia [42] shows the implementation progress of the winning projects in the PB of Rho (Italy). The available features are documented in Table 24.

Table 24 – Dirolamia Platform analysis

Functional aspects	
Deployment model	N/A
Technical skills required for deployment/configuration	U: basic web knowledge
Supported technologies/frameworks	N/A
Supported OS	N/A
List of functionalities	Authentication Content Management Monitoring of winning projects of the PB process: Static Monitoring including title, proposal description, budget, proponent, theme Dynamic Monitoring including implementation progress Qualitative Monitoring including milestones Data visualization: Progress bar
Standards & License	Description
License	N/A
Available License	N/A
Third Party software	N/A
Privacy & Data Protection	Description
Information Sheets regarding collected and mined data	N/A
Information Consent Request	N/A
Privacy & personal data protection security	Description
Authentication	N/A
Encryption	N/A
Password recovery	N/A
OAuth	N/A
Open Access	Description
Export of Data	N/A

Social media interaction	N/A
API availability	N/A
Ethical principles	N/A

2.1.24 Monzapartecipa

Dirolamia [42] shows the implementation progress of the winning projects in the PB of Monza (Italy). The available features are documented in Table 24.

Table 25 – Monzapartecipa Platform analysis

Functional aspects	
Deployment model	N/A
Technical skills required for deployment/configuration	U: basic web knowledge
Supported technologies/frameworks	N/A
Supported OS	N/A
List of functionalities	Authentication Content Management Monitoring of winning projects of the PB process: Static Monitoring including title, proposal description, budget, proponent, theme, town area, expected end date Dynamic Monitoring including implementation progress and status Qualitative Monitoring including milestones Data visualization: Aggregated tables
Standards & License	Description
License	N/A
Available License	N/A
Third Party software	N/A
Privacy & Data Protection	Description
Information Sheets regarding collected and mined data	N/A
Information Consent Request	N/A
Privacy & personal data protection security	Description
Authentication	N/A
Encryption	N/A
Password recovery	N/A
OAuth	N/A
Open Access	Description
Export of Data	N/A
Social media interaction	N/A
API availability	N/A
Ethical principles	N/A

2.1.25 OpenBudgets.eu

The OpenBudgests.eu [32] is a Europe H2020 funded project focused on the visualization and analysis of public budgets and spending data. As an example, this project has made several EU funding reports for diverse areas (social, ICT). Still under development, the project will result in an open source framework to help not only municipalities but also public and non-governmental organizations to promote fiscal transparency of the entities' budgets.

The project aims to develop three main components:

- **Semantic data model** – This allows for the analysis and integration of the budgets' data in the platform enabling the comparison between different budgets.
- **Libraries of visualisation** – Different tools to visualise and compare the results allow users with different levels of expertise to perform their own comparisons and evaluations of the existent data.
- **Feedback** – Allow users to provide data and feedback to the existent budgets or even on particular transactions.

2.1.26 DigiWhist

The Digital Whistleblower (DigiWhist) project [33] is funded by the H2020 programme and aims to enable fiscal transparency, risk assessment and the impact of good governance. In this context, the principles adhering to transparency, open data and accountability mechanisms are crucial and led to the design of opentender.eu platform [34], which is not available yet.

The project will be conducted simultaneously in 35 jurisdictions, the 28 EU Member States, the European Commission itself and some other European Countries. During its development, the project will gather legal and regulatory norms from the different members, collect open and structured datasets with organisations' data and develop indicators for transparency, corruption and administrative quality. Mobile apps and web portals will be created to provide access to the collect data and allow the submission of other relevant information, such as whistle-blower reports.

2.2 Summary

Several platforms and tools exist for participation, while some of them have enhanced features, others resort simply to promote the discussion of ideas and do not integrate tools to promote collaboration and involvement of users in projects. Additionally, some of these platforms and tools are based on commercial license models which limits their adaptation by municipalities and NGOs. For instance, CitizenBudget has a closed source license model which does not promote the interoperability between the platform and others tools that may exist.

After performing an analysis on all the platforms, a summary of the most important aspects identified is presented in Table 26. Since OpenBudgets.eu and DigiWhist are only aimed at governmental and fiscal transparency, and do not support participation actions, they are not presented in the summary, although their work is also important for EMPATIA project.

In the feature set, the Proposals feature includes the projects that are inside the PB management cycle. The feature of Ideas includes mainly concepts that are brought to discussion before a PB cycle by citizens or entities. The ideas can be used to prioritize the budget in different areas of a PB and/or can originate divers proposals. Both of ideas and proposals can have discussions and debates.

With the goal to increase the participation of citizens in the municipality projects to improve life conditions in the municipality, the identification of issues is relevant. For instance, the identification of holes in the roads, in the sidewalks or other similar issues improves the situation awareness of municipalities regarding the status of the diverse infrastructures they manage. The comments are transversal to all the features (e.g., proposals, ideas) and allow users to express their opinion, feelings.

Table 26 – Platform analysis summary

Feature	Your Priorities	Lomio	CitizenBudget	LiberOpinion	AppCivist	D-CENT	BiPart	Liberopinion	openDCN	Bürgerhaushalt	wmOP	Participare	Parelon	MyNeighbourhood
Proposals	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✗
Proposals' collaboration	✓	✓	✓	✓	✓	✓	✗	✗	✗	✗	✗	✗	✗	✗
Ideas	✓	✓	✓	✓	✗	✗	✓	✓	✓	✗	✗	✗	✓	✗
Discussions/Debates	✓	✓	✓	✓	✗	✗	✓	✗	✓	✗	✗	✗	✗	✓
Comments	✓	✓	✓	✓	✗	✓	✓	✗	✓	✓	✗	✗	✗	✓
Geotagging	✓	✓	✓	✓	✓	✗	✓	✓	✓	✓	✗	✗	✗	✗
Online voting	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✗
Offline voting	✗	✗	✗	✗	✗	✗	✗	✓	✗	✓	✓	✓	✗	✗
Issues	✓	✓	✓	✓	✓	✗	✗	✓	✓	✗	✗	✗	✗	✗
Events	✓	✓	✓	✓	✗	✗	✓	✗	✓	✓	✗	✗	✗	✓
Open data	✗	✓	✗	✗	✓	✓	✗	✓	✗	✗	✓	✓	✓	✗
PB Cycle management	✗	✗	✓	✓	✗	✓	✓	✓	✓	✓	✓	✓	✗	✗
Notifications	✓	✓	✓	✓	✗	✓	✓	✓	✓	✓	✗	✗	✗	✗
Social Media Integration	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✗	✗	✓
Citizen card validation	✗	✗	✗	✗	✗	✓	✓	✗	✓	✗	✓	✗	✗	✗
Multi-language support	✓	✓	✓	✓	✗	✓	✗	✗	✗	✗	✗	✓	✓	✗

3 Requirements

This section identifies the requirements for the EMPATIA Platforms. The requirements are specified considering the supported features of the tools, platforms and projects analysed in section 2, the EMPATIA use case scenarios described in Deliverable D1.1 [2] and the development progress reported in Deliverable D2.2 [3]. This document completes the initial requirements that have been specified in Deliverable D2.1 [1], the new requirements are identified with ***** for instance OR* identifies new requirements.

The requirements are split in the following categories:

- **Operational requirements**: define the environment and the actors that interact with the platform. Actors can include municipalities, entities and citizens in general.
- **Design requirements**: identify the practices to build, code and implement software components.
- **Performance requirements**: express the levels of quality on how the technical requirements must be supported. For instance, if the platform supports authentication, then this support must be offered in the same levels when performed for 50 or 500 thousand users.
- **Technical requirements**: technological aspects, core or ancillary functionalities requirements. The core functionalities describe the functionalities that a platform needs to support. Given the scope of this deliverable, these requirements are extensively specified, in order to enable an effective design of the EMPATIA architecture platform.

The technical requirements are organized in two sub-categories:

- **Functional requirements**: refers to functions of each component and module.
- **Non-functional requirements**: components and modules characteristics.

Additionally, for each technical requirement a priority level:

  - MUST: requirement that needs to be implemented.
  - SHOULD: requirement that is relevant, but is not required for a first version of the platform.
  - NICE TO HAVE: requirement that is interesting to be implemented.

Must and should requirements are mostly requirements that were identified in one of the EMPATIA pilots, or are identified in all EMPATIA use case scenarios. Nice to have correspond to the ones that are tied to a specific use case and is not required for it to be implemented.

In addition, the functional requirements are grouped by classes of requirements that relate between them (e.g. authentication, content management, community building).

3.1 Operational requirements

The operational requirements derive from the environment and actors’ interactions. The operational requirements are summarized in Table 27.

Table 27 – Operational Requirements

ID	Requirement	Description

OR 1	Support diverse actors	Must support government agencies, public organizations, municipalities, citizens with different ICT skills.
OR 2	Be accessible to all the stakeholders	Besides supporting different types of users, also needs to improve access to all the stakeholders, no matter the location they are (if at home, at work), and no matter the device they use to access the platform (computers, tablets, mobile phones, dedicated hardware platforms or others [44].
OR 3	Be operational	Must be operational during the lifetime of running project proposals, ideas gathering. Also the platforms must work beyond legal mandates of governments in municipalities [44].
OR 4	Support the technical analysis by municipalities	Municipalities must have the possibility to inspect, analyse the proposals of ideas, projects made by citizens. Indeed, the platform must support the technical analysis of proposals, considering restrictions related to budget, legal aspects among others.
OR* 5	Support different roles and permissions for actors	As actors can be of different types (e.g. general citizens, politicians legislative, politicians executive) different roles must be supported.
OR* 6	Support transparency and accountability	The component must support transparency and accountability in public administration, through the engagement of both its key-actors and others external stakeholders from civil society.
OR* 7	Be "independent from the first cycle"	The component must work independently from a PB first cycle.

3.2 Design requirements

The design of a participation platform involves the participation of multidisciplinary teams with specific specializations, including developers, designers, experts in PB processes, and final users (e.g. municipalities). The design of such platform must fulfil the requirements identified in Table 28.

Table 28 – Design Requirements

ID	Requirement	Description
DR 1	Support multidisciplinary teams	Requires the participation of teams with different skills, including software engineers, designers, PB experts, municipalities/NGOs and end-users. The communication flow between team members must be efficient and objective, considering the goals of the platform design.
DR 2	Support functional testing	Functionalities can only be considered as supported when their coding in software components has been tested and results are according the expected output. For instance, the testing of authentication must be successful with valid credentials and unsuccessful with invalid credentials.
DR 3	Support agile methodologies	The implementation must follow incremental activities, where base functionalities are implemented and tested in first place, while extra functionalities (e.g. nice to have) are implemented

		with less priority. The agile methodologies must provide several releases of the platform in a timeframe, incorporating correction of malfunctions from previous releases.
DR 4	Address in early phases feedback from users	End users, including citizens, municipalities must have access to early releases in order to provide useful user feedback that can enhance the platform and meet the users' expectations.

3.3 Performance requirements

The performance requirements are related with the technical requirements, although the performance expresses the quality levels on which technical requirements must be fulfilled, as described in Table 29.

Table 29 – Performance Requirements

ID	Requirement	Description
PR 1	Be available to a high number of users	Must, at least, support the access by all citizens/users in legal conditions (e.g. adults who can vote for electing government for municipality). The user experience with a low (e.g. 20) and a high number of simultaneous users (e.g. 1000) must be the same.
PR 2	Incorporate fault tolerance mechanisms	Components implementing the specified functionalities must support fault tolerance mechanisms, to ensure high levels of availability. Platform deployments must support redundancy of core components, not to compromise services in case of malfunction or errors.
PR 3	Support different deployment models	Should not impose restrictions regarding deployment models. For instance, one may opt to run the platforms on their ICT infrastructure, or use a public cloud infrastructure.

3.4 Functional requirements

This section presents the technical requirements of the EMPATIA Platform. The requirements are organized per component. Lastly, a general requirements section is presented, with requirements that are common to all the components and establish the foundations for the platform.

3.4.1 Analytics

The analytics component includes the requirement to allow the analysis of voting events, as specified in Table 30.

Table 30 – Functional Requirements Analytics

ID	Requirement	Description	Priority
FR1	Information of Votes	Information of Vote Event	Must
FR2	Information of Daily Votes	Information of daily vote event	Must
FR3	Statistics of Votes by gender	Total, first, second and statistics of votes by gender	Must

FR4	Statistics of Votes by profession	Total, first, second and statistics of votes by profession	Must
FR5	Statistics of Votes by channel	Total, first, second and statistics of votes by channel	Must
FR6	Statistics of Votes by Neighbourhood	Total, first, second and statistics of votes by Neighbourhood	Must
FR7	Statistics of Votes by Age	Total, first, second and statistics of votes by Age	Must
FR*8	Statistics by Date	Votes analysis by date	Should
FR*9	Statistics by Town	Votes analysis by town	Should
FR*10	Statistics by Education	Votes analysis by education	Should
FR*11	Proposal Analysis	Allow the analysis of proposals considering diverse criteria (e.g. Gender, Location, Channel)	Should
FR*12	Ideas Analysis	Allow the analysis of ideas considering diverse criteria (e.g. Gender, Location, Channel)	Should
FR*13	Issues Analysis	Allow the analysis of issues considering diverse criteria (e.g. Gender, Location, Channel)	Should
FR*14	Participation Analysis	Allow the analysis of participation considering diverse criteria (e.g. Gender, Location, Channel)	Should
FR*15	Customization of dashboards by entity	Allow entities to customize dashboards by adding/removing elements, modifying layouts	Nice to have
FR*16	Support elaboration of reports	Allow entities to export statistics into reports (in PDF, XLS, Word format)	Nice to have

3.4.2 Authentication & Authorization

Users/citizens, managers/municipality staff and administrators have to access the platform and perform actions. To support these features and assure the appropriate level of security, authentication and authorization specific requirements are described in Table 31. The Create, Read, Update, Delete (CRUD) operations represent the major operations that can be performed for a certain functionality.

Table 31 – Functional Requirements Authentication

ID	Requirement	Description	Priority
FR17	Users	Management of users	Must
FR18	Login with Credentials	Support login/password login	Must
FR19	Login with QRCode	Support QRCode login	Should

FR20	Login with RFID cards	Support RFID cards login	Should
FR21	Login with Alphanumeric code via SMS	Support Alphanumeric code login for easier and faster login, using SMS	Should
FR22	Login with Identity card	Login with the citizen's identification card	Nice to Have
FR23	Social Media Integration - Facebook	Login with Facebook	Should
FR24	Social Media Integration - Twitter	Login with Twitter	Nice to Have
FR25	Social Media Integration – Google +	Login with Google +	Nice to Have
FR26	Get users public details	Get user public details from user ids	Must
FR27	Support expiration mechanisms of authentication	Support expiration of authentication tokens to avoid security issues	Must
FR28	Support open standards for authorization	Support OAuth, OPenID standards to promote the information flow between cross-platforms	Should
FR29	Support authorization policies	Support the specification and enforcement of authorization policies based on diverse granular levels, For instance, per component, per proposal or per user role.	Must
FR30	Login with Passport	Login with Passport ID	Nice to Have
FR31	Login with Resident Card	Support login with resident card emitted by municipality	Nice to Have
FR32	Support dual-factor authentication	Support secure mechanisms for authentication, which integrate more than one mechanism (e.g., password + SMS).	Should
FR33	Support policies to renew credentials	Support policies that require renew of passwords after a certain period (e.g. each 3 months), requires passwords to be different than the 4 previous ones.	Should
FR*34	Support Single-Sign-On (SSO)	Support SSO with other platforms already existing in municipalities for PB	Nice to Have
FR*35	Support One-time generated password	Support Open Authentication (OATH) [45].	Nice to Have
FR*36	Triple-factor authentication for Administrator and super manager roles	Increase security levels of administrator and super manager roles.	Nice to Have

3.4.3 Content Management

Based on the described analysis, many of the participation events require a public website for dissemination of the actions and rules. The dynamic management of participation websites is required for EMPATIA Platform. The functionalities related to content management are presented in Table 32.

Table 32 – Functional Requirements Content Management

ID	Requirement	Description	Priority
FR37	Sites	Management of sites	Must
FR38	Pages	Management of pages	Must
FR39	News	Management of news	Must
FR40	Events	Management of events	Must
FR41	Menus	Management of menus	Must
FR42	Text content	Support text in pages and on other content containers. Text can also be supplied in standard document formats, such as doc, pdf.	Must
FR43	Image content	Support images in pages with different formats, including (jpeg, png, pdf, gif) and including the possibility to add the images.	Must
FR44	Video content	Support videos in pages with MPEG4, WebM and Ogg encoding.	Must
FR45	Content in HTML 5 format	Enable the support of HTML 5 media types, avoiding the requirement of installing additional plug-ins for playing video or other type of content	Nice to Have
FR46	Edit pages, news and events content in a simple and intuitive way	Support WYSIWYG text editors	Must
FR47	Newsletter	Support management of newsletter	Should
FR48	Reports	Support management of reports, including customization of fields to be included in reports	Should
FR49	Export of information	Support export of information in different format including PDF, DOC, ODF, XML	Nice to have
FR50	Support online edition of documents	Support online edition of documents similar to google docs	Nice to have
FR51	Support share	Support sharing of content, with personalized access controls	Nice to have

FR52	Access control	Support the access to information according to defined policies and roles	Should
FR*53	Actual News	Get actual news	Should
FR*54	Last News	Get the most recent news	Should
FR*55	Last Events	Get the last inserted events on the platform	Should
FR*56	Reorder Menus	Support customization of menus	Should
FR*57	Order Files in Lists	Support ordering of files in lists	Should
FR*58	Version Control of pages, news and events	Show version of pages, events and news	Should
FR*59	Publish/Unpublish content	Publish and unpublish content (e.g. news, events among others)	Should

3.4.4 PAD

The PAD component corresponds to the initial community building component. PAD aims to foster the participation and collaboration between citizens, communities and entities. Such collaboration includes forums, posts and space for discussion ideas and proposals. The community building functionalities are presented in Table 33.

Table 33 – Functional Requirements PAD

ID	Requirement	Description	Priority
FR60	Forums	Support forums, including their management.	Must
FR61	Ideas	Support ideas, including their management	Must
FR62	Proposals	Support proposals, including their management	Must
FR63	Discussions	Support discussions, including their management	Must
FR64	Comments	Support comments in all participation features	Must
FR65	Forums moderation	Control forum moderators: assign users, set permissions, etc.	Must
FR66	Ideas moderation	Control ideas moderators: assign users, set permissions, etc.	Must
FR67	Discussions moderation	Control discussions moderators: assign users, set permissions, etc.	Must
FR68	Report abuse	Support a transparent and effective report abuse feature that allows users to report abuses in participation	Nice to have

		sections and enable the blocking and/or editing of participations by moderators.	
FR69	Advanced search mechanisms	Implement search mechanisms that allow to find proposals, ideas, topics, discussions based on several data types.	Nice to have
FR70	Voting Analysis	Gender, Location, Socioeconomics	Must
FR71	Proposal Analysis	Gender, Location, Socioeconomics, number of participants, number of comments, number of associated ideas	Must
FR72	Public consultation	Support public consultation of proposals without requiring login	Should
FR73	Thematic consultation	Support consultation of proposals according to certain themes	Should
FR74	Access Control	Comments, new ideas, voting processes must be performed according to defined roles and policies	Must
FR75	Cooperation between ideas and proposals	Support mechanisms to identify, analyse ideas that are more interesting for a certain PB, according to several criteria	Nice to Have
FR76	Ideas Analysis	By gender, location, socioeconomics, number of participants, number of comments	Must
FR77	Issues	Support issues, including their management	Must
FR78	Issues Analysis	By type of issue, grouping issues by similarity, by location and others	Nice to have
FR79	Participation analysis	Analytics identifying users with more participation, neighbourhoods with more issues	Nice to Have
FR*80	List of PAD moderators	List of PAD moderators	Should
FR*81	Block/Unblock PADs	Block and unblock PADs	Should
FR*82	Manage Community Building and associated parameters	Manage PADs and their respective parameters	Should
FR*83	Version Control of Posts	Manage version controls of posts	Should

3.4.5 Design

This component corresponds to the initial Multichannel Participation (MP) component, which functionalities are related with the creation of participation models. The requirements are presented in Table 34.

Table 34 – Functional Requirements Design

ID	Requirement	Description	Priority
FR84	Define participation model	Configure the steps and necessary validations of the participation model, including the technical analysis, dates for participation, among other configuration parameters.	Must
FR85	Collaborative proposals	Manage proposals, define collaboration group, support merge/fork proposals, etc.	Must
FR86	Technical analysis of proposals	Municipalities must have the support to perform a technical analysis of proposals, in order to determine their feasibility according the several factors, such as budget, priority, law enforcement, among others.	Must
FR87	Define participation rules	Specification of rules that are used in different multichannel participation projects.	Must
FR88	Define access control policies	Specification of access control policies for the participation rules. For instance, a member can only perform view operations, while administrator can change rules and other sets.	Must
FR89	Define channels for participation	The proposals can include different channels for participation: platform supporting forums, discussions, specific hardware components such as kiosks for voting	Should
FR90	Specification of fraud detection	Include mechanisms to detect fraudulent participation of users (e.g., users compelling others to vote on their ideas, or proposals).	Should
FR*91	Design Template	Support of Multichannel Participation Design Templates	Should

3.4.6 Events

This component manages the creation of events and the associated payments and registrations. The requirements are presented in Table 35 .

Table 35 – Functional Requirements Events

ID	Requirement	Description	Priority

FR92	Events management	Manage events like conferences, workshops, including registration and possible required payments	Must
FR*93	Management of Sponsors	Manage sponsors of events	Should
FR*94	Management of Payments	Manage payments related with the participation in events	Should
FR*95	Verification of Registrations	Method to verify registrations, including verification of payments and other aspects	Should
FR*96	Management of Speakers	Manage speakers in events, keynotes, in sessions	Should
FR*97	List Speakers by Sessions	List speakers of sessions	Should
FR*98	Export events to calendar	Support export of events to calendar (i.e. ics format)	Should
FR*99	Download to pdf	Support download to pdf	Should

3.4.7 Files

This component manages the creation of events and the associated payments and registrations. The requirements are presented in Table 36.

Table 36 – Functional Requirements Files

ID	Requirement	Description	Priority
FR100	File upload	Support upload of files with different types, including images, videos or others.	Must
FR101	File download	Download as attachment or inline	Must
FR102	Chunked upload	Upload file in chunks when files are big	Should
FR*103	File list	Get list of files	Should
FR*104	Index Images	Support index of images for improved rendering	Should
FR*105	Search Files	Support search of files	Should
FR*106	Manage file usage and access rights	Support file usage and access rights	Should

3.4.8 Logs or Monitoring

This component manages all the logging functionalities of the platform. The requirements are presented in Table 37.

Table 37 – Functional Requirements Monitoring

ID	Requirement	Description	Priority
FR107	Management of logs at Database	CRUD logs in database	Must
FR108	Management of logs at Graylog	CRUD performance logs in database	Must
FR*109	Performance graphs of components	Support dashboard with infographics regarding performance of platform	Should
FR*110	Export performance data	Export performance data in JSON, including performance data of component and of system such as CPU, memory, disk usage)	Should
FR*111	Support multiple datasource of logs	Support logs from components, from diverse systems	Should

3.4.9 Notifications

This component manages the participation model for the participation channels. The requirements are presented in Table 38.

Table 38 – Functional Requirements Notifications

ID	Requirement	Description	Priority
FR112	Emails management	CRUD for default emails content (Registration confirmation, etc.)	Should
FR113	Notifications management	CRUD and definition of different types of notification mechanisms	Must
FR*114	Push notifications	Send SMS notifications	Should
FR*115	User subscribed notifications	Support follow up notifications	Should
FR*116	Time notifications	Support automatic notifications when deadline of events is approaching	Nice to have

3.4.10 Orchestrator

The Orchestrator component orchestrates the communication between the diverse components and allow entities, such as municipalities to customize the platform to meet their needs. The functional requirements of the orchestrator are detailed in Table 39 and are also related with the initial requirements of other components, reported in deliverable D2.1.

Table 39 – Functional Requirements Orchestrator

ID	Requirement	Description	Priority
FR117	Management of Menus per entity	Associated with FR20 – Management of Menus	Must

FR118	Management of Authentication Methods per entity	Associated with requirements to support multiple authentication methods (i.e. FR2 – with credentials, FR3 – with QRCode, FR4 – with RFID Cards, among others)	Must
FR119	Management of User Roles per entity	CRUD roles and permissions (initial FR86)	Must
FR120	Countries management per entity	CRUD countries (initial FR84)	Must
FR121	Currencies management per entity	CRUD currencies (initial FR82)	Must
FR122	Discussion Management per entity	Associated with FR35 - discussions management	Must
FR123	Entities management	CRUD entities	Must
FR124	Languages management	CRUD languages. Enable to define default language according the country where municipality is based	Must
FR125	Management of PADs Entities	To map PADs for each entity associated with requirements FR32, FR33, FR34 an FR35.	Should
FR126	Management of Forums per entity	Associated with requirement FR32	Should
FR127	Management of Geographic Areas	Associated with FR80	Should
FR128	Management of User Groups per entity	Manage the user groups, for instance departments, associated with FR80 and FR86	Should
FR129	Management of Ideas per entity	Associated with FR33	Should
FR130	Management of newsletter subscriptions per entity	Associated with FR26	Should
FR131	Management of permissions per entity	Associated with FR86	Should
FR132	Management of proposals per entity	Associated with FR34	Should
FR133	Management of public consultation per entity	Associated with FR48	Should
FR134	Management of roles per entity	Associated with FR86	Should
FR135	Time zones management	CRUD time zones, related with FR85	Must
FR*136	Management of PADs parameters templates	Manage the parameters of templates for PADs	Should
FR*137	Management of Co-Construction	Manage work in groups	Should

FR*138	Management of layouts per entity	Manage the layouts and interfaces in WUI component	Should
FR*139	Management of Home pages types	Manage the types of home pages for each entity	Should
FR*140	Management of Kiosks per entity	Enable the use of kiosks for an entity	Should
FR*141	Management of Kiosks types per entity	Manage which kiosks can be used by an entity	Should
FR*142	Management of Modules	Manage which components are activated for an entity	Should
FR*143	Management of Modules Types	Manage which options and other configurations items in components	Should
FR*144	Management of pages per entity	Manage content management pages	Should
FR*145	Management of PAD’s parameters	Manage PAD’s parameters	Should
FR*146	Management of PAD’s parameters Options	Manage the options of PAD’s parameters	Should
FR*147	Management of PAD’s parameters types	Manage the fields to apply to each PAD	Should
FR*148	Management of User parameters per entity	Manage user parameters	Should
FR*149	Management of User parameters options per entity	Manage the options of user parameters	Should
FR*150	Management of Site configurations per entity	Manage site configurations in CM component	Should
FR*151	Management of Site configurations groups per entity	Manage groups of site configurations in CM component	Should
FR*152	Management of Sites per entity	Manage site in CM component	Should
FR*153	Management of Translations configurations per entity	Manage translations configurations for contents held by the CM component	Should
FR*154	Management of thematic consultations per entity	Manage thematic consultations in the PAD component	Should
FR*155	Management of users per entity	Manage of user per entity, associated with FR1	Must
FR*156	Management of vote methods per entity	Manage of vote methods, associated with FR44	Must

3.4.11 Questionnaires

The functional requirements of questionnaires are detailed in Table 40. The questionnaires are an important component, as these allow to gather input from users.

Table 40 – Functional Requirements Questionnaires

ID	Requirement	Description	Priority
FR157	Questionnaires	Management of questionnaires	Must
FR158	Question groups	Management of question groups	Must
FR159	Questions	Management of questions and question types (open answers, check list, radio buttons, ordered list of options, etc.)	Must
FR160	Form builder	Create new questionnaires with all its elements (question groups, questions, answers) dynamically	Must
FR161	Anonymous answers	Support anonymous questionnaire answers	Must
FR162	Answers reports and analytics	Provide detailed or summary information of each questionnaire answers. Support the creation of answers reports and analytics	Nice to have
FR163	Export answers	Export to the following formats: - Excel - PDF - Word - ODF - CSV - JSON - Others, like R, SPSS, SAS	Nice to have
FR164	Data anonymization	When required, ensure data analysis and export take into consideration of data anonymization to comply with confidentiality requirements	Must

3.4.12 Vote or Voting

The vote or voting component assures all the functionalities related with votes. The requirements of this component are summarized in Table 41.

Table 41 – Functional Requirements Vote

ID	Requirement	Description	Priority
FR165	Vote sessions management	CRUD vote sessions	Must
FR166	Vote methods management	CRUD vote methods	Must

FR167	Set/unset votes	Register and cancel votes, keep information regarding vote modifications (timestamps)	Must
FR168	Anonymous voting	Support anonymous voting for certain proposals, ideas.	Should
FR*169	List Votes	List votes per session or per voting event	Should

3.4.13 WUI

The WUI component acts as the entry point of the EMPATIA, making the interface with all the components. The functional requirements of WUI are summarized in Table 42, it should also be noticed that this component is also related with non-functional requirements (e.g. security, performance, and others).

Table 42 – Functional Requirements WUI

ID	Requirement	Description	Priority
FR170	Support simultaneous users	Support the usage of the platform by simultaneous users	Must
FR171	Correct handling of requests	Forward the information to the correct component	Must

3.4.14 Kiosk

The kiosk component (in terms of software) assures all the functionalities with the hardware components of kiosks. The requirements of this components are summarized in Table 43.

Table 43 – Functional Requirements Kiosk

ID	Requirement	Description	Priority
FR*172	Support multiple authentication methods	Enable authentication with RFID cards, and other mechanisms	Must
FR*173	Support authorization policies	Allow users to logout after a certain period of inactivity	Must
FR*174	Define channels for participation	Support votes in sessions and events	Must
FR*175	Show proposals	Show information of proposals according to the type of kiosk	Must
FR*176	Support multiple languages	Support multiple languages to provide information of proposals	Must

3.4.15 Open Data

The Open data component aims to assure export and import of data functionalities to promote the interoperability with other platforms and to adhere to the Open Data principles as described in the DMP of EMPATIA. The functional requirements of Open Data are summarized in Table 44.

Table 44 – Functional Requirements Open Data

ID	Requirement	Description	Priority
FR*177	Support CKAN APIs	Support CKAN APIs to allow import and export of data	Must
FR*178	Support High Availability	Support High Availability of data	Must
FR*179	Support Dataset relationships	Enable the support of dataset relationships, for cross-relation events	Should

3.4.16 2nd PB Cycle

The 2nd PB Cycle component aims to manage the 2nd cycle of participatory budget projects . The functional requirements of 2nd PB cycle are summarized in Table 45 .

Table 45 – Functional Requirements 2nd PB Cycle

ID	Requirement	Description	Priority
FR*180	Implementation monitoring	Support monitoring of 2nd PB cycle, including static, dynamic and qualitative monitoring	Must
FR*181	Community building and co-implementation	Support Civic and communities' engagement	Must
FR*182	Evaluation management	Support management of surveys, questionnaires	Must

3.4.17 Community Building

The Community Building component does not directly correspond to the CB component initially specified in D2.1, such correspondence is for PAD. The functionalities of this component is related with the management of a community and its members. The functional requirements of CB are summarized in Table 46.

Table 46 – Functional Requirements Community Building

ID	Requirement	Description	Priority
FR*183	Information community	Manage information for a community	Must
FR*184	News inside a community	Manage news targeting a community	Must
FR*185	Events inside a community	Manage events inside a community	Must
FR*186	Ideas in a community	Manage and promote discussion of ideas, proposals in a community	Must

FR*187	Blog	Allow blogging of members in a community	Should
FR*188	Members management	Manage users in a community, their enrolment, their roles	Must

3.4.18 General Requirements

As stated previously the general requirements correspond to functionalities that are not mapped directly to a group of requirements, and that can be common to several of the identified requirement groups. Such requirements are summarized in Table 47.

Table 47 – Functional Requirements General

ID	Requirement	Description	Priority
FR189	Export Open data	Export require reports with dynamic content fields	Should
FR190	Charts management	Create charts with dynamic selection of fields to present and correlate	Should
FR191	Implementation follow-up	Provide proposals' implementation data, such as, status, problems, etc. Allow citizens to be involved in the implementation process of projects	Nice to Have
FR192	Currency conversion	Support conversion between different currencies. For instance, between USD and EUR	Should
FR193	Connect with other users	The equivalent of Facebook friending (bilateral feed of performed actions, send and receive)	Should
FR194	Follow other users	The equivalent of Twitter following (unilateral feed of performed actions., receive)	Should
FR*195	Expenditures management	CRUD expenditures	Must
FR*196	Challenges Management	CRUD challenges	Should
FR*197	Challenges discussion moderation	Approve/disapprove users discussions/comments regarding challenges	Should
FR*198	List users in challenges	List users adhering to challenges or discussing/commenting	Should
FR*199	Cooperative work	Support Co-implementation features to enable collaborative initiatives between a group of citizens	Should
FR*200	Implementation Monitoring	Components should allow users to collect and share information regarding the implementation phase of projects	Should

FR*201	Support Co-funding models	Support other models of co-funding projects like crowd-funding initiatives	Nice to Have
FR*202	Archive of previous projects	Allow users to collect, search and browse data of projects from previous PB editions	Nice to Have
FR*203	Support Static Monitoring	Support monitoring of specific fields from projects, such as title, description, location, budget among others. To support the monitoring from another person rather the one that created the project	Nice to Have
FR*204	Support Dynamic Monitoring	Support monitoring of project progress in terms of implementation plan (if been followed per established deadlines), if affected resources are according the plan as well as expenditures	Nice to Have
FR*205	Support Qualitative Monitoring	Support qualitative monitoring of project progress	Nice to Have

3.5 Non-functional requirements

The non-functional requirements include requirements that ideally must be supported by all the modules/components of a platform. For instance, if a component exposes a security breach the overall platform might be compromised. The non-functional requirements are summarized in Table 48.

Table 48 – Non-Functional Requirements

ID	Requirement	Description	Priority
NFR1	Auditability	Must register all actions from users and services to allow system audits	Must
NFR2	Scalability	The platform must be scalable according to the load introduced (scale-in and scale-out): - High simultaneous number of users - High number of simultaneous institutions	Must
NFR3	Security	The platform must ensure security, confidentiality, authenticity and integrity of all data provided to and from users	Must
NFR4	Usability	The platform must provide efficient and easy to use interfaces	Must
NFR5	Interoperability	The platform should provide well-define APIs in order the easily integrate with other platforms. This is particular relevant for open data support.	Must

NFR6	Configurability	The platform must support certain levels of customization, which can include the options in menus, layout adaptations, among others.	Must
NFR7	Flexible deployment	The platform must support different deployment options, which can include cloud based deployment, distributed deployments using diverse virtual machines.	Must
NFR8	Adaptability	The platform and the diverse component must adapt to different conditions, in terms of hardware, environment, number of users to support.	Must

4 EMPATIA architecture

The EMPATIA architecture has been designed to fulfil the requirements identified in the previous section. Key design concerns were considered from the beginning of the architecture specification, which include the need to be modular, scalable and secure. The integration and support for existing and future tools/systems is promoted through the use of public APIs (e.g. maps, translation, currency conversion).

The concepts that have been followed for the design of the architecture are:

- **Spaces**, for users (e.g. citizens), management (e.g. municipal employees) and administration (e.g. EMPATIA team)
- **Community building**, where users participate, in a collaborative way, in forum, ideas/problems, proposals and discussions to express their opinions
- **Multichannel Participations** which implements participation models, which may include community building, voting, questionnaires, budgets, etc.
- **Reports & infographics (analytics)** that includes analytics and statistics
- **Notifications** by email, SMS or other format
- **Social media integration**, with support for facebook and other social media platforms
- **External tools integration**, such as free coin, D-CENT tools, or any other existing tools.
- **Maps integration**, supporting different types of maps, such as Google Maps, Bing, OpenStreetMaps, among others
- **Voting** to allow different voting methods, such as multiple, negative, weighted votes
- **Capacity** supporting courses and games
- **Authentication** to perform authentication via different methods, such as national/international identity cards, SMS, mail, and social media.
- **Visualization** module to produce graphics, reports, tables or documents in PDF, word or Excel format.
- **Currency converter** module to perform the conversion of currency, through XE Currency Converter API [46] and others.
- **Translation** module also acting as a key for internationalization, by supporting Google, Bing and other translation APIs.

Additionally, the EMPATIA architecture is split in several levels:

- Feature: distinct property or piece of technology;
- Tool: group of features that can be used to achieve a specific goal;
- Module: group of features in a component;
- Component: individual block, composed of one or more modules, that can be used standalone (except in some cases), and interacts with other components through APIs;

Platform: full system, with all components.

4.1 High level architecture

Figure 1 depicts the high level of the EMPATIA architecture, with two levels of APIs: a public interface and a private interface for inter EMPATIA components communication (or with external tools). APIs comply with REST and JSON technologies.

The architecture includes components (some based on the requirements groups), following the modular principle, with specific functionalities and interfaces. All components are detailed in the following subsections. The Orchestrator component is responsible for managing the interactions between the different components, and includes specific feature, like addressing many of the authorization requirements.

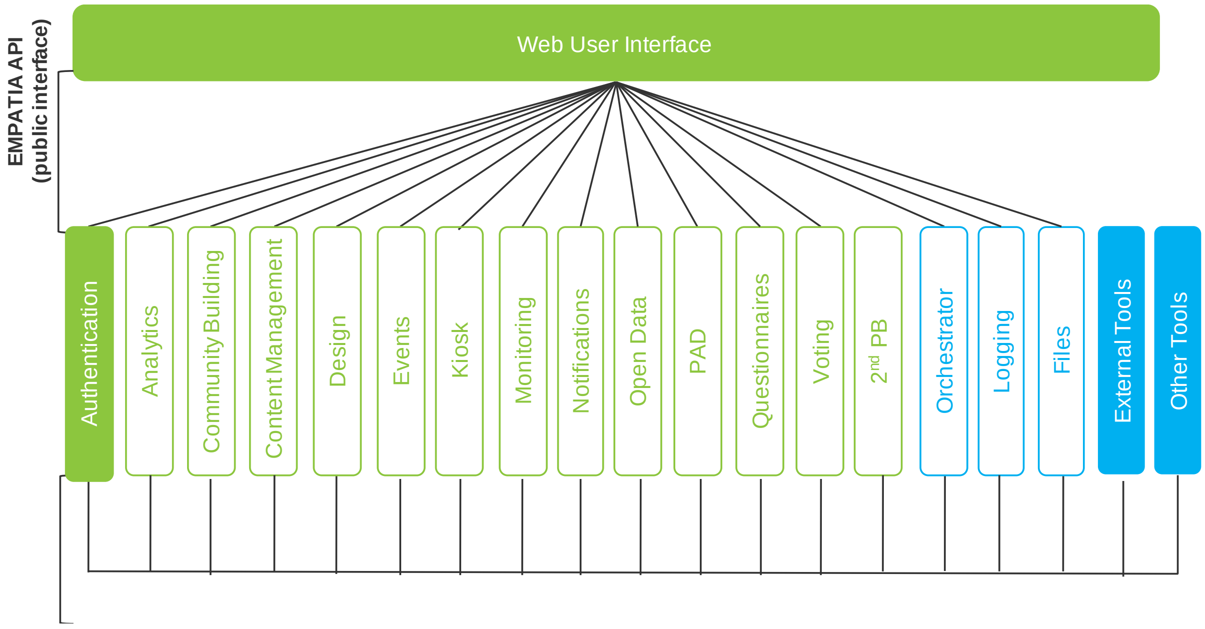


Figure 1 – EMPATIA high level architecture

Each component of the architecture is designed to function independently, and work in a modular fashion, with specific and well identified functionalities.

4.2 Components

To support different deployment models, in a single machine (all-in-one) or in several virtual machines (public or private clouds), the components are designed to be scalable and fully redundant. Each component is composed of: one NoSQL Redis databases to manage component session information and information caches (for performance optimizations), one MySQL persistent database, and when required, a high performance and high availability shared filesystem (like [47]). Figure 2 presents a basic component deployment that does not address scalability, redundancy, nor fault tolerance.

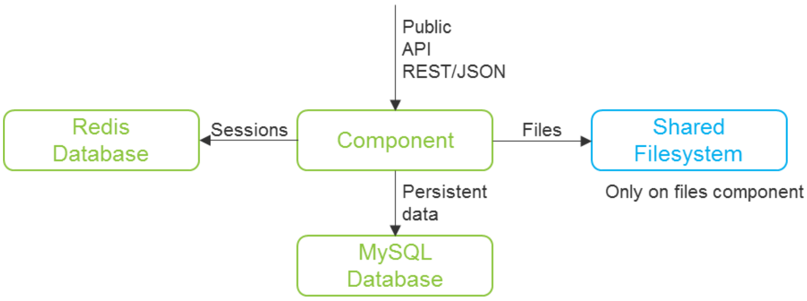


Figure 2 – Components basic architecture

Figure 3 presents an example of a complex component deployment addressing scalability, redundancy and fault tolerance. These goals are achieved through load balancers that receive the public API requests (e.g. through multiple Apache Proxy servers), that forward these requests to one of the multiple component servers (through a secure connection). Redis, MySQL and Hadoop already include scalability, redundancy and fault tolerance mechanisms.

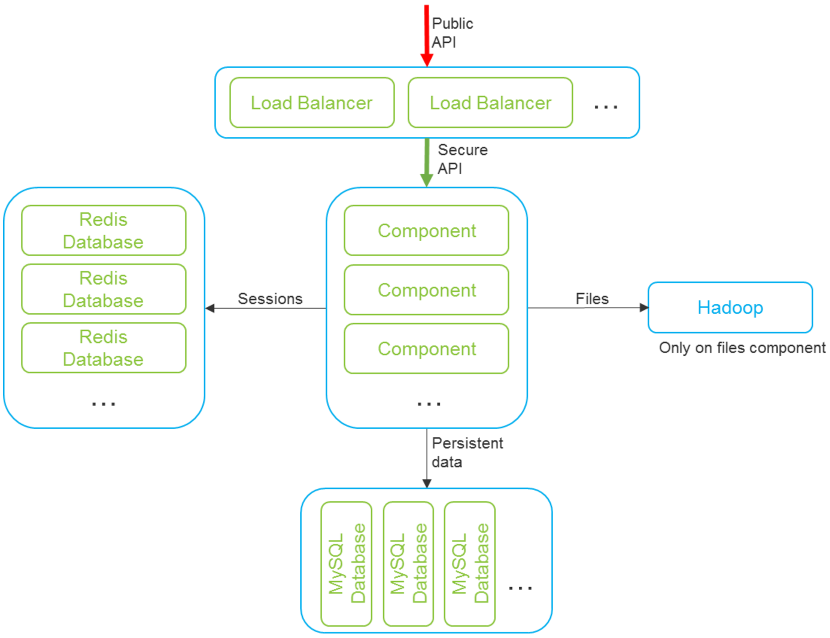


Figure 3 – Components advanced architecture

The number of load balancers, components servers and Redis, MySQL and Hadoop nodes depends on the required levels of scalability, redundancy and fault tolerance required. Also these component can be strategically placed in different geographical areas to address service placement requirements (e.g. deploy each replica of each server in different cloud datacenters to ensure fail over redundancy in case of geographical catastrophe).

In the next subsections are described in more detail each EMPATIA component. Additional components may be added later on to the architecture.

4.2.1 Analytics

The analytics component aims to support two type of analytics: real-time and batch/deferred analytics. In this sense this module extracts data from other components and from the logging component (where monitored Key Performance Indicators (KPIs) are kept).

This component includes the following functionalities:

- Extract data from several sources, which include other components or external tools
- Aggregate and transform collected data according to defined rules
- Present data according defined rules
- Support the definition of rules that enable the processing, transformation and representation of data
- Support streaming of processed data to other components
- Support real-time and batched/deferred analysis

This module also interacts with the WUI component to provide data to allow graphical representation of information in the web interface.

4.2.2 Authentication

The authentication component is responsible for all user personal data and security related functionalities which include:

- Users personal data database (e.g. name, email, address, age, gender)
- Authentication of users
- JSON Web Token (JWT) validation. JWT tokens are employed in all the transactions after a successful authentication of users
- Registration of users in the platform
- Profile management, users details and public information to display
- OAuth server and client
- Verification of users' identity methods
- Customize notification mechanisms (by email, SMS or others)

Given the aforementioned functionalities, this module has a key role in the EMPATIA architecture, since all requests need to be validated. All EMPATIA components use this component to validate the received authentication token and request user personal data (based on restrict EMPATIA rules and security mechanisms).

4.2.3 PAD

The PAD corresponds to the old Community Building (CB) component includes diverse functionalities related to the creation of participation, collaboration and communities, such as forums, ideas/problems and discussions. Despite the differences between these tools, the PAD component does not distinguish them internally (all features are available to all PAD types. Each PAD is composed of topics, posts and comments, that depending on the configuration will perform as required.

Each PAD includes the following functionalities:

- PAD creation, update and delete (CRUD)
- PAD configuration management (such as the definition of moderators, open/close dates)

- Topics creation, update and delete (CRUD)
- Topics configuration management (such as the definition of moderators per topic, open/close topics, deal with report abuses)
- Posts creation, update and delete (CRUD)
- Posts configuration management
- Attachments managements
- Threads managements

The functionalities associated with comments include:

- Abuse report
- Integration with social networks
- Moderation
- Comments/replies

4.2.4 Community Building

The Community Building (CB) component aims to manage all the information inside a community.

Each CB includes the following functionalities:

- CB creation, update and delete (CRUD)
- CB configuration management (such as the definition of moderators, open/close dates)
- CB members management (CRUD of users)
- CB news management
- CB ideas management
- CB blog management
- CB events management

4.2.5 Content Management

The Content Management (CM) component supports all functionalities related with static content, such as, sites, pages, menus, news, and other types of content.

The main functionalities:

- Create, update and delete (CRUD) of sites, pages, menus, news and events
- Menus multiple levels (child-menus) and sort support
- Menu actions (send to content, forum, link, etc.)
- Translations of contents

- Top/side banners (before upload it should be possible to edit the image, since, for instance, banners enforce aspect ratios)
- Versioning of content with the possibility to preview and revert to any specific version
- Set dates and publish/unpublish contents (news and events)
- Edit content using one WYSIWYG tool

4.2.6 Events

The Events component provides all the necessary mechanisms to organize and manage an event, such as a conference or a workshop. To do this, it allows to create events, sessions and associate speakers and sponsors while also giving the possibility to accept registrations and payments from users.

This component includes the following functionalities:

- Create, update and delete (CRUD) Events, Sessions, Speakers and Sponsors
- Management of events, such as included sessions and sponsors
- Management of sessions, such as speakers assigned to each session

Accept registrations and payments to complete or partial events (allow to select desired sessions if permitted)

4.2.7 Files

The Files component is responsible to store all the upload files and to serve them when requested. When a file is upload, the file information, such as name, type and size are stored in a database along with the local file name (securely generated) and a verification code for secure request of the file. It supports chunk upload of files.

4.2.8 Logs or Monitoring

The Logs component is responsible for store all log information from all components. This information is intended for auditing purposes and to provide relevant data to the analytics component.

Although it was first intended to develop a specific logging component, it was later decided to use the open source Graylog2 [48] platform that supports all requirements for this component and is widely used and referenced as a state of the art tool. This tool supports Elastic Search.

4.2.9 Multichannel Participatory

The Multichannel Participatory (MP) component is responsible for the management of a participation model. Through the creation and configuration of CBs, voting phases, other EMPATIA component features, and specific MP parameters (e.g. budget), participation models (e.g. Participatory Budgeting) are transposed to the EMPATIA Platform. Through these configurations (e.g., dates, CBs configurations, budget, rules), each step of the MP perform as expected to achieve the model goal.

4.2.10 Notify

The Notification (Notify) component is responsible to manage different types of communication channels.

This component supports:

- Create, update and delete (CRUD) communication templates
- Create, update and delete (CRUD) communications
- Send and receive communications through different channels (e.g. email and SMS) and from/to different components
- Correlate notifications with specified users, entities, CB and voting processes
- Validate communication user authentication

4.2.11 Orchestrator

The Orchestrator component manages the interaction between all components (that are built to be independent), correlates data between different components, provides management facilities for EMPATIA related configurations, and authorizations.

The orchestrator component supports:

- Management of components per deployment (components' location and access link)
- Management of components authorization (by using access tokens, it verifies if one component can be accessed by another component while serving a user from a specific entity)
- Create, update and delete (CRUD) entities, languages, currencies, countries, time zones, authentication methods, voting methods
- Management of entities, such as the default and available languages, currencies, etc.
- Management of entity sites, since an entity may have multiple sites
- Management of user roles, such as admins, managers and regular users
- Management of authentication methods per entity
- Management of voting methods per entity

As stated in the architecture overview, this component interacts with all components, providing necessary information and authorization for the communication between components and authorization of the users when performing protected actions.

4.2.12 Questionnaire

The Questionnaire (Q) component allows the creation of dynamic questionnaires with different kinds of questions/answers while also presenting them and storing the user answers.

In order to provide this, it has the following functionalities:

- Create, update and delete (CRUD) questionnaires, question groups, questions and answers
- Management of question groups, such as the order on which they appear
- Management of questions, such as their position inside a question group

This component allows to create public questionnaires, which are open to anyone without authentication, and questionnaires that require authentication (both anonymously or not).

4.2.13 Vote

The Vote component provides the means to store all kinds of votes, whether they are likes on forums' posts or votes on proposals which are only available through certain configurable periods and from certain users.

To accomplish this, the Vote component provides the following functionalities:

- Create, update and delete (CRUD) vote method groups, vote methods, voting events, voting configurations and votes
- Management of vote method groups
- Management of vote methods
- Management of vote events and configurations
- Votes validation and storage
- Storage of metadata associated with each vote
- Support anonymous voting in specific proposals and ideas

4.2.14 WUI

The Web User Interface (WUI) is the frontend of the EMPATIA Platform providing the interfaces to interact with all the backend components and perform all support actions.

The interface is divided in two main areas, public and private, being the private area only accessed by managers and administrator.

The administrators functionalities are:

- Entities management, which includes the entities CRUD and configuration, such as the entities languages and its moderators
- Languages management
- Currencies management
- Countries management
- Time zones management
- Authentication methods management
- Voting methods management
- Sites management
- Manage user profile

These functionalities are related with the platform configuration and base information, which can then be selected and used by the managers in their entities.

Managers are able to access the following features:

- Users management
- Contents management (which include pages, news, menus, etc.)
- Forums, discussions and ideas management (all CB types)
- Events management
- Questionnaires management
- Kiosks management (cf. Section 4.5)
- Manage user profile

A user, on the public side, can access the following features:

- Authentication
- Public/authenticated contents (menus, pages, news, etc.)
- Forums, topics, ideas, etc.
- Be a moderator of CBs
- Post, comment, like and vote in authorized CBs
- Manage user profile

4.2.15 Kiosks

The Kiosks component is specifically designed to manage the hardware component of kiosks, as such the functionalities are as follows:

- Send data through socket.io communication
- Support several types of authentication, including RFID
- Support authentication and authorization policies (i.e. force users to logout)
- Support vote in sessions and other events
- Show/select proposals, ideas
- Show multiple topics at the same time
- Support multiple hardware types of kiosks (i.e. with and without screen)
- Support multiple languages

4.2.16 Open Data

The Open Data component is present in the architecture to manage all the processes of import, export data to and from other platforms. In addition, this component also assures the open access to the data that is generated by the platform and that has is compliant with the ethical policies of the Data Management Plan.

The Open Data functionalities are as follows:

- Support import and export of data to CKAN
- Interoperability with CKAN APIs
- Support high availability of data, no matter the volume or the number of users requesting it
- Support correlation of data, by supporting dataset relationships

4.2.17 2nd PB Cycle

The 2nd PB cycle component has the following functionalities:

- Monitoring of project progress
- Monitoring of ideas, events
- Manage community building (CRUD)
- Support evaluation (surverys, questionnaires)

4.3 Technologies, tools and protocols

The platform runs on a Linux environment, Debian [49], with all its components being develop in PHP [50] and using the Laravel framework [51]. To store the platform data, two different database technologies are used, MySQL [52] which is a Structured Query Language (SQL) relational database where all the persistent data is stored, and Redis [53], which is an in-memory data structure store – stores the data in a key/value approach – that is used to store temporary data, such as Sessions and Cache, and also used as a message broker to provide push notifications when paired with a Socket.io [54] server.

All backend components implement REpresentational State Transfer (REST) APIs accepting and providing all data in the JSON format. The APIs implement security mechanisms in order to communicate between them, and to be accessed externally (access token, which ensures that all direct access to the backend modules is controlled).

The users' authentication and validation is performed on every request, and the platform uses JSON Web Tokens (JWT) [55] and supports OAuth2 [56] server and client.

The EMPATIA Platform is cloud-ready and can be deployed bare metal, using Docker containers [57] or virtual machines (VMWare/VirtualBox appliance). Dockers containers will be available at Docker hub under the EMPATIA organization [58].

To keep version control on the components development and to allow the maintenance of entity specific versions, GIT [59] repositories are used. All EMPATIA stable open source code will be release on GitHub [60].

4.4 Deployment options

The EMPATIA Platform offers different ways to be contracted and deployed. The platform can be contracted as a service or deployed either as an all-in-one solution or standalone components.

As EMPATIAaaS, the platform supports the management of different entities and participation processes (including different sites, configurations, currencies, etc.). EMPATIA project will host an EMPATIAaaS platform to allow public access to EMPATIA Platform.

When deploying the platform, it is possible to select between an all-in-one integrated deployment or deploy all components standalone. These versions can be deployed in privately owned infrastructure or cloud providers.

The all-in-one solution (all components in one server) is aimed to be used only for testing purposes or very small usage. The platform can be obtained in a pre-configured Docker container or Virtual Machine image (deployment in less than 2 minutes).

In the standalone approach, each component can be obtained as a pre-configured Docker container, Virtual Machine image, or source code. Load balancing and complex databases deployments are not provided by EMPATIA.

To deploy a production environment, the minimum infrastructures requirements are:

- 1 Linux (Debian) server per component, ~12 total
- 1 MySQL database
- 1 Redis database

In this minimum requirement, both database solutions can be located in the same server, but should be a different one from the components. This deployment can be made in around 2 hours. To deploy a full scalable and fault tolerant environment, the minimum requirements are:

- 2 different geographic placements – different power and Internet connectivity supplies
- Components redundancy – at least 2 servers per component
- Multiple load balancers with redundancy – at least 2 load balancers per component
- MySQL databases with redundancy and replication
- Redis databases with redundancy and replication

For this kind of production environment, the deployment will take 2 or more days, depending on the number of desired redundancies to have.

4.5 Hardware/Kiosks

Although the EMPATIA Platform’s main aim is web and mobile interaction, the need was identified to test and validate solutions to improve the quality and reduce the fraud in face-to-face assemblies, mainly in vote processes. Taking into consideration the low level of ICT experience and knowhow of many of the EMPATIA Platform final users, a set of interactive Kiosks have been identified. At least two voting kiosks approaches will be designed and developed: one Voting Kiosk that connects to an HDMI monitor/TV that allows citizens to read proposals and vote on them; and a Ballot Kiosk that will include a traditional printed ballot alike interface, plus buttons to record the voting preferences and a small touch screen display with status information. Both prototypes use RFID based authentication, and the support of additional authentication methods is assured with a dedicated component (i.e. Kiosks).

These kiosks will mainly be used in the EMPAVILLE role-playing game, for prototype and validation purposes. Since no similar equipment was found in the SoA analysis, and the use cases do not detail this aspect of participation/voting, the developed prototypes are being used to identify the requirements for these types of kiosks in order to define a set of guidelines and good practices on these type of solutions.

5 APIs specification

One of the most important EMPATIA architecture aspects are the public APIs. These APIs will allow the interaction between external tools/platforms and EMPATIA. In the next subsections are presented the APIs overviews, organized per component. Due to the complexity and extent of the full documentation, this is available in Annex B, following the Open APIs Initiative standards [64].

As presented before, these components API are implemented using REST/JSON interfaces and follow the current state of the art guidelines (e.g. Facebook, Twitter, Google). Due to the rapid changes and new standards on REST/JSON API specification, all EMPATIA API may change (e.g. RAML [61], API Blueprint [63], WADL [62], OAI [64] or JSON API [65]).

5.1 Analytics

The main functionalities of the Analytics include the Vote Event and Daily Vote Event and are summarized in Table 49.

Table 49 – API information of Analytics

Vote Event		Daily Vote Event
• Information of Votes • Total Votes By Gender • First Votes By Gender • Second Votes By Gender • Verify Votes By Profession • Count Votes By Profession • First Vote By Profession • Second Vote By Profession • Votes By Channel • Count Votes By Channel • First Vote By Channel	• Second Vote By Channel • Votes By Neighborhood • Count Votes By Neighborhood • First Vote By Neighborhood • Second Vote By Neighborhood • Count Votes By Age • First Vote By Age • Second Vote By Age • Votes by Town • Votes by Education • Customize Dashboards • Elaboration of reports	• Daily Vote event

5.2 Authentication

The main functionalities of the Authentication are summarized in Table 50.

Table 50 – API information of Authentication

Vote Event	
• Authenticate • Authenticate RFID • Authenticate Alphanumeric Code • Logout User • Validate User • Confirm User Email	• Get User (retrieve user details) • List Users • List specific Users • List User Names • Create User • Update User • Delete User

5.3 PAD

The main functionalities of the PAD component are summarized in Table 51.

Table 51 – API information of PAD

PAD	Topic	Posts
List PADs Show PAD Create PAD Update PAD Delete PAD List specific PADs Statistics Change Status PAD Blocks a PAD Unblocks a PAD Set configurations Show Configurations List Moderators Check Moderator Add Moderators Delete Moderator Get Topics List with First Post Get Topics List with Last Post Add Vote List Votes Delete Vote Set parameters List parameters Delete Parameter List Abuses List CB Abuses List CB Parameters Options Access Control	List Topics Show Topic Create Topic Update Topic Delete Topic Request first post Statistics Change Topic status Block Topic Unblock Topic List Posts Set parameters List parameters Show PAD First Post Show Topic Data Show Topic child topics with all their attributes List Topic Moderators List Topic Abuses Version Control	List Posts Show Topic Create Post Update Post Delete Post Change Post Status Get Post History Get Post Files Get Post File details Add File to Post Update Post File Delete Post File Change Post File Position List Post Abuses Show All User Posts Version Control
Parameter	**ParameterType**	**Post Abuse**
List Parameters List Parameters with Options Show Parameter Create Parameter Update Parameter Delete Parameter	List ParameterTypes Show ParameterType Store ParameterType Update ParameterType Delete ParameterType	List Post Abuses List Post Abuses by PAD List Post Abuses by Topic Show Post Abuse

• Show Parameter with Options • Add Option to Parameter • Add Multiple Options to Parameter • Edit Option of Parameter • Remove Option from Parameter		• Create Post Abuse • Update Post Abuse • Delete Post Abuse
PAD Moderators	**Topic Moderators**	**Configurations**
• List PAD Moderators • Show PAD Moderator • Create PAD Moderator • Update PAD • Delete PAD Moderator	• List Topic Moderators • Show Topic Moderator • Create Topic Moderator • Update Topic • Delete Topic Moderator	• List Configurations • Show Configuration • Create Configuration • Update Configuration • Delete Configuration • Add Option to Configuration • Delete Configuration Option

5.4 Content Management

The main functionalities of the Content Management (CM) are summarized in Table 52.

Table 52 – API information of Content Management

Content	Menu
• List Contents • Show Content • Create Content • Update Content • Delete Content • Attach a File to Content in storage • Update file details • Delete the attached file • Sets a new position for the attached file in a list • Show file details • Get files by type and content • Show versions • Show version • Enable version • Publish a new version • Removes a published version • Version Control for news and pages	• List Menus • List Menus by Access Id • Show Menu • Create Menu • Update Menu • Reorder Menu Item • Delete Menu • Get a Menu Tree • Version Control

5.5 Events

The main functionalities of the Events are summarized in Table 53.

Table 53 – API information of Events

Events	Sessions	Speakers
• List Events • Show Event • Create Event • Update Event • Delete Event • Show Sessions • Show Sponsors • Show Registrations • Export to calendar • Support download to PDF	• List Sessions • Show Session • Create Session • Update Session • Delete Session • Show Speakers • Show Registrations	• List Speakers • Show Speaker • Create Speaker • Update Speaker • Delete Speaker
Sponsors	**Items**	
• List Sponsors • Show Sponsor • Create Sponsor • Update Sponsor • Delete Sponsor	• List Items • Show Item • Create Item • Update Item • Delete Item	

5.6 Files

The main functionalities of the Files are summarized in Table 54.

Table 54 – API information of Files

Files	
• List Files • Show Files • Create Files • Download File	• File Usage and access rights • Search Files • Index Images • Upload File

5.7 Notify

The main functionalities of the Notify are summarized in Table 55.

Table 55 – API information of Notify

Emails	Email Templates
• Send Email	• List Email Templates • Show Email Template • Show Email Template with all translations • Create Email Template

	• Edit Email Template • Delete Email Template
General Notifications	
• Time notifications (when deadline is approaching) • User subscribed notifications	

5.8 Orchestrator

The main functionalities of the Orchestrator are summarized in Table 56.

Table 56 – API information of Orchestrator

User	Authentication	Entity
• List User roles • List Users • Show User • Create User • Update User • Delete User	• Verify Authentication • Verify role	• List Entities • Show Entity • Create Entity • Update Entity • Delete Entity • Add Language • Default Language • Remove Language
Currency	**Country**	**Language**
• List Currencies • Show Currency • Create Currency • Update Currency • Delete Currency	• List Countries • Show Country • Create Country • Update Country • Delete Country	• List Languages • List All Languages • Show Language • Create Language • Update Language • Delete Language
Timezone	**Vote Method**	**Authentication Method**
• List Timezones • Show Timezone • Create Timezone • Update Timezone • Delete Timezone	• List Vote Methods • Show Vote Method • Create Vote Method • Update Vote Method • Delete Vote Method	• List Authentication Methods • Show Authentication Method • Create Authentication Method • Update Authentication Method • Delete Authentication Method
Forum	**Discussion**	**Access Types**
• List Forums • Show Forum • Create Forum	• List Discussions • Show Discussion • Create Discussion	• List Access Types

• Update Forum • Delete Forum	• Update Discussion • Delete Discussion	
Access Menu	**Pages**	**Geographic Area**
• List Access Menu • Active Access Menu • Show Access Menu • Create Access Menu • Update Access Menu • Delete Access Menu • Activate Access Menu	• List Page • Show Page • Create Page • Update Page • Delete Page	• List Geographic Areas • Show Geographic Areas • Create Geographic Area • Update Geographic Area • Delete Geographic Area
Idea	**Category**	**Module**
• List Ideas • Show Idea • Create Idea • Update Idea • Delete Idea • Show Categories in Idea • Create Categories in Idea • Update Categories in Idea	• List Categories • Show Category • Create Category • Update Category • Delete Category • Create Categories	• List Modules • Show Module • Register Module • Check Token
Kiosk	**Kiosk Type**	**Site**
• List Kiosk • Show Kiosk • Create Kiosk • Update Kiosk • Delete Kiosk • List Proposals • Create Proposal • Delete Proposal • Reorder Proposals • Support Multilanguage • Show proposal	• List Kiosk Types • Show Kiosk Type	• List Sites • Show Site • Create Site • Update Site • Delete Site • Show Site Entity

Parameter	**ParameterType**	**Module**
• List Parameters • Store Parameter • Update Parameter • Delete Parameter • Add Option to Parameter • Add Multiple Options to Parameter • Edit Option of Parameter	• List ParameterTypes • Show ParameterType • Store ParameterType • Update ParameterType • Delete ParameterType	• List Modules • Show Module • Register Module • Check Token

• Remove Option from Parameter		

5.9 Questionnaire

The main functionalities of the Questionnaire are summarized in Table 57.

Table 57 – API information of Questionnaire

Forms	Question Groups	Questions
• List forms • Show Form • Create Form • Update Form • Delete Form • List of Question Groups of the Form • List all information about Form • List all replies of a Form	• List Question Groups • Show Question Group • Create Question Group • Update Question Group • Delete Question Group • List Questions of a Question Group • List Questions Options of a Question Group • Update position of a Question Group	• List Questions • Show Question • Create Question • Update Question • Delete Question • List Options of a Question • Update position of a Question
Question Options	**Question Types**	
• List Question Options • Show Question Options • Create Question Option • Update Question Option • Delete Question Option	• List Question Types • Show Question Type • Create Question Type • Update Question Type • Delete Question Type	

5.10 Vote

The main functionalities of the Vote are summarized in Table 57.

Table 58 – API information of Vote

Method Groups	Method	Configuration
• List Method Groups • Show Method Group • Create Method Group • Update Method Group • Delete Method Group	• List Method • Show Method • Create Method • Update Method • Delete Method	• List Configurations • Show Configuration • Create Configuration • Update Configuration • Delete Configuration
Events	**Vote**	
• List Events • Show Event	• Show Vote • Create Vote	

• Create Event • Update Event • Delete Event • Create Event Configuration • Delete Event Configuration • Show Events • Check Event Open	• Show All Votes • List Votes • Show Votes Details by Vote Key • Show User Votes

5.11 WUI

The main functionalities of the WUI are summarized in Table 59.

Table 59 – API information of WUI

Components	Interface
• Route information to specific component • Activate/Deactivate usage of specific component	• Implement interface of specific component • Support dynamic layouts • Support dynamic pages (e.g. work in mobile and web browsers)

5.12 Kiosks

The main functionalities of the WUI are summarized in Table 60.

Table 60 – API information of Kiosks

HW Kioks	General
• Support multiple types of kiosks • Support vote • Show proposals • Show multiple topics	• Support multi language • Support multiple authentication methods • Support authentication and authorization policies

5.13 Open Data

The main functionalities of Open Data include management of data access using CKAN APIs.

5.14 2nd PB Cycle

The main functionalities of 2nd PB Cycle are documented in Table 61.

Table 61 – API information of 2nd PB Cycle

2nd PB Cycle
• Monitoring of projects, proposals, ideas • Evaluation of projects, proposals. ideas

5.15 Lime Survey

The main functionalities of Lime Survey include management of questionnaires using Lime survey library.

6 EMPATIA Platform development

This section details the development strategy followed to develop the EMPATIA Platform. The platform development uses an agile methodology based on a SCRUM approach. The usage of this methodology allows to quickly adapt to new requirements and deadlines (e.g. from project pilots, other pilots, the EMPAVILLE role-playing game and other external requests).

With the high demand to use the EMPATIA Platform with users in the games and pilots, the quality of the intermediate versions for the platform needs to be ready for public usage. Thus, a set of testing environments were design to validate the platform. The first step of the testing is the internal testing based on Unit Tests, followed by functional tests. Afterwards, the platform is deployed in limited public environments like the EMPATIA website (validated by EMPATIA partners and visitors) and then the EMPAVILLE role-playing game (validated by the game players in dedicated sessions). Finally, the platform is deployed in pilots (both EMPATIA pilots and other external pilots) with pre-production testing by the pilots' entity (and responsible pilot EMPATIA partner).

6.1 SCRUM agile methodology

The SCRUM agile methodology for the EMPATIA Platform development includes different teams with respective roles, as summarized in Table 62.

Table 62 – Participatory budget platforms

Team / Person	Description of Role	Partner
Product Owner	• Define release dates • Organize Milestones review • Negotiate Priorities • Demonstrate platform to key stakeholders • Ensure that product backlog is clear	Technical Project Coordinator
Scrum Master	• Help product owner, in the design of the solution and technical choices • Help product owner in keeping a good understanding of the product backlog • Help integrating feedback from key stakeholders • Coaching the development and integration teams • Facilitating team events to ensure regular process	ONE (WP2 Leader)
Development and Integration Teams	• Implement functionalities, • Test & Fix implemented functionalities • Plans required work to achieve sprint goals	ONE, D21, UNIMI

The workflow included in the scrum methodology is illustrated in Figure 4, highlighting, the diverse actors.

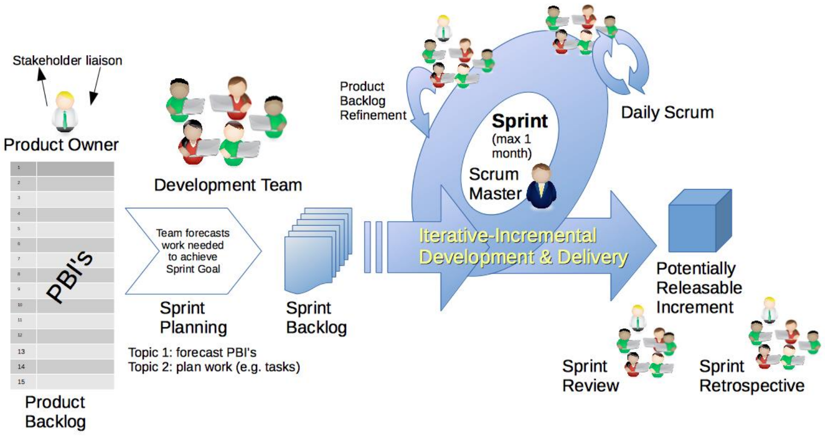


Figure 4 – Agile methodology

The workflow is triggered by a product backlog that aims to specify the functional and non-functional requirements of the platform. The product backlog also identifies the priority of the diverse functionalities for different (planned) sprints, which lead to potential releases of the platform.

The workflow includes diverse sprints and releases. The sprints have associated a duration which can go from two weeks to a maximum of one month. Releases occur according to the planning of the product owner and when the following conditions are met:

- Software has been integrated and fully tested, according the functional testing for the functionalities specified in the product log
- Software for key stakeholders has been documented
- Software has been deployed, according to the deployment model planned by the product owner

In the planning phase there is communication between the product owner and the remaining actors, including the selection of functionalities that will be implemented in a given sprint. This phase aims mainly to specify the sprint backlog to detail the work needed to finished the selected product backlog items with the entire team.

During the incremental development & delivery phase all the development and integration activities are performed by the development & integration teams and monitored by the scrum master. This phase also includes daily scrum, which have a limited duration and requires the participation of all the members of the development teams. All the testing & fixing is performed in this phase, according to the defined testing methodology, as detailed in the next subsections.

The sprint review aims to review the work that was completed and the planned work that was not accomplished. The sprint retrospective considers the previous sprint(s) in order to identify the continuous improvement actions.

6.2 Unit tests

The unit tests have been designed and implemented for each developed component in order to ensure that tests can be carried in a controlled way and are not extensively error prone due to missing functionalities (i.e. not yet developed). Unit tests are mainly conveyed by the development team ensuring that the promised features are delivered and at the quality required for the functional testing and release phases.

There is no limitation in number of unit tests per component or size of a unit test. Indeed, the tests are expected to cover the functionalities with must priority levels, so that the quality level required for functional testing and release phases. In this sense, each component includes unit tests for these functionalities with higher priority levels. Such tests are performed with a test program that is implemented to test such functionalities. For instance, in the authentication component a Unit Test includes testing successful logins (i.e. valid user credentials) while another test includes invalid logins (i.e. incorrect user credentials). In other words, the unit test describes the complete behaviour of the component, and the test program aims to test such behaviour/functionalities in certain conditions.

6.3 Functional tests

The development team has to develop the test cases that will have to be executed against the component they are integrating with. The test cases developed by each team for its own component are designed to test the component itself against another fully functional component.

The functional test cases should settle in a clear and crisp manner the dependencies parameters based on the dependency characterization parameters defined for each component. For instance, to test the Community Building component, the authentication component works as a dependency, since the user needs to the authenticated to post in forums or to comment ideas or proposals. A functional test case aims to assess that the functionalities designed for a specific component are well implemented and are provided according the expected behaviour of a component. To be complete, a functional test case assesses the behaviour of a component and its associated APIs, which allow the integration of other component or the interface with external components of the EMPATIA architecture.

The results are meant to decide if the component is functioning correctly on the implemented APIs and that the two components that have to be integrated are fit to be used together (e.g. Authenticaiton + Community Building). These tests will also be useful for later, in case internal modifications (i.e. fixes) have to be made to any of the components, in order to determine if the overall functionality has been altered or not.

The functional test cases, assume that the unit tests for each component are all successful, which means that the developed functionalities inside each component are provided as expected.

The functional tests have different steps, which are detailed in the following subsections.

6.4 Testing deployments

This subsection exemplifies the tests for the diverse deployments of the EMPATIA Platform performed till the date of writing this deliverable. Such deployments include the EMPATIA website and the EMPAVILLE role-playing game.

Both of these deployments share the same testbed, which is described in Table 63.

Table 63 – Environment to test EMPATIA Platform deployments

Type	Description
OS	Linux (Debian)
Language	PHP
Framework	Laravel

Database	MySQL (SQL) Redis (no SQL)
File System	EXT4 and Hadoop
APIs	REST/JSON
Authentication	JWT Token

6.4.1 EMPATIA website

At the time of writing this deliverable the EMPATIA website powered by the EMPATIA Platform is still under development. This section provides the testing details of the EMPATIA Platform using the EMPATIA website.

As presented in the previous subsection (cf. Section 6.1), the SCRUM agile methodology has been employed to develop the EMPATIA website. All the design of the website has been documented in Deliverable D5.1. The coding by the development team has been performed using the tools and frameworks detailed in Table 63, namely using the Laravel, that is a PHP framework that supports scalable and diverse deployment models, as envisioned by the EMPATIA architecture.

The EMPATIA website has been employed to test and enhance the components of the EMPATIA Platform, namely in the following aspects:

1. Testing the northbound APIs of components, APIs that enable access to the functionalities of the component
2. Testing southbound APIs of components, which allow the persistent storage of data
3. Testing east and west bound APIs of components, which are used to store files in the file system, when applicable and to manage session related information of the component
4. Testing the functionalities of the component with higher priority
5. Testing the integration of components for pre-release

The testing of APIs and internal functionalities follows the unit testing and functional testing as specified in section 6.2 and section 6.3, respectively. One of the components, worth mentioning in the testing activities includes the Web User Interface, as this component receives all the requests and interacts with the remaining components through their respective northbound APIs. The testing of the WUI involved test programs, in the unit testing, but also users from the integration team, assessing how the components behaved regarding the supported functionalities.

6.4.2 EMPAVILLE

EMPAVILLE as the role-playing game has been an excellent way of testing the EMPATIA Platform. In fact, the components of the EMPAVILLE platform did not need to be modified to support EMPAVILLE. The modifications relied mainly in terms of design and layout to accommodate a dedicated graphical design and additional features like a demonstration presentation that is used during the game to explaining the rules and results of EMPAVILLE.

EMPAVILLE also required other type of devices and interfaces – the Kiosks to allow an easy voting in proposals without requiring a computer for the participation of citizens (cf. Section 4.5).

Up to now, EMPAVILLE has been demonstrated and used in three events:

- Boston, in the participatory budgeting project conference, 20 of May 2016, with around 40 users providing feedback on the questionnaires
- London in PB workshop, 2 of June 2016, with around 32 users providing feedback on the questionnaires
- Aquila, Italy, in the "Participation Festival: Wake Up Italy", 9 of July 2016, with more than 30 participants

In each event, a specific questionnaire has been made available and answers gathered with feedback (cf. Annex A). The outputs of these questionnaires are being used to enhance the EMPATIA Platform (including the kiosks prototypes). Changes implemented based on EMPAVILLE feedback have successfully improved the following sessions (e.g. minor, but of upmost importance layout changes, kiosks feedback/information/help).

The replies to questions like "How could we improve EMPAVILLE experience" and "Has Today's event given you any ideas on how to improve a PB process" are currently being analysed in order to identify the most demanded functionalities required by end-users.

6.4.3 Pilots

At the time of writing this deliverable, the planned pilots have not started.. EMPATIA will use these pilots to test and validate the EMPATIA Platform in different countries, different Participatory Budgeting (PB) models and different users' profiles.

Meanwhile, in the Lisbon pilot several demonstrations have been performed to the municipality staff (including the vice-Mayer responsible for the Lisbon PB implementation) and are now starting a pre-pilot test phase. A first workshop was held in Wuppertal in Germany to gather initial feedback regarding desired features and their prioritization.

These preliminary testing/demonstrations have been fruitful in collecting valuable feedback. Table 64 summarizes the input received from the Lisbon municipality. It should be noticed that Lisbon has an active platform for participatory budget, but is not complete, and the provided feedback, serving as input for the EMPATIA Platform is based on the experience accumulated with the running platform and the first contacts of the pre-releases of EMPATIA Platform.

Table 64 – Feedback from Lisbon Municipality

Component	Feedback provided	Input to EMPATIA
Mutichannel Participations	• Enable project proposals • Support the technical analysis of proposals • Allow selection of proposals by voting • Monitor the progress of chosen project	The required functionalities are currently supported by EMPATIA.
Community Building	• Support comments by citizens in public consultation projects	EMPATIA Platform can support the public consultation projects. The

	(mandatory by law) and thematic consultations • Allow entity to specify public consultation projects • Public consultation projects are confined to geographic areas. • Supports ideas for the Lisbon Municipality, with analytics, regarding the number of likes. • The registration of ideas can be performed in different georeferenced means, video, images, text, documents.	information related with public consultation projects must include, description, the consultation period, the thematic area, the typology (evaluation of impact in the ambient, prevention of accidents), participation forms. The submission of ideas is already supported in EMPATIA, nonetheless, the information regarding the analytics related to ideas will be included in the next iteration of the EMPATIA Platform
External components (Na Minha rua)	• Allow citizens to request green spaces in their street • Allow citizens to request maintenance of equipment/building in municipal habitations • Allow the request for public illumination • Allow the request for maintenance and repair of public roads	Integrate EMPATIA with the external module (na Minha Rua) that supports already a number of features. This integration aims mainly to facilitate the prioritization of the several requests, by analysing the number of likes, or votes in a certain request.
Community Building	• Component to support surveys in order to promote the debate of themes identified as pertinent by the Municipality • Allow the citizen to comment, vote on such proposals	Integrate support in EMPATIA for surveys, to straight the connection between the municipality and its citizens, by providing opinions in early stage of projects.

As the example of Lisbon municipality the pilots will allow to collect feedback to improve EMPATIA Platform in the following iterations. In particular, such feedback will be important to identify external components that can enhance the functionalities of the EMPATIA Platform.

7 EMPATIA Platform Integration

The EMPATIA Platform has been designed and developed to promote the integration with other platforms and to allow the export of data.

The first integration phase of EMPATIA is to integrate existing platforms and tools with the EMPATIA Platform features. The main platforms analysed and integrated are open data, croudfunding, surveys and logging. In the next subsections, we describe the selected platforms to integrated with EMPATIA, some were selected due to pilot requirements, others based on state-of-the-art analysis.

The second integration phase corresponds to the integration of EMPATIA into other tools. Examples of tools being analysed are Content Management Systems (CMS), such as Drupal [68], and specific file formats and standards to be compliant and interoperable with specific applications, such as XML, CSV, etc.

7.1 CKAN

CKAN [4] is considered as one of the leading open-source data portal platforms. The integration with this platform is being performed through the available APIs, and SDK that is available in github [69].

7.2 BoaBoa

The EMPATIA Platform is being integrated with crowd-sourcing platforms due to the needs of specific municipalities. The Lisbon Municipality already supports the BoaBoa [70] crowdfunding platform to support campaigns developed by the community seeking for funding to implement the projects. Other crowdfunding platforms are being evaluated to be integrated.

7.3 Lime Survey

The Lime survey library is used to develop enriched questionnaires, that go beyond the functionalities that are already supported by Questionnaires component. The integration with this platform allows to reach a wider public, and to foster the participation of citizens.

7.4 Graylog

Graylog is employed mainly to collect, analyse performance measurement data of the platform. For such, the components of the platform provide data to graylog, which can be used and correlated with the data that is collected in the system where the components run.

8 Implementation plans

This section details the final implementation plans including the next steps (roadmap) for the implementation of EMPATIA Platform.

The implementation plans include two releases of the platform: P1 – includes the first version (v1) of components, which correspond to basic functionalities; and P2 – which includes the final version (v2) of components. Table 65 identifies the planned functionalities for each component per each platform release.

Table 65 – Implementation plans for the platform releases

Component	Partner	P1	P2
Analytics	ONE	v1 • *Aggregate, transform collected data,* • *Support streaming of processed data to other components*	v2 • *Support customizable graphics,* • *Support dashboards and advanced rules for presenting and exporting data*
Auth	ONE	v1 • *JWT validation* • *Registration of users* • *Profile management* • *Basic authentication*	v2 • *Support authentication using identity cards* • *Support granular authorization mechanisms* • *Enable accounting of users operations*
PAD	ONE	v1 • *Management of PADs* • *Management of topics* • *Posts management* • *Moderation* • *Comments*	v2 • *Integration with social networks*
CM	ONE	v1 • *Menus management* • *Content management*	N/A
Events	ONE	v1 • *Events management* • *Sponsors management* • *Sessions management* • *Payments management*	N/A
Files	ONE	v1 • *Upload* • *Download* • *Support for diverse type of files*	N/A

Monitoring	ONE	v1 • *Support of Graylog2*	v1 • *Export of data*
Design	ONE	v1 • *MP management*	V2 • *MP management*
Notify	ONE	v1 • *Email management*	v2 • *SMS management* • *Online management*
Orchestrator	ONE	v1 • *Manage components* • *Entities management* • *Sites management* • *User roles management* • *Authentication methods per entity* • *Voting methods per entity*	v2 • *Support components of v2*
Q	ONE	v1 • *Questionnaires management*	N/A
Vote	ONE	v1 • *Vote methods management (e.g. kiosks)* • *Vote events management* • *Polls management*	v2 • *Support analysis of voting*
WUI	ONE	v1 • *Manage private area of the platform to manage the diverse components (entities, languages, currencies, time zones, etc)* • *Manage public area of the platform*	v2 • *Management of new models introduced in final version*
Projects (2nd PB Cycle)	UNIMI and ONE	v1 • *Projects Management*	v2 • *Projects Management*
OpenData	ONE	N/A	v1 • *Data Export tools management*
CB	ONE	N/A	v1 • *Info community* • *Management of Members*
Kiosks	ONE	Not implemented as a component, developed in Hardware	v1 • *Support Vote* • *Support Multilanguage*

			- *Support multiple authentication methods*

8.1 Implementation status

This subsection provides the status of the implementation of the diverse components of the EMPATIA Platform. A detailed overview of the diverse functionalities is provided per each component in section 8.2 along with the respective roadmap.

Table 66 summarizes the status and the partner responsibility of each component. As identified, some of the components will be enhanced in the second version of the platform implementation.

Table 66 – Implementation status of the diverse components

Component	Partner	Status
Analytics	ONE	First version implemented.
Auth	ONE	First version implemented.
PAD	ONE	First version implemented.
CM	ONE	First version implemented.
Events	ONE	First version implemented.
Files	ONE	First version implemented.
Monitoring	ONE	First version implemented.
Design	ONE	Initial functionalities implemented, under development
Notify	ONE	First version implemented with email support.
Orchestrator	ONE	First version implemented.
Q	ONE	First version implemented.
Vote	ONE	First version implemented with likes, multiple and negative voting methods support.
WUI	ONE	First version implemented with support of all implemented components features.
Projects	ONE & UNIMI	Not implemented.
OpenData	ONE	Not implemented.

8.2 Roadmap

The roadmap targeting the second version of the EMPATIA Platform is depicted in Figure 5 (some partners' responsibilities are to be confirmed). The first version of the platform is released at M10, while the final version is to be released at M22.

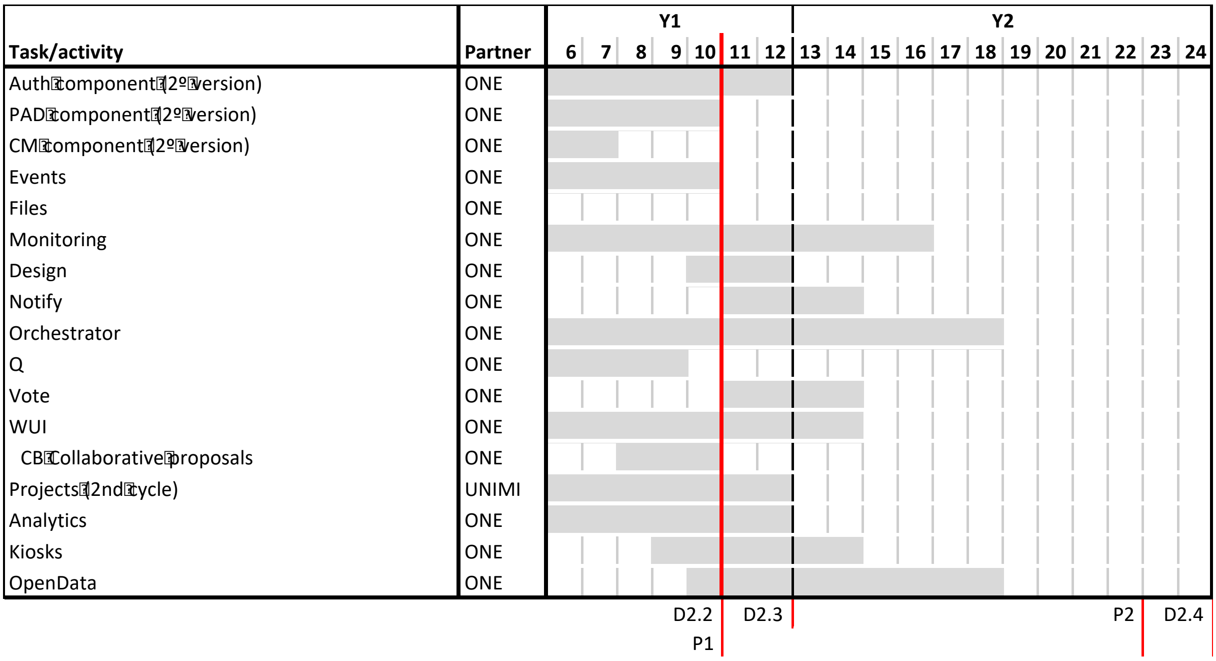


Figure 5 – Roadmap for second version of EMPATIA Platform

The second version of the platform (P2) will include enhanced and complete full set of functionalities in the EMPATIA components implemented in the first version of the platform.

The following subsections detail the roadmap per each component and detail the functionalities that have already been implemented.

8.2.1 Analytics

The roadmap for the Analytics component is summarized in Table 67. In this table **IMP** corresponds to functionalities that have already been implemented, and MXX represents the EMPATIA project month where the functionality is expected to be made available.

Table 67 – Roadmap of Analytics Component

Functionality	Month	Functionality	Month
Information of Votes	IMP	Count Votes by Age	IMP
Information of Votes		First Votes by Age	
Total Votes by gender		Second Votes by Age	
First Votes by gender		Statistics by Date	
Second Votes by gender		Statistics by Town	
Verify Votes by profession		Statistics by Education	
Count Votes by profession		Proposal Analysis	M22
First Votes by profession		Ideas Analysis	M22
Second Votes by profession		Issues Analysis	M22
Verify Votes by Channel		Participation Analysis	M22
Count Votes by Channel		Customization of dashboards by entity	M22
First Votes by Channel		Support elaboration of reports	M22

Second Votes by Channel	IMP	First Votes by Neighborhood	IMP
Verify Votes by Neighborhood		Second Votes by Neighborhood	
Count Votes by Neighborhood			

8.2.2 Authentication

The roadmap of the Authentication component is summarized in Table 68.

Table 68 – Roadmap of Authentication Component

Functionality	Month	Functionality	Month
Management of Users	IMP	Login with Alphanumeric Code via SMS	M14
Authenticate with Credentials		Login with Identity Card	M14
Authenticate with QRCode		Login with Social Media (Twitter)	M14
Authenticate with RFID		Login with Social Media (Google+)	M14
Authenticate with Alphanumeric code		Login with passport	M14
Login with Facebook		Login with resident card	M14
Retrieve users' details		Support dual-factor authentication	M14
Expiration mechanisms of authentication (logout)		Support policies to renew credentials	M14
Open standards for authorization & Authentication		Support Single-Sign-On	M14
Authorization policies per user role		Support one-time generated password (OATH support)	M14
		Require Triple-factor authentication for Administrator and Super manager roles	M14

8.2.3 PAD

The roadmap of the PAD component is summarized in Table 69.

Table 69 – Roadmap of PAD Component

Functionality	Month	Functionality	Month
Management of Comments	IMP	Management of PADs	IMP
Block Posts		Access Control	
Management of Post Abuses		Manage Topics Cooperators	
List Post Abuse by Topic		Block/Unblock PADs	
List Post Abuse by PAD		Manage Community Building and associated parameters	
Management of Moderators		Version Control Posts	
List of PAD Moderators		Advanced search mechanisms	M16
Management of Votes Sessions		Cooperation between ideas and proposals	M16
Management of Vote Methods		Collaborative proposals	M16

8.2.4 Content Management

The roadmap of Content Management (CM) component is summarized in Table 70.

Table 70 – Roadmap of Content Management Component

Functionality	Month	Functionality	Month
Management of News	IMP	Order Files in lists	IMP
Get List of News		Show version of pages	
Get actual news		Get list of page versions	
Get Last News		Video in MPEG4 format	
Management of Events		Video in webM format	

Get Last Events		Video in Ogg format	
Management of Menus		Content in HTML5	
Reorder Menus		Support WYSIWYG editor	
Management of Text Content		Publish/Unpublish Content	
Add Files		Access Control	
Management of Content		Version control for pages, news and events	
Get First Files		Reports	M12
Management of File's Details		Export of information	M12
Image in jpeg format		Support online edition of documents	M12
Image in png format		Support Share	M12
Image in gif format		Version Control for menus	M12
Image in pdf format			

8.2.5 Design

The roadmap of the Design component is summarized in Table 71.

Table 71 – Functionalities of Design Component

Functionality	Month	Functionality	Month
Definition of dates for participation model	IMP	Technical analysis of proposals	M18
Graphical specification of steps for participation model (i.e. diagrams)		Define participation rules	M18
Detailed specification of steps for participation model		Define access control policies	M18
Support of Design Template		Define channels for participation	M18
Definition of participation model	M18	Specification of fraud detection	M18
Support MP Template	M18	Cooperation between ideas and proposals	M18

8.2.6 Events

The roadmap of the Events component is summarized in Table 72.

Table 72 – Roadmap of Events Component

Functionality	Month	Functionality	Month
Management of Events	IMP	Verification of Registrations	IMP
Management of Sessions		Management of Speakers	
List Sessions by Event		List Speakers by Sessions	
Management of Sponsors		Management of Events Items (e.g. keynotes, presentations)	
Management of Payments		Export to calendar (ics)	M12
Management of Registrations		Support download to pdf	M12

8.2.7 Files

The roadmap of the Files component is summarized in Table 73.

Table 73 – Roadmap of Files Component

Functionality	Month	Functionality	Month
Download Files	IMP	Generate upload Keys	IMP
Upload Files		Search Files	
Get File list		Upload file in chunks when files are big	M12
Index Images		Manage file usage and access rights	M12

8.2.8 Monitoring

The roadmap of Monitoring component is summarized in Table 74.

Table 74 – Roadmap of Monitoring Component

Functionality	Month	Functionality	Month
Management of logs at Database	IMP	Support multiple datasources (logs, monitoring components)	M12
Management of logs at Graylog		Management of logs in Graylog	M17
Performance graphs of components		Retrieve data from logging system	M17
Export performance data of components in JSON		Export performance data (CPU, mem usage) (JSON)	M12
Proposal Analysis	M12		

8.2.9 Notify

The Notifications component's roadmap is presented in Table 75.

Table 75 – Roadmap of Notifications Component

Functionality	Month	Functionality	Month
Management of Emails' Groups	IMP	SMS Notifications (push notifications)	M20
Management of Email templates		User subscribed notifications (follow-up)	M20
Send Emails		Time notifications (e.g. deadline of event approaching)	M20
Time notifications (e.g. deadline of event approaching)		Newsletter	M20

8.2.10 Open Data

The Open Data component is planned for the 2nd prototype of the EMPATIA Platform. This component will implement the functionalities detailed in section 3.4.15.

8.2.11 Orchestrator

The orchestrator component has implemented all the functionalities that have been identified in the initial design phase for the prototype version 1 (P1) and for the prototype version 2 (P2). Considering this, the roadmap for this component only updates the functionalities reported in section 3.4.10, if the new or other functionalities planned for other components require modifications in the Orchestrator component. For instance, if the planned functionalities of Analytics require additional orchestration from the Orchestration component.

8.2.12 Questionnaires

The Questionnaires component assures the functionalities related with questionnaires. The overall functionalities supported by this component are summarized in Table 76.

Table 76 – Roadmap of Questionnaires Component

Functionality	Month	Functionality	Month
Management of Question groups	IMP	Management of Icons	IMP
Management of Questions		Management of Event Schedules	
Management of Questions Types		Management of Participants in Questionnaires	
Management of Questions Options		Export answers - EXCEL	M16

Management of Forms		Export answers - WORD	M16
Management of Form Configurations		Export answers - ODF	M16
Management of Form Reply answers		Export answers - CSV	M16
Anonymous Answers		Export answers - JSON	M16
Answers reports and analytics		Export answers - Others, like R, SPSS, SAS	M16
Export answers to PDF		Data anonymization	M16
		Quick Questionnaires	M16

8.2.13 Voting

The roadmap of the Voting component is summarized in Table 77.

Table 77 – Roadmap of Voting Component

Functionality	Month	Functionality	Month
Management of Votes	IMP	Submit votes	IMP
List Votes		Statistics of Votes per Event	
Details of Votes		Votes Rules	
Timeline of Votes		Event Votes	
Management of Votes Methods		Management of Events Votes	
Management of Votes Method Groups		Anonymous voting	M12

8.2.14 WUI

The WUI component is responsible to receive all the requests and forward them to the right component for the respective processing. This component will be enhanced if the planned functionalities for the remaining components require modifications in the WUI component.

8.2.15 Kiosk

The roadmap for the kiosk component is summarized in Table 78.

Table 78 – Roadmap of Kiosk Component

Functionality	Month	Functionality	Month
Login in Kiosks	IMP	Show multiple languages	M12
Logout in Kiosks		Show multiple topics at the same time	M14
		Support multiple types of kiosks (without and with screen)	M14
Show/Select a proposal/Idea	M20		

9 Conclusion

This deliverable presented the first version of the EMPATIA Platform architecture, according to the design and implementation activities in WP2. The design and implementation of the architecture has been driven by the priorities of the identified requirements in terms of functional and non-functional aspects. Such requirements are mainly derived from a SoA analysis and from the EMPATIA use case scenarios, specified in D1.1. The requirements associated with the must implement priority level have been implemented and documented in this deliverable.

The EMPATIA architecture is composed by diverse components with specific functionalities and designed to accomplish scalability, redundancy and fault tolerance goals. The EMPATIA Platform was also architected to support different deployment models including cloud, docker containers, as well as the concept of EMPATIAaaS. This last model enables commercial exploitation activities of the platform. Indeed, care has been taken to provide enough detail of the design and document components and respective APIs.

While the development is based on an incremental and iterative process, several EMPATIA/EMPAVILLE demonstrations allowed the collection of key user feedback, which is being analysed to be included in the next phases of the development cycle.

10 References

[1] EMPATIA, Deliverable D2.1 – Platform architecture and specification - preliminary, July 2016

[2] EMPATIA, Deliverable D1.1, April 2016

[3] EMPATIA, Deliverable D2.2 – Development and integration of the platform - preliminary, October 2016

[4] CKAN, the world's leading open-source data portal platform, available at: http://ckan.org/.

[5] Your Priorities, available at: https://yrpri.org/domain/3

[6] Non-Profit Citizens Foundation Iceland, available at: http://www.citizens.is/

[7] Loomio, available at: https://www.loomio.org/

[8] Loomio Manual, available at: https://loomio.gitbooks.io/manual/content/en/

[9] CitizenBudget, available at: http://citizenbudget.com/

[10] Paris Budget Participatif, available at: https://budgetparticipatif.paris.fr/bp/

[11] Paris Budget Participatif, 2nd Cycle, available at: Cycle https://budgetparticipatif.paris.fr/bp/jsp/site/Portal.jsp?page=search-solr&conf=list_projects

[12] Kriplean, T., Morgan, J., Freelon, D., Borning, A., & Bennett, L. (2012). Supporting reflective public thought with considerit. *Proceedings of the ACM 2012 Conference on Computer Supported Cooperative Work - CSCW '12*, 265. http://doi.org/10.1145/2145204.2145249

[13] Participare, available at: https://www.participare.io/

[14] Participare export of data, available at: https://participare.uservoice.com/knowledgebase/articles/506352-can-i-export-my-data

[15] Participare, available APIs, available at: https://participare.uservoice.com/knowledgebase/articles/505907-what-api-are-available

[16] Consul, available at: https://decide.madrid.es/

[17] Consul ethical principles, available at: https://transparencia.madrid.es/

[18] wmOP, available online: http://www.wiremaze.com/wmop.html

[19] Bürgerhaushalt Lichtenberg, available online: https://www.buergerhaushalt-lichtenberg.de

[20] Binary Objects, available online: http://www.binary-objects.de/

[21] DEMOS, available online: http://www.demos-deutschland.de/

[22] AppCivist, available online: http://citris-uc.org/social-apps-lab/project/appcivist/

[23] Pathak, A., Issarny, V., & Holston, J. (2015). AppCivist - A Service-Oriented Software Platform for Socially Sustainable Activism. *Proceedings - International Conference on Software Engineering*, *2*, 515–518. http://doi.org/10.1109/ICSE.2015.185

[24] D-CENT project, available online: http://dcentproject.eu/

[25] OpenAhjo Open API, available online: http://dev.hel.fi/apis/openahjo/

[26] BiPart, available online: http://www.bipart.it/

[27] OpenDCN, available online: http://www.opendcn.org/

[28] Foundation RCM, available online: http://www.fondazionercm.it/

[29] Civic Informatics Laboratory, available online: http://www.lic.di.unimi.it/

[30] Parelon, available online: http://www.parelon.com/

[31] LiquidFeedback, available online: http://liquidfeedback.org/

[32] OpenBudgets.eu, available online: http://openbudgets.eu/

[33] DIGIWHIST: The Digital Whistleblower, available online: http://digiwhist.eu/

[34] Opentender.eu, available online: https://opentender.eu/

[35] MyNeighbourhood, available online: http://www.my-n.eu/

[36] De Olho nas Metas, available at: http://deolhonasmetas.org.br/

[37] Monitoraggio Opere Pubbliche, available at: http://monitoraggio.nuvole.org

[38] OpenExpo 2015, available at: http://dati.openexpo2015.it/it

[39] Open Ricostruzione, available at: http://www.openricostruzione.it

[40] OpenSpending, available at: https://openspending.org

[41] Bilanciovico, available at: http://bilanciovico.it

[42] Dirolamia, available at: http://dirolamia.it

[43] Monzapartecipa, available at: http://www.monzapartecipa.it/Monza_partecipa/bilancio-partecipativo/Avanzamento-progetti-vincitori/

[44] Berner, Maureen, and Sonya Smith. "The State of the States: A Review of State Requirements for Citizen Participation in the Local Government Budget Process." *State & Local Government Review* 36.2 (2004): 140-50. Web

[45] OATH Authentication, available at: https://openauthentication.org/about-oath/

[46] XE Currency Converter, available online: http://www.xe.com/currencyconverter/

[47] Apache Hadoop, available online: http://hadoop.apache.org/

[48] Graylog, available online: https://www.graylog.org/

[49] Debian, available online: https://www.debian.org/

[50] PHP, available online: http://php.net/

[51] Laravel framework, available online: https://laravel.com/

[52] MySQL, available online: https://www.mysql.com/

[53] Redis, available online: http://redis.io/

[54] Socket.io, available online: http://socket.io/

[55] JSON Web Tokens, available online: https://jwt.io/

[56] OAuth 2.0, available online: http://oauth.net/2/

[57] Docker, available online: https://www.docker.com/

[58] Docker Hub, available online: https://hub.docker.com/u/empatia/

[59] GIT, available online: https://git-scm.com/

[60] EMPATIA GitHub, available online: https://github.com/EMPATIA

[61] RAML, available online: http://raml.org/

[62] WADL, available online: https://www.w3.org/Submission/wadl/

[63] API Blueprint, available online: https://apiblueprint.org/

[64] OAI, available online: https://openapis.org/

[65] JSON API, available online: http://jsonapi.org/

[66] LiberOpinion, available at: https://liberopinion.com/

[67] Consider.it, available at: https://consider.it

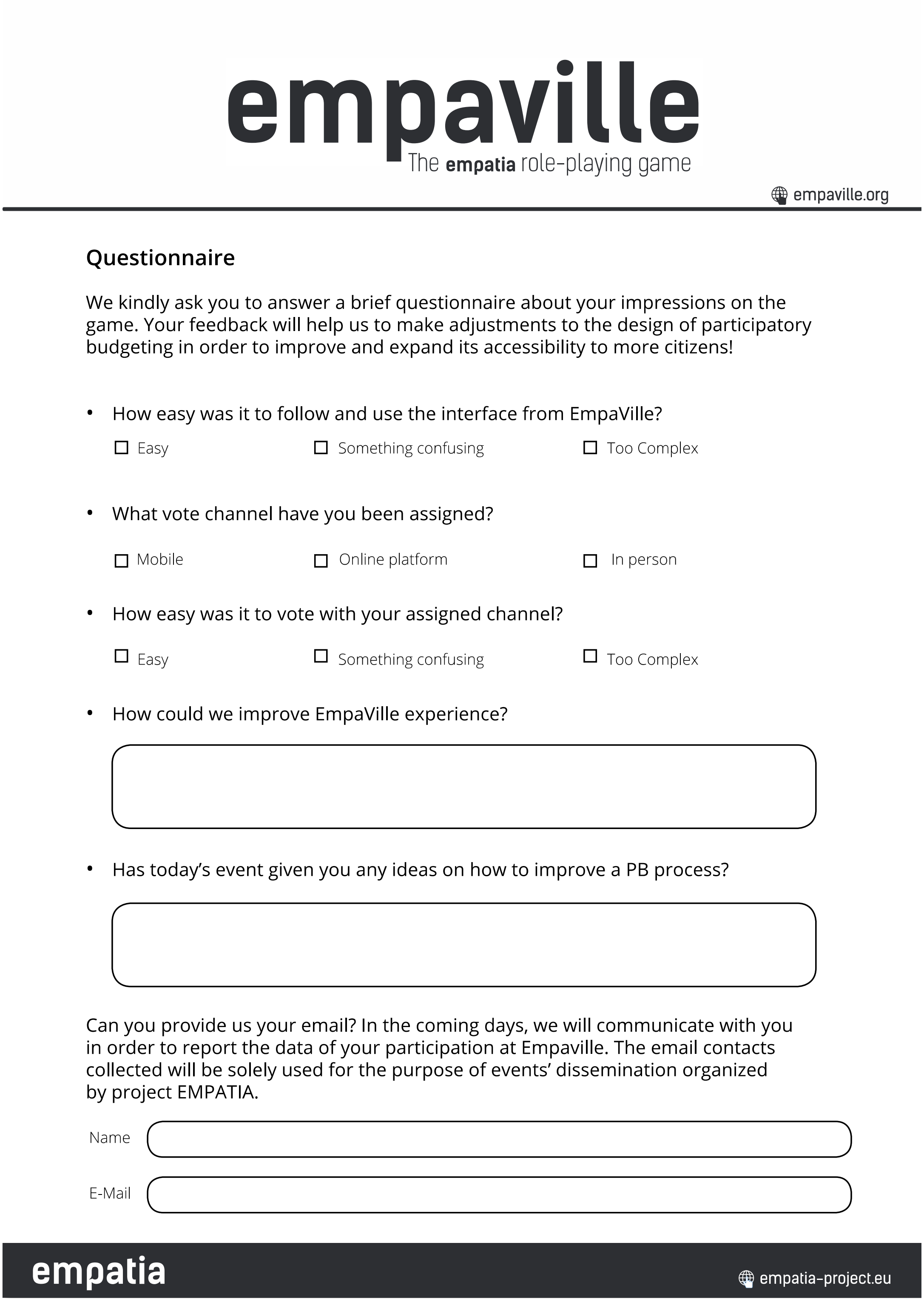
[68] Drupal, available at: https://www.drupal.org

[69] CKAN github repository, available at: https://github.com/ckan/ckan

[70] Boaboa crowdfunding platform, available at: http://www.boaboa.pt

Annex A – EMPAVILLE QUESTIONNAIRE

The questionnaire that was elaborated to collect feedback from users using the EMPAVILLE questionnaire, is depicted in Figure 6.



empaville
The empatia role-playing game
empaville.org

Questionnaire

We kindly ask you to answer a brief questionnaire about your impressions on the game. Your feedback will help us to make adjustments to the design of participatory budgeting in order to improve and expand its accessibility to more citizens!

- How easy was it to follow and use the interface from EmpaVille?
  ☐ Easy ☐ Something confusing ☐ Too Complex
- What vote channel have you been assigned?
  ☐ Mobile ☐ Online platform ☐ In person
- How easy was it to vote with your assigned channel?
  ☐ Easy ☐ Something confusing ☐ Too Complex
- How could we improve EmpaVille experience?
- Has today's event given you any ideas on how to improve a PB process?

Can you provide us your email? In the coming days, we will communicate with you in order to report the data of your participation at Empaville. The email contacts collected will be solely used for the purpose of events' dissemination organized by project EMPATIA.

Name

E-Mail

empatia
empatia-project.eu

Figure 6 – EMPAVILLE questionnaire

Annex B – EMPATIA API Specification

Analytics Module

Version 1.0.0

Contact information
pvalente@onesource.pt (mailto:pvalente@onesource.pt)

Paths

`/q/{form_key}/statistics`

`GET /q/{form_key}/statistics`

Questionnaire Statistics

Summary

Show questionnaire statistics

Parameters

Name	Located in	Description	Required	Schema
form_key	path	Form Key	Yes	⇄ `string`
X-MODULE-TOKEN	header	Module Token	Yes	⇄ `string`
X-ENTITY-KEY	header	Entity Key	Yes	⇄ `string`

Responses

Code	Description	Schema
200	Show questionnaire statistics	⇄
400	Error trying to retrieve data	⇄

200 Show questionnaire statistics

```
▼questionnaireStatisticsReply {
  all of:
    ▼ {
      form_key:          string
      entity_key:        string
      title:             string
      description:       string
      public:            boolean
      created_by:        string
      start_date:        string
      end_date:          string
      link:              string
      file_id:           integer
      created_at:        string
      updated_at:        string
      question_groups:   ►[]
    }
}
```

400 Error trying to retrieve data

```
▼analyticsErrorDefault {
   error: string
}
```

Try this operation

`/voteEvent/{event_key}/votes/{cb_key}`

`GET /voteEvent/{event_key}/votes/{cb_key}`

Vote Statistics

Summary

Show Information about vote

Parameters

Name	Located in	Description	Required	Schema
event_key	path	Event Key	Yes	⇄ `string`
cb_key	path	Cb Key	Yes	⇄ `string`
X-MODULE-TOKEN	header	Module Token	Yes	⇄ `string`
LANG-CODE	header	User Language	Yes	⇄ `string`
LANG-CODE-DEFAULT	header	Entity default Language	Yes	⇄ `string`

Responses

Code	Description	Schema

200	Show vote Information	⇄ ▼ **votesInfoReply {**
		all of:
		▼ **{**
		data: ▶ [ ]
		summary: ▶ **{ }**
		}
		}

400	Error trying to retrieve data	⇄ ▼ **analyticsErrorDefault {**
		error: string
		}

Try this operation

`/voteEventDaily/{event_key}/votes/{cb_key}`

`GET /voteEventDaily/{event_key}/votes/{cb_key}`

Vote Statistics

Summary

Show Information about daily votes

Parameters

Name	Located in	Description	Required	Schema
event_key	path	Event Key	Yes	⇄ `string`
cb_key	path	Cb Key	Yes	⇄ `string`
X-MODULE-TOKEN	header	Module Token	Yes	⇄ `string`

Responses

Code **Description** **Schema**

200 Show vote daily Information ⇄

```
▼ votesDailyReply {
   all of:
     ▼ {
         data: ▼ {
                 all of:
                   ▼ {
                       date_value: ▶ { }
                     }
               }
       }
}
```

400 Error trying to retrieve data ⇄

```
▼ analyticsErrorDefault {
    error: string
}
```

Try this operation

/voteEvent/{event_key}/countByGender/{cb_key}

`GET /voteEvent/{event_key}/countByGender/{cb_key}`

Vote Statistics

Summary

Show votes count by gender statistics

Parameters

Name	Located in	Description	Required	Schema
event_key	path	Event Key	Yes	⇄ `string`
cb_key	path	Cb Key	Yes	⇄ `string`
X-MODULE-TOKEN	header	Module Token	Yes	⇄ `string`
X-AUTH-TOKEN	header	User Auth Token	Yes	⇄ `string`

Responses

Code	Description	Schema
200	Show vote count by gender statistics	⇄
400	Error trying to retrieve data	⇄

200

```
▼ votesCountByParamReply {
  all of:
     ▼  {
          data: ▶ { }
     }
}
```

400

```
▼ analyticsErrorDefault {
   error: string
}
```

☐ Try this operation

/voteEvent/{event_key}/firstByGender/{cb_key}

GET /voteEvent/{event_key}/firstByGender/{cb_key}

Vote Statistics

Summary

Show first votes by gender statistics

Parameters

Name	Located in	Description	Required	Schema
event_key	path	Event Key	Yes	⇄ `string`
cb_key	path	Cb Key	Yes	⇄ `string`
X-MODULE-TOKEN	header	Module Token	Yes	⇄ `string`
X-AUTH-TOKEN	header	User Auth Token	Yes	⇄ `string`

Responses

Code **Description** **Schema**

200 Show first votes by gender statistics ⇄

```
▼ votesFirstByParamReply {
   all of:
        ▼ {
           data: ▶ { }
        }
}
```

400 Error trying to retrieve data ⇄

```
▼ analyticsErrorDefault {
    error: string
}
```

Try this operation

/voteEvent/{event_key}/secondByGender/{cb_key}

GET /voteEvent/{event_key}/secondByGender/{cb_key}

Vote Statistics

Summary

Show second votes by gender statistics

Parameters

Name	Located in	Description	Required	Schema
event_key	path	Event Key	Yes	⇄ string
cb_key	path	Cb Key	Yes	⇄ string
X-MODULE-TOKEN	header	Module Token	Yes	⇄ string
X-AUTH-TOKEN	header	User Auth Token	Yes	⇄ string

Responses

Code	Description	Schema
200	Show second votes by gender statistics	⇄

```
▼ votesFirstByParamReply {
   all of:
      ▼ {
         data: ▶ { }
      }
}
```

Code	Description	Schema
400	Error trying to retrieve data	⇄

```
▼ analyticsErrorDefault {
   error: string
}
```

Try this operation

/voteEvent/{event_key}/votesByProfession/{cb_key}

`GET /voteEvent/{event_key}/votesByProfession/{cb_key}`

Vote Statistics

Summary

Show votes by profession statistics

Parameters

Name	Located in	Description	Required	Schema
event_key	path	Event Key	Yes	⇄ `string`
cb_key	path	Cb Key	Yes	⇄ `string`
X-MODULE-TOKEN	header	Module Token	Yes	⇄ `string`
LANG-CODE	header	User Language	Yes	⇄ `string`
LANG-CODE-DEFAULT	header	Entity default Language	Yes	⇄ `string`
X-AUTH-TOKEN	header	User Auth Token	Yes	⇄ `string`

Responses

Code	Description	Schema
200	Show votes by profession statistics	⇄
400	Error trying to retrieve data	⇄

200

```
▼ votesByProfessionReply {
  all of:
    ▼ {
      data:          ▼ [
                       ▼ {
                         title:        string
                         budget:       integer
                         category:     string
                         geo_area:     string
                         professions: ▶ { }
                       }
                     ]
      professions:   ▼ [
                       string
                     ]
    }
}
```

400

```
▼ analyticsErrorDefault {
   error: string
}
```

Try this operation

`/voteEvent/{event_key}/countByProfession/{cb_key}`

GET /voteEvent/{event_key}/countByProfession/{cb_key}

Vote Statistics

Summary

Show votes count by profession statistics

Parameters

Name	Located in	Description	Required	Schema
event_key	path	Event Key	Yes	⇄ string
cb_key	path	Cb Key	Yes	⇄ string
X-MODULE-TOKEN	header	Module Token	Yes	⇄ string
X-AUTH-TOKEN	header	User Auth Token	Yes	⇄ string

Responses

Code	Description	Schema
200	Show vote count by profession statistics	⇄
400	Error trying to retrieve data	⇄

200

```
▼ votesCountByParamReply {
    all of:
       ▼ {
          data: ▶ { }
       }
}
```

400

```
▼ analyticsErrorDefault {
    error: string
}
```

Try this operation

/voteEvent/{event_key}/firstByProfession/{cb_key}

`GET /voteEvent/{event_key}/firstByProfession/{cb_key}`

Vote Statistics

Summary

Show first votes by profession statistics

Parameters

Name	Located in	Description	Required	Schema
event_key	path	Event Key	Yes	⇄ `string`
cb_key	path	Cb Key	Yes	⇄ `string`
X-MODULE-TOKEN	header	Module Token	Yes	⇄ `string`
X-AUTH-TOKEN	header	User Auth Token	Yes	⇄ `string`

Responses

Code	Description		Schema
200	Show first votes by profession statistics	⇄	see below
400	Error trying to retrieve data	⇄	see below

200 schema:

```
▼ votesFirstByParamReply {
   all of:
      ▼ {
         data: ▼ {
                 all of:
                    ▼ {
                       parameter_type: ▶
                                        {
                                        }
                    }
               }
      }
}
```

400 schema:

```
▼ analyticsErrorDefault {
   error: string
}
```

Try this operation

`/voteEvent/{event_key}/secondByProfession/{cb_key}`

`GET /voteEvent/{event_key}/secondByProfession/{cb_key}`

Vote Statistics

Summary

Show second votes by profession statistics

Parameters

Name	Located in	Description	Required	Schema
event_key	path	Event Key	Yes	⇄ `string`
cb_key	path	Cb Key	Yes	⇄ `string`
X-MODULE-TOKEN	header	Module Token	Yes	⇄ `string`
X-AUTH-TOKEN	header	User Auth Token	Yes	⇄ `string`

Responses

Code	Description	Schema
200	Show second votes by profession statistics	⇄
400	Error trying to retrieve data	⇄

200

```
▼ votesFirstByParamReply {
   all of:
      ▼  {
         data: ▼  {
                  all of:
                     ▼  {
                        parameter_type: ▶
                                          {
                                          }
                        }
                  }
         }
}
```

400

```
▼ analyticsErrorDefault {
   error: string
}
```

Try this operation

/voteEvent/{event_key}/votesByChannel/{cb_key}

`GET /voteEvent/{event_key}/votesByChannel/{cb_key}`

Vote Statistics

Summary

Show votes by channel statistics

Parameters

Name	Located in	Description	Required	Schema
event_key	path	Event Key	Yes	⇄ string
cb_key	path	Cb Key	Yes	⇄ string
X-MODULE-TOKEN	header	Module Token	Yes	⇄ string
LANG-CODE	header	User Language	Yes	⇄ string
LANG-CODE-DEFAULT	header	Entity default Language	Yes	⇄ string
X-AUTH-TOKEN	header	User Auth Token	Yes	⇄ string

Responses

Code	Description	Schema
200	Show votes by channel statistics	⇄

```
▼votesByChannelReply {
 all of:
    ▼ {
     data: ▼[
             ▼ {
               title:   string
               budget:  integer
               category: string
               geo_area: string
               channels: ▼ {
                           all of:
                             ▼ {
                               balance:   integer
                               positives: integer
                               negatives: integer
                             }
                         }
             }
           ]
    }
}
```

400	Error trying to retrieve data	⇄

```
▼analyticsErrorDefault {
  error: string
}
```

Try this operation

/voteEvent/{event_key}/countByChannel/{cb_key}

`GET /voteEvent/{event_key}/countByChannel/{cb_key}`

Vote Statistics

Summary

Show votes count by channel statistics

Parameters

Name	Located in	Description	Required	Schema
event_key	path	Event Key	Yes	⇄ `string`
cb_key	path	Cb Key	Yes	⇄ `string`
X-MODULE-TOKEN	header	Module Token	Yes	⇄ `string`
X-AUTH-TOKEN	header	User Auth Token	Yes	⇄ `string`

Responses

Code	Description	Schema
200	Show vote count by channel statistics	⇄
400	Error trying to retrieve data	⇄

200

```
▼votesCountByParamReply {
  all of:
    ▼ {
        data: ▼ {
                all of:
                  ▼ {
                     parameter_value: ▶
                                      {
                                      }
                    }
              }
      }
}
```

400

```
▼analyticsErrorDefault {
  error: string
}
```

Try this operation

/voteEvent/{event_key}/firstByChannel/{cb_key}

GET /voteEvent/{event_key}/firstByChannel/{cb_key}

Vote Statistics

Summary

Show first votes by channel statistics

Parameters

Name	Located in	Description	Required	Schema
event_key	path	Event Key	Yes	⇄ `string`
cb_key	path	Cb Key	Yes	⇄ `string`
X-MODULE-TOKEN	header	Module Token	Yes	⇄ `string`
X-AUTH-TOKEN	header	User Auth Token	Yes	⇄ `string`

Responses

Code	Description	Schema
200	Show first votes by channel statistics	⇄
400	Error trying to retrieve data	⇄

200 Show first votes by channel statistics ⇄

```
▼votesFirstByParamReply {
  all of:
    ▼ {
       data: ▼ {
               all of:
                 ▼ {
                     parameter_type: ▶
                                       {
                                       }
                   }
             }
      }
}
```

400 Error trying to retrieve data ⇄

```
▼analyticsErrorDefault {
   error: string
}
```

Try this operation

/voteEvent/{event_key}/secondByChannel/{cb_key}

`GET /voteEvent/{event_key}/secondByChannel/{cb_key}`

Vote Statistics

Summary

Show second votes by channel statistics

Parameters

Name	Located in	Description	Required	Schema
event_key	path	Event Key	Yes	⇄ `string`
cb_key	path	Cb Key	Yes	⇄ `string`
X-MODULE-TOKEN	header	Module Token	Yes	⇄ `string`
X-AUTH-TOKEN	header	User Auth Token	Yes	⇄ `string`

Responses

Code	Description	Schema
200	Show second votes by channel statistics	⇄
400	Error trying to retrieve data	⇄

200

```
▼ votesFirstByParamReply {
   all of:
      ▼ {
         data: ▼ {
                  all of:
                     ▼ {
                        parameter_type: ▶
                                        {
                                        }
                     }
               }
      }
}
```

400

```
▼ analyticsErrorDefault {
   error: string
}
```

Try this operation

`/voteEvent/{event_key}/votesByNb/{cb_key}`

`GET /voteEvent/{event_key}/votesByNb/{cb_key}`

Vote Statistics

Summary

Show votes by neighbourhood statistics

Parameters

Name	Located in	Description	Required	Schema
event_key	path	Event Key	Yes	⇄ `string`
cb_key	path	Cb Key	Yes	⇄ `string`
X-MODULE-TOKEN	header	Module Token	Yes	⇄ `string`
LANG-CODE	header	User Language	Yes	⇄ `string`
LANG-CODE-DEFAULT	header	Entity default Language	Yes	⇄ `string`
X-AUTH-TOKEN	header	User Auth Token	Yes	⇄ `string`

Responses

Code	Description	Schema
200	Show votes by neighbourhood statistics	⇄
400	Error trying to retrieve data	⇄

200:

```
▼ votesByNbReply {
  all of:
     ▼ {
        data: ▼ [
                 ▼ {
                    title:                string
                    budget:               integer
                    category:             string
                    geo_area:             string
                    neighbourhood_area:  ► { }
                   }
               ]
       }
}
```

400:

```
▼ analyticsErrorDefault {
   error: string
}
```

Try this operation

`/voteEvent/{event_key}/countByNb/{cb_key}`

GET /voteEvent/{event_key}/countByNb/{cb_key}

Vote Statistics

Summary

Show votes count by neighbourhood statistics

Parameters

Name	Located in	Description	Required	Schema
event_key	path	Event Key	Yes	⇄ `string`
cb_key	path	Cb Key	Yes	⇄ `string`
X-MODULE-TOKEN	header	Module Token	Yes	⇄ `string`
X-AUTH-TOKEN	header	User Auth Token	Yes	⇄ `string`

Responses

Code	Description	Schema
200	Show vote count by neighbourhood statistics	⇄
400	Error trying to retrieve data	⇄

200

```
▼votesCountByParamReply {
  all of:
    ▼ {
      data: ▼ {
        all of:
          ▼ {
            parameter_value: ▶
                              {
                              }
          }
      }
    }
}
```

400

```
▼analyticsErrorDefault {
   error: string
}
```

Try this operation

/voteEvent/{event_key}/firstByNb/{cb_key}

`GET /voteEvent/{event_key}/firstByNb/{cb_key}`

Vote Statistics

Summary

Show first votes by neighbourhood statistics

Parameters

Name	Located in	Description	Required	Schema
event_key	path	Event Key	Yes	⇄ `string`
cb_key	path	Cb Key	Yes	⇄ `string`
X-MODULE-TOKEN	header	Module Token	Yes	⇄ `string`
X-AUTH-TOKEN	header	User Auth Token	Yes	⇄ `string`

Responses

Code	Description	Schema
200	Show first votes by neighbourhood statistics	⇄
400	Error trying to retrieve data	⇄

200

```
▼votesFirstByParamReply {
  all of:
    ▼ {
        data: ▼ {
              all of:
                ▼ {
                    parameter_type: ▶
                                    {
                                    }
                  }
              }
      }
}
```

400

```
▼analyticsErrorDefault {
   error: string
}
```

Try this operation

`/voteEvent/{event_key}/secondByNb/{cb_key}`

`GET /voteEvent/{event_key}/secondByNb/{cb_key}`

Vote Statistics

Summary

Show second votes by neighbourhood statistics

Parameters

Name	Located in	Description	Required	Schema
event_key	path	Event Key	Yes	⇄ `string`
cb_key	path	Cb Key	Yes	⇄ `string`
X-MODULE-TOKEN	header	Module Token	Yes	⇄ `string`
X-AUTH-TOKEN	header	User Auth Token	Yes	⇄ `string`

Responses

Code	Description	Schema
200	Show second votes by neighbourhood statistics	⇄
400	Error trying to retrieve data	⇄

200

```
▼ votesFirstByParamReply {
    all of:
        ▼ {
            data: ▼ {
                all of:
                    ▼ {
                        parameter_type: ▶
                                        {
                                        }
                    }
            }
        }
}
```

400

```
▼ analyticsErrorDefault {
    error: string
}
```

Try this operation

`/voteEvent/{event_key}/countByAge/{cb_key}`

GET /voteEvent/{event_key}/countByAge/{cb_key}

Vote Statistics

Summary

Show votes count by age statistics

Parameters

Name	Located in	Description	Required	Schema
event_key	path	Event Key	Yes	⇄ `string`
cb_key	path	Cb Key	Yes	⇄ `string`
X-MODULE-TOKEN	header	Module Token	Yes	⇄ `string`
X-AUTH-TOKEN	header	User Auth Token	Yes	⇄ `string`

Responses

Code	Description	Schema
200	Show vote count by age statistics	⇄
400	Error trying to retrieve data	⇄

200

```
▼ votesCountByParamReply {
   all of:
      ▼ {
         data: ▼ {
                  all of:
                     ▼ {
                        parameter_value: ▶
                                           {
                                           }
                     }
                 }
        }
}
```

400

```
▼ analyticsErrorDefault {
   error: string
}
```

Try this operation

/voteEvent/{event_key}/firstByAge/{cb_key}

GET /voteEvent/{event_key}/firstByAge/{cb_key}

Vote Statistics

Summary

Show first votes by age statistics

Parameters

Name	Located in	Description	Required	Schema
event_key	path	Event Key	Yes	⇄ `string`
cb_key	path	Cb Key	Yes	⇄ `string`
X-MODULE-TOKEN	header	Module Token	Yes	⇄ `string`
X-AUTH-TOKEN	header	User Auth Token	Yes	⇄ `string`

Responses

Code	Description	Schema
200	Show first votes by age statistics	⇄
400	Error trying to retrieve data	⇄

Schema for **200**:

```
▼ votesFirstByParamReply {
  all of:
     ▼ {
        data: ▼ {
                 all of:
                    ▼ {
                       parameter_type: ▶
                                        {
                                        }
                    }
                 }
     }
}
```

Schema for **400**:

```
▼ analyticsErrorDefault {
  error: string
}
```

Try this operation

/voteEvent/{event_key}/secondByAge/{cb_key}

GET /voteEvent/{event_key}/secondByAge/{cb_key}

Vote Statistics

Summary

Show second votes by age statistics

Parameters

Name	Located in	Description	Required	Schema
event_key	path	Event Key	Yes	⇄ string
cb_key	path	Cb Key	Yes	⇄ string
X-MODULE-TOKEN	header	Module Token	Yes	⇄ string
X-AUTH-TOKEN	header	User Auth Token	Yes	⇄ string

Responses

Code	Description	Schema
200	Show second votes by age statistics	⇄
400	Error trying to retrieve data	⇄

200

```
▼votesFirstByParamReply {
   all of:
      ▼ {
         data: ▼ {
               all of:
                  ▼ {
                     parameter_type: ▶
                                     {
                                     }
                  }
               }
      }
}
```

400

```
▼analyticsErrorDefault {
   error: string
}
```

Try this operation

/voteEvent/{event_key}/statisticsByDate

GET /voteEvent/{event_key}/statisticsByDate

Vote Statistics

Summary

Show vote statistics by date

Parameters

Name	Located in	Description	Required	Schema
event_key	path	Event Key	Yes	⇄ `string`
X-MODULE-TOKEN	header	Module Token	Yes	⇄ `string`

Responses

Code	Description	Schema
200	Show vote statistics by date	⇄
400	Error trying to retrieve data	⇄

200

```
▼voteByDateReply {
  all of:
     ▼ {
        data: ▼ {
               all of:
                 ▼ {
                   negative_vote: boolean
                   total:         ▶ { }
                   balance:       ▶ { }
                   positive:      ▶ { }
                   negative:      ▶ { }
                  }
             }
       }
}
```

400

```
▼analyticsErrorDefault {
   error: string
}
```

Try this operation

/voteEvent/{event_key}/statisticsByTown

`GET /voteEvent/{event_key}/statisticsByTown`

Vote Statistics

Summary

Show vote statistics by town

Parameters

Name	Located in	Description	Required	Schema
event_key	path	Event Key	Yes	⇄ `string`
X-AUTH-TOKEN	header	User Auth Token	Yes	⇄ `string`
X-ENTITY-KEY	header	Entity Key	Yes	⇄ `string`
X-MODULE-TOKEN	header	Module Token	Yes	⇄ `string`
LANG-CODE	header	User Language	Yes	⇄ `string`
LANG-CODE-DEFAULT	header	Entity default Language	Yes	⇄ `string`

Responses

Code	Description	Schema
200	Show vote statistics by town	⇄
400	Error trying to retrieve data	⇄

200

```
▼ votesByTownReply {
   all of:
     ▼ {
        data: ▼ {
                all of:
                  ▼ {
                     in_person_votes: ▶
                                       { }
                     web_votes:       ▶
                                       { }
                     all_votes:       ▶
                                       { }
                    }
              }
      }
}
```

400

```
▼ analyticsErrorDefault {
   error: string
}
```

Try this operation

/voteEvent/{event_key}/statisticsByAge

`GET /voteEvent/{event_key}/statisticsByAge`

Vote Statistics

Summary

Show vote statistics by age

Parameters

Name	Located in	Description	Required	Schema
event_key	path	Event Key	Yes	⇄ `string`
X-AUTH-TOKEN	header	User Auth Token	Yes	⇄ `string`
X-ENTITY-KEY	header	Entity Key	Yes	⇄ `string`
X-MODULE-TOKEN	header	Module Token	Yes	⇄ `string`
LANG-CODE	header	User Language	Yes	⇄ `string`
LANG-CODE-DEFAULT	header	Entity default Language	Yes	⇄ `string`

Responses

Code	Description	Schema
200	Show vote statistics by age	⇄
400	Error trying to retrieve data	⇄

200

```
▼ votesByParameterTownReply {
   all of:
     ▼ {
        data: ▼ {
                all of:
                  ▼ {
                     in_person_votes: ▶
                                       { }
                     web_votes:        ▶
                                       { }
                  }
              }
     }
}
```

400

```
▼ analyticsErrorDefault {
   error: string
}
```

Try this operation

`/voteEvent/{event_key}/statisticsByGender`

`GET /voteEvent/{event_key}/statisticsByGender`

Vote Statistics

Summary

Show vote statistics by gender

Parameters

Name	Located in	Description	Required	Schema
event_key	path	Event Key	Yes	⇄ `string`
X-AUTH-TOKEN	header	User Auth Token	Yes	⇄ `string`
X-ENTITY-KEY	header	Entity Key	Yes	⇄ `string`
X-MODULE-TOKEN	header	Module Token	Yes	⇄ `string`
LANG-CODE	header	User Language	Yes	⇄ `string`
LANG-CODE-DEFAULT	header	Entity default Language	Yes	⇄ `string`

Responses

Code	Description	Schema
200	Show vote statistics by age	⇄
400	Error trying to retrieve data	⇄

200

```
▼ votesByParameterTownReply {
   all of:
     ▼ {
        data: ▼ {
                all of:
                  ▼ {
                     in_person_votes: ▶
                                      { }
                     web_votes:       ▶
                                      { }
                    }
                }
       }
  }
```

400

```
▼ analyticsErrorDefault {
   error: string
  }
```

Try this operation

`/voteEvent/{event_key}/statisticsByProfession`

GET /voteEvent/{event_key}/statisticsByProfession

Vote Statistics

Summary

Show vote statistics by profession

Parameters

Name	Located in	Description	Required	Schema
event_key	path	Event Key	Yes	⇄ `string`
X-AUTH-TOKEN	header	User Auth Token	Yes	⇄ `string`
X-ENTITY-KEY	header	Entity Key	Yes	⇄ `string`
X-MODULE-TOKEN	header	Module Token	Yes	⇄ `string`
LANG-CODE	header	User Language	Yes	⇄ `string`
LANG-CODE-DEFAULT	header	Entity default Language	Yes	⇄ `string`

Responses

Code	Description	Schema
200	Show vote statistics by profession	⇄
400	Error trying to retrieve data	⇄

200

```
▼votesByParameterTownReply {
  all of:
    ▼ {
      data: ▼ {
              all of:
                ▼ {
                    in_person_votes: ▶ { }
                    web_votes:       ▶ { }
                  }
            }
      }
}
```

400

```
▼analyticsErrorDefault {
  error: string
}
```

Try this operation

/voteEvent/{event_key}/statisticsByEducation

`GET /voteEvent/{event_key}/statisticsByEducation`

Vote Statistics

Summary

Show vote statistics by education

Parameters

Name	Located in	Description	Required	Schema
event_key	path	Event Key	Yes	⇄ `string`
X-AUTH-TOKEN	header	User Auth Token	Yes	⇄ `string`
X-ENTITY-KEY	header	Entity Key	Yes	⇄ `string`
X-MODULE-TOKEN	header	Module Token	Yes	⇄ `string`
LANG-CODE	header	User Language	Yes	⇄ `string`
LANG-CODE-DEFAULT	header	Entity default Language	Yes	⇄ `string`

Responses

Code	Description	Schema
200	Show vote statistics by education	⇄
400	Error trying to retrieve data	⇄

200

```
▼ votesByParameterTownReply {
   all of:
      ▼ {
         data: ▼ {
                  all of:
                     ▼ {
                        in_person_votes: ▶
                                          {
                                          }
                        web_votes:       ▶
                                          {
                                          }
                       }
                 }
        }
}
```

400

```
▼ analyticsErrorDefault {
   error: string
}
```

Try this operation

`/voteEvent/{event_key}/statisticsByTop`

`POST /voteEvent/{event_key}/statisticsByTop`

Vote Statistics

Summary

Show vote statistics by Top topic

Parameters

Name	Located in	Description	Required	Schema
top	body	Top to show	Yes	⇄ (see below)
event_key	path	Event Key	Yes	⇄ `string`
X-MODULE-TOKEN	header	Module Token	Yes	⇄ `string`

top:

```
▼topStatistics {
   top: integer
}
```

Responses

Code	Description	Schema
200	Show vote statistics by Top topic	⇄ (see below)
400	Error trying to retrieve data	⇄ (see below)

200:

```
▼voteByTopReply {
   all of:
     ▼ {
        data: ▼ {
                 all of:
                   ▼ {
                      negative_vote: boolean
                      total:          ▶[]
                      balance:        ▶[]
                      positive:       ▶[]
                      negative:       ▶[]
                   }
               }
     }
}
```

400:

```
▼analyticsErrorDefault {
   error: string
}
```

Try this operation

`/voteEvent/{event_key}/statisticsTopByDate`

`POST /voteEvent/{event_key}/statisticsTopByDate`

Vote Statistics

Summary

Show vote statistics by Top topic and Date

Parameters

Name	Located in	Description	Required	Schema
top	body	Top to show	Yes	⇄ topStatistics
event_key	path	Event Key	Yes	⇄ `string`
X-MODULE-TOKEN	header	Module Token	Yes	⇄ `string`

top:

```
▼topStatistics {
   top: integer
}
```

Responses

Code	Description	Schema
200	Show vote statistics by Top topic and Date	⇄ voteByTopDateReply
400	Error trying to retrieve data	⇄ analyticsErrorDefault

200:

```
▼voteByTopDateReply {
  all of:
    ▼ {
      data: ▼ {
            all of:
              ▼ {
                negative_vote: boolean
                total:         ▶[]
                balance:       ▶[]
                positive:      ▶[]
                negative:      ▶[]
              }
            }
    }
}
```

400:

```
▼analyticsErrorDefault {
  error: string
}
```

Try this operation

`/voteEvent/{event_key}/statisticsByParameter`

`POST /voteEvent/{event_key}/statisticsByParameter`

Vote Statistics

Summary

Show vote statistics by parameter

Parameters

Name	Located in	Description	Required	Schema

Parameter Key	body	Parameter to show statistics	Yes	⇄ `▼parameterStatistics {`
				`parameter_key: string`
				`}`

event_key	path	Event Key	Yes	⇄ `string`
X-AUTH-TOKEN	header	User Auth Token	Yes	⇄ `string`
X-ENTITY-KEY	header	Entity Key	Yes	⇄ `string`
X-MODULE-TOKEN	header	Module Token	Yes	⇄ `string`
LANG-CODE	header	User Language	Yes	⇄ `string`
LANG-CODE-DEFAULT	header	Entity default Language	Yes	⇄ `string`

Responses

Code	Description	Schema
200	Show vote statistics by Top topic and Date	⇄

```
▼voteByParameterReply {
  all of:
    ▼ {
      data: ▼ {
              all of:
                ▼ {
                  negative_vote: boolean
                  total:         ▶[]
                  balance:       ▶[]
                  positive:      ▶[]
                  negative:      ▶[]
                }
            }
    }
}
```

Code	Description	Schema
400	Error trying to retrieve data	⇄

```
▼analyticsErrorDefault {
  error: string
}
```

Try this operation

Models

questionnaireStatisticsReply

```
▼questionnaireStatisticsReply {
  all of:
    ▼ {
      form_key:         string
      entity_key:       string
      title:            string
      description:      string
      public:           boolean
      created_by:       string
      start_date:       string
      end_date:         string
      link:             string
      file_id:          integer
      created_at:       string
      updated_at:       string
      question_groups: ▼[
                          ▼ {
                            question_group_key: string
                            form_id:            integer
                            title:              string
                            description:        string
                            position:           integer
                            created_at:         string
                            updated_at:         string
                            questions:          ►[]
                          }
                        ]
    }
}
```

votesInfoReply

```
▼votesInfoReply {
  all of:
    ▼ {
      data:    ▼[
                  ▼ {
                         balance:   integer
                         title:     string
                         budget:    integer
                         category:  string
                         geo_area:  string
                         positives: integer
                         negatives: integer
                         winner:    boolean
                       }
⇄                    ]
            summary: ▼ {
                       all of:
                         ▼ {
                           total:             integer
                           total_positives:   integer
                           total_negatives:   integer
                           total_users_voted: integer
                           start_date:        string
                           end_date:          string
                           total_balance:     integer
                         }
                     }
          }
    }
```

votesDailyReply

```
    ▼ votesDailyReply {
      all of:
        ▼ {
            data: ▼ {
                    all of:
                      ▼ {
                          date_value: ▼ {
                                        all of:
⇄                                         ▼ {
                                              positives: integer
                                              negatives: integer
                                            }
                                      }
                        }
                  }
          }
    }
```

votesCountByParamReply

```
▼ votesCountByParamReply {
  all of:
    ▼ {
      data: ▼ {
              all of:
                 ▼ {
                   parameter_value: ▼ {
                                      all of:
                                         ▼ {
                                           positives: integer
                                           negatives: integer
                                           total:     integer
                                         }
                                    }
                 }
            }
    }
}
```

votesByProfessionReply

```
▼ votesByProfessionReply {
  all of:
    ▼ {
      data:         ▼ [
                      ▼ {
                        title:       string
                        budget:      integer
                        category:    string
                        geo_area:    string
                        professions: ▼ {
                                       all of:
                                          ▼ {
                                            balance:   integer
                                            positives: integer
                                            negatives: integer
                                          }
                                     }
                      }
                    ]
      professions:  ▼ [
                      string
                    ]
    }
}
```

votesFirstByParamReply

```
▼votesFirstByParamReply {
  all of:
    ▼ {
      data: ▼ {
              all of:
                ▼ {
                  parameter_type: ▼ {
                                    all of:
                                      ▼ {
                                        positives: integer
                                        negatives: integer
                                      }
                                  }
                }
            }
    }
}
```

votesByChannelReply

```
▼votesByChannelReply {
  all of:
    ▼ {
      data: ▼[
              ▼ {
                title:    string
                budget:   integer
                category: string
                geo_area: string
                channels: ▼ {
                            all of:
                              ▼ {
                                balance:   integer
                                positives: integer
                                negatives: integer
                              }
                          }
              }
            ]
    }
}
```

votesByNbReply

```
  ▼ votesByNbReply {
    all of:
       ▼ {
          data: ▼ [
                  ▼ {
                     title:               string
                     budget:              integer
                     category:            string
                     geo_area:            string
                     neighbourhood_area: ▼ {
⇄                                          all of:
                                             ▼ {
                                                balance:   integer
                                                positives: integer
                                                negatives: integer
                                             }
                                         }
                  }
                ]
       }
  }
```

analyticsErrorDefault

```
  ▼ analyticsErrorDefault {
⇄    error: string
  }
```

topStatistics

```
  ▼ topStatistics {
⇄    top: integer
  }
```

parameterStatistics

```
  ▼ parameterStatistics {
⇄    parameter_key: string
  }
```

votesChannelReply

```
  ▼ votesChannelReply {
    all of:
       ▼ {
          in_person_votes: ▼ {
                              all of:
                                 ▼ {
                                    value: integer
```

```
                            }
                        }
          web_votes:        ▼ {
                              all of:
⇄                               ▼ {
                                  value: integer
                                }
                            }
          all_votes:        ▼ {
                              all of:
                                ▼ {
                                  value: integer
                                }
                            }
        }
    }
```

votesByTownReply

```
    ▼ votesByTownReply {
      all of:
        ▼ {
          data: ▼ {
                  all of:
                    ▼ {
                      in_person_votes: ▼ {
                                         all of:
                                           ▼ {
                                             town_name: integer
                                           }
                                       }
                      web_votes:       ▼ {
                                         all of:
⇄                                          ▼ {
                                             town_name: integer
                                           }
                                       }
                      all_votes:       ▼ {
                                         all of:
                                           ▼ {
                                             town_name: integer
                                           }
                                       }
                    }
                }
        }
}
```

voteByDateReply

```
▼ voteByDateReply {
  all of:
     ▼ {
       data: ▼ {
               all of:
                 ▼ {
                   negative_vote: boolean
                   total:          ▼ {
                                     all of:
                                       ▼ votesChannelReply {
                                         all of:
                                            ▼ {
                                              in_person_votes: ▼ {
                                                                 all of:
                                                                   ▼ {
                                                                      value:
                                                                   }
                                                               }
                                              web_votes:       ▼ {
                                                                 all of:
                                                                   ▼ {
                                                                      value:
                                                                   }
                                                               }
                                              all_votes:       ▼ {
                                                                 all of:
                                                                   ▼ {
                                                                      value:
                                                                   }
                                                               }
                                           }
                                       }
                                   }
                   balance:        ▼ {
                                     all of:
                                       ▼ votesChannelReply {
                                         all of:
                                            ▼ {
                                              in_person_votes: ▼ {
                                                                 all of:
                                                                   ▼ {
                                                                      value:
                                                                      }
                                                                  }
                                                web_votes:        ▼ {
                                                                   all of:
                                                                      ▼ {
                                                                        value:
                                                                      }
                                                                  }
                                                all_votes:        ▼ {
                                                                   all of:
                                                                      ▼ {
                                                                        value:
                                                                      }
                                                                  }
                                              }
⇄
                                         }
                                     }
                      positive:      ▼ {
                                       all of:
                                          ▼votesChannelReply {
                                           all of:
                                              ▼ {
                                                in_person_votes:  ▼ {
                                                                   all of:
                                                                      ▼ {
                                                                        value:
                                                                      }
                                                                  }
                                                web_votes:        ▼ {
                                                                   all of:
                                                                      ▼ {
                                                                        value:
                                                                      }
                                                                  }
                                                all_votes:        ▼ {
                                                                   all of:
                                                                      ▼ {
                                                                        value:
                                                                      }
                                                                  }
                                              }
                                         }
                                     }
                      negative:      ▼ {
                                       all of:
                                           ▼ votesChannelReply {
                                             all of:
                                               ▼ {
                                                 in_person_votes: ▼ {
                                                                    all of:
                                                                      ▼ {
                                                                        value:
                                                                      }
                                                                   }
                                                 web_votes:       ▼ {
                                                                    all of:
                                                                      ▼ {
                                                                        value:
                                                                      }
                                                                   }
                                                 all_votes:       ▼ {
                                                                    all of:
                                                                      ▼ {
                                                                        value:
                                                                      }
                                                                   }
                                               }
                                           }
                                      }
                     }
                }
        }
    }
```

votesByParameterTownReply

```
    ▼ votesByParameterTownReply {
      all of:
        ▼ {
          data: ▼ {
                  all of:
                     ▼ {
                       in_person_votes: ▼ {
                                          all of:
                                             ▼ {
                                               town: ▼ {
                                                       all of:
                                                          ▼ {
                                                            value: integer
 ⇄                                                       }
                                                     }
                                             }
```

```
                                        }
                       web_votes:       ▼ {
                                          all of:
                                             ▼ {
                                               value: integer
                                             }
                                        }
                     }
                }
        }
    }
```

voteByTopReply

```
    ▼ voteByTopReply {
      all of:
        ▼ {
          data: ▼ {
                  all of:
                     ▼ {
                       negative_vote: boolean
                       total:         ▼ [
                                        ▼ {
                                          position:    integer
                                          topic_name:  string
                                          total_votes: integer
                                        }
                                      ]
                       balance:       ▼ [
                                        ▼ {
                                          position:    integer
                                          topic_name:  string
                                          total_votes: integer
                                        }
                                      ]
                       positive:      ▼ [
                                        ▼ {
                                          position:    integer
                                          topic_name:  string
                                          total_votes: integer
                                        }
                                      ]
                       negative:      ▼ [
                                        ▼ {
                                          position:    integer
```

```
                              topic_name:  string
                              total_votes: integer
                            }
                          ]
                }
             }
        }
    }
```

voteByTopDateReply

```
▼ voteByTopDateReply {
  all of:
    ▼ {
      data: ▼ {
          all of:
            ▼ {
              negative_vote: boolean
              total:          ▼ [
                                 ▼ {
                                   topic_name: string
                                   votes:       ▼ {
                                                  all of:
                                                    ▼ {
                                                      date_value: integer
                                                    }
                                                }
                                 }
                               ]
              balance:        ▼ [
                                 ▼ {
                                   topic_name: string
                                   votes:       ▼ {
                                                  all of:
                                                    ▼ {
                                                      date_value: integer
                                                    }
                                                }
                                 }
                               ]
              positive:       ▼ [
                                 ▼ {
                                   topic_name: string
                                   votes:       ▼ {
                                                  all of:
```

```
                                                             ▼ {
                                                              date_value: integer
                                                            }
                                                        }
                                         }
                                       ]
                       negative:       ▼ [
                                          ▼ {
                                           topic_name: string
                                           votes:       ▼ {
                                                         all of:
                                                             ▼ {
                                                              date_value: integer
                                                            }
                                                        }
                                         }
                                       ]
                     }
                 }
        }
    }
```

voteByParameterReply

```
    ▼ voteByParameterReply {
      all of:
        ▼ {
          data: ▼ {
                   all of:
                      ▼ {
                       negative_vote: boolean
                       total:          ▼ [
                                          ▼ {
                                           value: integer
                                         }
                                       ]
                       balance:        ▼ [
                                          ▼ {
                                           value: integer
                                         }
                                       ]
                       positive:       ▼ [
                                          ▼ {
                                           value: integer
                                         }
                                       ]
                       negative:       ▼[
                                         ▼ {
                                           value: integer
                                         }
                                       ]
                     }
                 }
        }
    }
```

Auth Module

Version 1.0.0

Contact information
pvalente@onesource.pt (mailto:pvalente@onesource.pt)

Paths

/auth/authenticate

POST /auth/authenticate

User

Summary

Authentication of a user

Parameters

Name	Located in	Description	Required	Schema

Body	body	User login information	Yes	⇄ ▾**userAuthentication** {
				email: string *
				password: string *
				}

X-MODULE-TOKEN	header	Module Token	Yes	⇄ string
X-AUTH-TOKEN	header	Authentication Token	No	⇄ string

Responses

Code	Description	Schema

200	Authentication token for user	⇄ ▾**replyUserAuthentication** {
		token: string *
		}

401	Invalid Credentials	⇄ ▾**userErrorDefault** {
		error: string *
		}

500	Could not create token	⇄ ▾**userErrorDefault** {
		error: string *
		}

Try this operation

/auth/authenticateRFID

`POST /auth/authenticateRFID`

User

Summary

Authentication of a user by RFID

Parameters

Name	Located in	Description	Required	Schema

Body	body	User RFID code	Yes	⇄ ▾**`rfidAuthentication {`**
				`rfid: string *`
				`}`

X-MODULE-TOKEN	header	Module Token	Yes	⇄ `string`

Responses

Code	Description	Schema

200	Authentication token for user	⇄ ▾**`replyUserAuthentication {`**
		`token: string *`
		`}`

401	Invalid Credentials	⇄ ▾**`userErrorDefault {`**
		`error: string *`
		`}`

500	Could not create token	⇄ ▾**`userErrorDefault {`**
		`error: string *`
		`}`

Try this operation

`/auth/authenticateAlphanumeric`

POST /auth/authenticateAlphanumeric

User

Summary

Authentication of a user by an Alphanumeric code

Parameters

Name	Located in	Description	Required	Schema
Body	body	User Alphanumeric code	Yes	⇄ `undefined`
X-MODULE-TOKEN	header	Module Token	Yes	⇄ `string`

Responses

Code	Description	Schema

200	Authentication token for user	⇄ ▾**replyUserAuthentication {**
		`token: string *`
		}

401	Invalid Credentials	⇄ ▾**userErrorDefault {**
		`error: string *`
		}

500	Could not create token	⇄ ▾**userErrorDefault {**
		`error: string *`
		}

[Try this operation]

/auth/validate

`GET /auth/validate`

User

Summary

Validation of a user

Parameters

Name	Located in	Description	Required	Schema
X-MODULE-TOKEN	header	Module Token	Yes	⇄ `string`
X-AUTH-TOKEN	header	Authentication Token	Yes	⇄ `string`

Responses

Code	Description	Schema
200	User Key of the user	⇄
400	Token Absent	⇄
401	Token Expired	⇄

200

```
▾replyValidateUser {
   user_key: string *
}
```

400

```
▾userErrorDefault {
   error: string *
}
```

401

```
▾userErrorDefault {
   error: string *
}
```

Try this operation

`/user/{user_key}`

`GET /user/{user_key}`

User

Summary

Show a User

Parameters

Name	Located in	Description	Required	Schema
user_key	path	User Key	Yes	⇄ string
X-AUTH-TOKEN	header	User Auth Token	Yes	⇄ string
X-MODULE-TOKEN	header	Module Token	Yes	⇄ string

Responses

Code **Description** **Schema**

200 Show the User data ⇄

```
▾replyUser {
   data: ▾ {
                all of:
                    ▾User {
                      all of:
                        ▾ {
                             name:          string *
                             email:         string *
                             password:      string *
                             identity_card: string *
                             vat_number:    string *
                             birthday:      string *
                          }
                       }
                }
}
```

404 User not Found ⇄

```
▾userErrorDefault {
   error: string *
}
```

500 Failed to retrieve User ⇄

```
▾userErrorDefault {
   error: string *
}
```

☐ Try this operation

/auth

POST /auth

User

Summary

Creation of a user

Parameters

Name	Located in	Description	Required	Schema

Body	body	User registry information	Yes	⇄ ▾**createUser** {
				name: string *
				email: string *
				password: string *
				identity_card: string *
				vat_number: string *
				birthday: string *
				}

X-MODULE-TOKEN	header	Module Token	Yes	⇄ string

Responses

Code	Description	Schema

201	the newly created user	⇄ ▾**replyCreateUser** {
		confirmation_code: string *
		user: ▾ {
		}
		}

401	Invalid Credentials	⇄ ▾**userErrorDefault** {
		error: string *
		}

409	ID number already exists | Email already exists	⇄ ▾**userErrorDefault** {
		error: string *
		}

500	Failed to store new User	⇄ ▾**userErrorDefault** {
		error: string *
		}

Try this operation

/auth/{user_key}

PUT /auth/{user_key}

User

Summary

Update a user

Parameters

Name	Located in	Description	Required	Schema

Body	body	User update information	Yes	⇄ ▾**createUser** {
				name: string *
				email: string *
				password: string *
				identity_card: string *
				vat_number: string *
				birthday: string *
				}

user_key	path	User Key	Yes	⇄ `string`
X-AUTH-TOKEN	header	Authentication Token	Yes	⇄ `string`
X-MODULE-TOKEN	header	Module Token	Yes	⇄ `string`

Responses

Code	Description	Schema

200	the newly created user	⇄ ▾**replyCreateUser** {
		confirmation_code: string *
		user: ▾ {
		}
		}

403	Trying to update profile of different user	⇄ ▾**userErrorDefault** {
		error: string *
		}

404	User not Found	⇄ ▾**userErrorDefault** {
		error: string *
		}

409	ID number already exists | Email already exists	⇄ ▾**userErrorDefault** {
		error: string *
		}

500	error	⇄ ▾**userErrorDefault** {
		error: string *
		}

Try this operation

`DELETE /auth/{user_key}`

User

Summary

Delete a user

Parameters

Name	Located in	Description	Required	Schema
user_key	path	User Key	Yes	⇄ `string`
X-MODULE-TOKEN	header	Module Token	Yes	⇄ `string`
X-AUTH-TOKEN	header	Authentication Token	Yes	⇄ `string`

Responses

Code	Description	Schema

200	OK	⇄ ▾**replyDeleteAuthentication {**
		`string: string *`
		}

401	Unauthorized	⇄ ▾**userErrorDefault {**
		`error: string *`
		}

404	User not Found	⇄ ▾**userErrorDefault {**
		`error: string *`
		}

500	Failed to delete the user	⇄ ▾**userErrorDefault {**
		`error: string *`
		}

Try this operation

Models

userErrorDefault

```
   ▾userErrorDefault {
⇄    error: string *
   }
```

User

```
    ▾User {
      all of:
        ▾ {
          name:          string *
          email:         string *
⇄         password:      string *
          identity_card: string *
          vat_number:    string *
          birthday:      string *
        }
    }
```

userAuthentication

```
    ▾userAuthentication {
      email:    string *
⇄
      password: string *
    }
```

replyUserAuthentication

```
    ▾replyUserAuthentication {
⇄     token: string *
    }
```

rfidAuthentication

```
    ▾rfidAuthentication {
⇄     rfid: string *
    }
```

replyRfidAuthentication

```
    ▾replyRfidAuthentication {
⇄     token: string *
    }
```

alphanumericdAuthentication

```
    ▾alphanumericdAuthentication {
⇄     alphanumeric_code: string *
    }
```

replyAlphanumericAuthentication

```
    ▾replyAlphanumericAuthentication {
⇄     token: string *
    }
```

validateUser

```
⇄   ▾validateUser undefined
```

replyValidateUser

```
   ▾replyValidateUser {
⇄    user_key: string *
   }
```

replyUser

```
   ▾replyUser {
     data: ▾ {
                all of:
                   ▾User {
                     all of:
                        ▾ {
                          name:          string *
                          email:         string *
⇄                         password:      string *
                          identity_card: string *
                          vat_number:    string *
                          birthday:      string *
                        }
                   }
             }
   }
```

createUser

```
   ▾createUser {
     name:          string *
     email:         string *
     password:      string *
⇄    identity_card: string *
     vat_number:    string *
     birthday:      string *
   }
```

replyCreateUser

```
   ▾replyCreateUser {
     confirmation_code: string *
⇄    user:               ▾ {
                         }
   }
```

updateUser

```
   ▾updateUser {
     name:          string *
     email:         string *
     password:      string *
⇄    identity_card: string *
     vat_number:    string *
     birthday:      string *
   }
```

replyUpdateUser

```
   ▾replyUpdateUser {
      user: ▾ {
⇄
             }
   }
```

deleteUser

```
⇄  ▾deleteUser undefined
```

replyDeleteAuthentication

```
   ▾replyDeleteAuthentication {
⇄     string: string *
   }
```

CB Module

Version 1.0.0

Contact information
pvalente@onesource.pt (mailto:pvalente@onesource.pt)

Paths

/annotation/{key}

GET /annotation/{key}

Annotations

Summary

Show Annotations

Parameters

Name	Located in	Description	Required	Schema
key	path	Topic Key	Yes	⇄ string
X-MODULE-TOKEN	header	Module Token	Yes	⇄ string

Responses

Code **Description** **Schema**

200 Show Annotations data ⇄

```
▾annotationsResponse {
  all of:
    ▾ {
        annotation_key: string *
        text:           string
        quote:          string
      }
}
```

404 Annotations not Found ⇄

```
▾annotationsErrorDefault {
  error: string *
}
```

500 Failed to retrieve Annotations ⇄

```
▾annotationsErrorDefault {
  error: string *
}
```

Try this operation

PUT /annotation/{key}

Annotations

Summary

Update an Annotation

Parameters

<table>
<tr><th>Name</th><th>Located in</th><th>Description</th><th>Required</th><th>Schema</th></tr>
<tr><td>key</td><td>path</td><td>Annotation Key</td><td>Yes</td><td>⇄ string</td></tr>
<tr><td>Body</td><td>body</td><td>Annotation Update text</td><td>Yes</td><td>⇄<pre>
▾annotationsUpdate {
  all of:
    ▾ {
      text: string *
      tags: ▾[
              ▾ {
                code: string
              }
            ]
    }
}
</pre></td></tr>
<tr><td>X-AUTH-TOKEN</td><td>header</td><td>User Auth Token</td><td>Yes</td><td>⇄ string</td></tr>
<tr><td>X-MODULE-TOKEN</td><td>header</td><td>Module Token</td><td>Yes</td><td>⇄ string</td></tr>
</table>

Responses

<table>
<tr><th>Code</th><th>Description</th><th>Schema</th></tr>
<tr><td><b>200</b></td><td>The updated User</td><td>⇄<pre>
▾annotationsResponse {
  all of:
    ▾ {
      annotation_key: string *
      text: string
      quote: string
    }
}
</pre></td></tr>
<tr><td><b>401</b></td><td>Unauthorized</td><td>⇄<pre>
▾annotationsErrorDefault {
  error: string *
}
</pre></td></tr>
<tr><td><b>404</b></td><td>User not Found</td><td>⇄<pre>
▾annotationsErrorDefault {
  error: string *
}
</pre></td></tr>
<tr><td><b>500</b></td><td>Failed to update User</td><td>⇄<pre>
▾annotationsErrorDefault {
  error: string *
}
</pre></td></tr>
</table>

Try this operation

`DELETE /annotation/{key}`

Annotations

Summary

Delete an Annotation

Parameters

Name	Located in	Description	Required	Schema
key	path	Annotation Key	Yes	⇄ `string`
X-MODULE-TOKEN	header	Module Token	Yes	⇄ `string`
X-AUTH-TOKEN	header	Authentication Token	Yes	⇄ `string`
X-ENTITY-KEY	header	Entity Key	Yes	⇄ `string`

Responses

Code	Description	Schema

200	OK	⇄ ▾**replyDeleteAnnotation** {
		`string: string`
		}

401	Unauthorized	⇄ ▾**annotationsErrorDefault** {
		`error: string *`
		}

404	Annotation not Found	⇄ ▾**annotationsErrorDefault** {
		`error: string *`
		}

500	Failed to delete Annotation	⇄ ▾**annotationsErrorDefault** {
		`error: string *`
		}

Try this operation

`/annotation`

POST /annotation

Annotations

Summary

Creation of an Annotation

Parameters

Name	Located in	Description	Required	Schema
Body	body	Annotation data	Yes	⇄
X-AUTH-TOKEN	header	User Auth Token	Yes	⇄ string
X-MODULE-TOKEN	header	Module Token	Yes	⇄ string
X-ENTITY-KEY	header	Entity Key	Yes	⇄ string

Body:

```
▾annotations {
   all of:
     ▾ {
         text:    string *
         quote:   string *
         ranges:  ▸[]
         tags:    ▾[
                    ▾ {
                        code: string
                      }
                   ]
       }
 }
```

Responses

Code	Description	Schema
201	the newly created Annotation	⇄
401	Unauthorized	⇄
404	Unauthorized	⇄
409	Entity not Found	⇄
500	Failed to store new Annotation	⇄

201

```
▾annotationsResponse {
   all of:
     ▾ {
         annotation_key: string *
         text:           string
         quote:          string
       }
 }
```

401

```
▾annotationsErrorDefault {
   error: string *
 }
```

404

```
▾annotationsErrorDefault {
   error: string *
 }
```

409

```
▾annotationsErrorDefault {
   error: string *
 }
```

500

```
▾annotationsErrorDefault {
   error: string *
 }
```

Try this operation

/cb/{key}

GET /cb/{key}

Cbs

Summary

Show a Cb

Parameters

Name	Located in	Description	Required	Schema
key	path	Cb Key	Yes	⇄ `string`
X-MODULE-TOKEN	header	Module Token	Yes	⇄ `string`

Responses

Code **Description** **Schema**

200 Show the Cb data ⇄

```
▾cb {
  all of:
    ▾ {
      title:        string *
      contents:     string *
      created_by:   string *
      blocked:      string *
      status_id:    string *
      layout_code:  string *
      parent_cb_id: string *
      start_date:   boolean *
      end_date:     boolean *
    }
}
```

404 Cb not Found ⇄

```
▾cbErrorDefault {
  error: string *
}
```

500 Failed to retrieve Cb ⇄

```
▾cbErrorDefault {
  error: string *
}
```

Try this operation

PUT /cb/{key}

Cbs

Summary

Update a Cb

Parameters

Name	Located in	Description	Required	Schema
Body	body	Cb Update data	Yes	⇄

```
▾cb {
  all of:
    ▾ {
      title:         string
                     *
      contents:      string
                     *
      created_by:    string
                     *
      blocked:       string
                     *
      status_id:     string
                     *
      layout_code:   string
                     *
      parent_cb_id:  string
                     *
      start_date:    boolean
                     *
      end_date:      boolean
                     *
    }
}
```

Name	Located in	Description	Required	Schema
key	path	Cb Key	Yes	⇄ string
X-AUTH-TOKEN	header	User Auth Token	Yes	⇄ string
X-MODULE-TOKEN	header	Module Token	Yes	⇄ string

Responses

Code	Description	Schema

200 The updated Cb ⇄

```
▾cbResponse {
  all of:
    ▾ {
        cbKey:         string *
        title:         string *
        contents:      string *
        created_by:    string *
        blocked:       string *
        status_id:     string *
        layout_code:   string *
        parent_cb_id:  string *
        start_date:    boolean *
        end_date:      boolean *
      }
}
```

401 Unauthorized ⇄

```
▾cbErrorDefault {
   error: string *
}
```

404 Cb not Found ⇄

```
▾cbErrorDefault {
   error: string *
}
```

500 Failed to update Cb ⇄

```
▾cbErrorDefault {
   error: string *
}
```

Try this operation

`DELETE /cb/{key}`

Cbs

Summary

Delete a Cb

Parameters

Name	Located in	Description	Required	Schema
key	path	Cb Key	Yes	⇄ `string`
X-MODULE-TOKEN	header	Module Token	Yes	⇄ `string`
X-AUTH-TOKEN	header	Authentication Token	Yes	⇄ `string`
X-ENTITY-KEY	header	Entity Key	Yes	⇄ `string`

Responses

Code	Description		Schema
200	OK	⇄	▾ Object ▾ properties: Object ► string: Object title: "replyDeleteCb"

401	Unauthorized	⇄	▾ **cbErrorDefault {**
			error: string *
			}

404	Cb not Found	⇄	▾ **cbErrorDefault {**
			error: string *
			}

500	Failed to delete Cb	⇄	▾ **cbErrorDefault {**
			error: string *
			}

Try this operation

`/cb`

POST /cb

Cbs

Summary

Creation of a Cb

Parameters

Name	Located in	Description	Required	Schema

Body	body	Cb data	Yes	⇄ ▾cb {
				all of:
				▾ {
				title: string *
				contents: string *
				created_by: string *
				blocked: string *
				status_id: string *
				layout_code: string *
				parent_cb_id: string *
				start_date: boolean *
				end_date: boolean *
				}
				}

X-AUTH-TOKEN	header	User Auth Token	Yes	⇄ string
X-MODULE-TOKEN	header	Module Token	Yes	⇄ string
X-ENTITY-KEY	header	Entity Key	Yes	⇄ string

Responses

Code	Description	Schema

201 the newly created Cb ⇄

```
▾cb {
  all of:
    ▾ {
      title:        string *
      contents:     string *
      created_by:   string *
      blocked:      string *
      status_id:    string *
      layout_code:  string *
      parent_cb_id: string *
      start_date:   boolean *
      end_date:     boolean *
    }
}
```

401 Unauthorized ⇄

```
▾cbErrorDefault {
  error: string *
}
```

500 Failed to store new Cb ⇄

```
▾cbErrorDefault {
  error: string *
}
```

Try this operation

/comments/{commentKey}

GET /comments/{commentKey}

Comments

Summary

Show a Comment

Parameters

Name	Located in	Description	Required	Schema
commentKey	path	Comment Key	Yes	⇄ string
timezone	header	Timezone	Yes	⇄ string
X-MODULE-TOKEN	header	Module Token	Yes	⇄ string

Responses

Code Description Schema

200 Show the Comment data ⇄

```
▾commentsResponse {
  all of:
    ▾ {
        comment_key: string *
        title:       string *
        content:     string *
      }
}
```

404 Comment not Found ⇄

```
▾commentErrorDefault {
   error: string *
}
```

500 Failed to retrieve Comment ⇄

```
▾commentErrorDefault {
   error: string *
}
```

Try this operation

PUT /comments/{commentKey}

Comments

Summary

Update a Comment

Parameters

Name	Located in	Description	Required	Schema
Body	body	Comment Update data	Yes	⇄ commentsUpdate
commentKey	path	Comment Key	Yes	⇄ string
X-AUTH-TOKEN	header	User Auth Token	Yes	⇄ string
X-MODULE-TOKEN	header	Module Token	Yes	⇄ string

```
▾commentsUpdate {
  all of:
    ▾ {
      public:  string
               *
      title:   string
               *
      content: string
               *
    }
}
```

Responses

Code	Description	Schema
200	The updated Comment	⇄ commentsResponse
401	Unauthorized	⇄ commentErrorDefault
404	Comment not Found	⇄ commentErrorDefault
500	Failed to update Comment	⇄ commentErrorDefault

```
▾commentsResponse {
  all of:
    ▾ {
      comment_key: string *
      title:       string *
      content:     string *
    }
}
```

```
▾commentErrorDefault {
  error: string *
}
```

```
▾commentErrorDefault {
  error: string *
}
```

```
▾commentErrorDefault {
  error: string *
}
```

Try this operation

DELETE /comments/{commentKey}

Comments

Summary

Delete a Comment

Parameters

Name	Located in	Description	Required	Schema
commentKey	path	Comment Key	Yes	⇄ `string`
X-MODULE-TOKEN	header	Module Token	Yes	⇄ `string`
X-AUTH-TOKEN	header	Authentication Token	Yes	⇄ `string`
X-ENTITY-KEY	header	Entity Key	Yes	⇄ `string`

Responses

Code	Description	Schema

200	OK	⇄ ▾**replyDeleteComment** {
		string: string
		}

401	Unauthorized	⇄ ▾**commentErrorDefault** {
		error: string *
		}

404	Comment not Found	⇄ ▾**commentErrorDefault** {
		error: string *
		}

500	Failed to delete Comment	⇄ ▾**commentErrorDefault** {
		error: string *
		}

Try this operation

/comments

POST /comments

Comments

Summary

Creation of a Comment

Parameters

Name	Located in	Description	Required	Schema
Body	body	Comment data	Yes	⇄

```
▾comments {
  all of:
    ▾ {
      public:     string
                  *
      title:      string
                  *
      content:    string
                  *
      created_by: string
                  *
    }
}
```

Name	Located in	Description	Required	Schema
X-AUTH-TOKEN	header	User Auth Token	Yes	⇄ `string`
X-MODULE-TOKEN	header	Module Token	Yes	⇄ `string`
X-ENTITY-KEY	header	Entity Key	Yes	⇄ `string`

Responses

Code	Description	Schema
201	the newly created user	⇄

```
▾commentsResponse {
  all of:
    ▾ {
      comment_key: string *
      title:       string *
      content:     string *
    }
}
```

Code	Description	Schema
404	Unauthorized	⇄

```
▾commentErrorDefault {
  error: string *
}
```

Code	Description	Schema
500	Failed to store new Comment	⇄

```
▾commentErrorDefault {
  error: string *
}
```

Try this operation

/configurationType/{id}

GET /configurationType/{id}

ConfigurationType

Summary

Show a Configuration Type

Parameters

Name	Located in	Description	Required	Schema
id	path	Configuration Type Id	Yes	⇄ string
X-MODULE-TOKEN	header	Module Token	Yes	⇄ string

Responses

Code	Description	Schema
200	Show the Configuration Type data	⇄
404	Configuration Type not Found	⇄
500	Failed to retrieve Configuration Type	⇄

200 Show the Configuration Type data

```
▾configurationTypeResponse {
   all of:
     ▾ {
        code:          string *
        created_by:    string *
        created_at:    string *
        updated_at:    string *
        translations: ▾[
                        ▾ {
                          configuration_type_id: integer
                          language_code:         string
                          title:                 string
                          description:           string
                          created_at:            string
                          updated_at:            string
                        }
                      ]
     }
}
```

404 Configuration Type not Found

```
▾configurationTypeErrorDefault {
   error: string *
}
```

500 Failed to retrieve Configuration Type

```
▾configurationTypeErrorDefault {
   error: string *
}
```

Try this operation

PUT /configurationType/{id}

ConfigurationType

Summary

Update a User

Parameters

Name	Located in	Description	Required	Schema
Body	body	Configuration Type Update data	Yes	⇄
id	path	COnfiguration Type Id	Yes	⇄ string
X-AUTH-TOKEN	header	User Auth Token	Yes	⇄ string
X-MODULE-TOKEN	header	Module Token	Yes	⇄ string

```
▾configurationType {
  all of:
    ▾ {
      code:          string
                     *
      translations:  ▸[]
    }
}
```

Responses

Code	Description	Schema
200	The updated Configuration Type	⇄
401	Unauthorized	⇄
404	User not Found	⇄

```
▾configurationTypeResponse {
  all of:
    ▾ {
      code:          string *
      created_by:    string *
      created_at:    string *
      updated_at:    string *
      translations:  ▾[
                       ▾ {
                         configuration_type_id: integer
                         language_code:         string
                         title:                 string
                         description:           string
                         created_at:            string
                         updated_at:            string
                       }
                     ]
    }
}
```

```
▾configurationTypeErrorDefault {
  error: string *
}
```

```
▾configurationTypeErrorDefault {
  error: string *
}
```

Code	Description	Schema

500	Failed to update Configuration Type	⇄ ▾**configurationTypeErrorDefault {**
		`error: string *`
		`}`

Try this operation

DELETE /configurationType/{id}

ConfigurationType

Summary

Delete a Configuration Types

Parameters

Name	Located in	Description	Required	Schema
id	path	Configuration Type Id	Yes	⇄ `string`
X-MODULE-TOKEN	header	Module Token	Yes	⇄ `string`
X-AUTH-TOKEN	header	Authentication Token	Yes	⇄ `string`
X-ENTITY-KEY	header	Entity Key	Yes	⇄ `string`

Responses

Code	Description	Schema

200	OK	⇄ ▾**replyDeleteConfigurationType {**
		`string: string`
		`}`

401	Unauthorized	⇄ ▾**configurationTypeErrorDefault {**
		`error: string *`
		`}`

404	Configuration Type not Found	⇄ ▾**configurationTypeErrorDefault {**
		`error: string *`
		`}`

500	Failed to delete Configuration Type	⇄ ▾**configurationTypeErrorDefault {**
		`error: string *`
		`}`

Try this operation

/configurations/{id}

GET /configurations/{id}

Configuration

Summary

Show a Configuration

Parameters

Name	Located in	Description	Required	Schema
id	path	Configuration Id	Yes	⇄ string
X-MODULE-TOKEN	header	Module Token	Yes	⇄ string
LANG_CODE	header	Lang Code	Yes	⇄ string

Responses

Code **Description** **Schema**

200 Show the Configuration data ⇄

```
▾configurationResponse {
  all of:
     ▾ {
       code: string *
     }
}
```

404 Configuration not Found ⇄

```
▾configurationErrorDefault {
  error: string *
}
```

500 Failed to retrieve Configuration ⇄

```
▾configurationErrorDefault {
  error: string *
}
```

Try this operation

PUT /configurations/{id}

Configuration

Summary

Update a Configuration

Parameters

Name	Located in	Description	Required	Schema
Body	body	Configuration Update data	Yes	⇄
id	path	Configuration Id	Yes	⇄ string
X-AUTH-TOKEN	header	User Auth Token	Yes	⇄ string
X-MODULE-TOKEN	header	Module Token	Yes	⇄ string

Body:

```
▾configurations {
  all of:
    ▾ {
      configuration_type_id: string
                             *
      code:                  string
                             *
      created_by:            string
                             *
      translations:          ▸[]
    }
}
```

Responses

Code	Description	Schema
200	The updated Configuration	⇄
401	Unauthorized	⇄
404	Configuration not Found	⇄
500	Failed to update Configuration	⇄

200:

```
▾configurationResponse {
  all of:
    ▾ {
      code: string *
    }
}
```

401:

```
▾configurationErrorDefault {
  error: string *
}
```

404:

```
▾configurationErrorDefault {
  error: string *
}
```

500:

```
▾configurationErrorDefault {
  error: string *
}
```

Try this operation

DELETE /configurations/{id}

Configuration

Summary

Delete a Configuration

Parameters

Name	Located in	Description	Required	Schema
id	path	Configuration Id	Yes	⇄ string
X-MODULE-TOKEN	header	Module Token	Yes	⇄ string
X-AUTH-TOKEN	header	Authentication Token	Yes	⇄ string
X-ENTITY-KEY	header	Entity Key	Yes	⇄ string

Responses

Code	Description	Schema

200	OK	⇄ ▾**replyDeleteConfiguration** {
		string: string
		}

401	Unauthorized	⇄ ▾**configurationErrorDefault** {
		error: string *
		}

404	Configuration not Found	⇄ ▾**configurationErrorDefault** {
		error: string *
		}

500	Failed to delete Configuration	⇄ ▾**configurationErrorDefault** {
		error: string *
		}

Try this operation

/configurations

POST /configurations

Configuration

Summary

Creation of a Configuration

Parameters

Name	Located in	Description	Required	Schema
Body	body	Configuration data	Yes	⇄

```
▾configurations {
 all of:
   ▾ {
     configuration_type_id: string
                            *
     code:                  string
                            *
     created_by:            string
                            *
     translations:          ▸[]
   }
}
```

Name	Located in	Description	Required	Schema
X-AUTH-TOKEN	header	User Auth Token	Yes	⇄ string
X-MODULE-TOKEN	header	Module Token	Yes	⇄ string
X-ENTITY-KEY	header	Entity Key	Yes	⇄ string

Responses

Code	Description	Schema
201	the newly created Configuration	⇄

```
▾configurationResponse {
 all of:
   ▾ {
     code: string *
   }
}
```

Code	Description	Schema
401	Unauthorized	⇄

```
▾configurationErrorDefault {
  error: string *
}
```

Code	Description	Schema
404	Unauthorized	⇄

```
▾configurationErrorDefault {
  error: string *
}
```

Code	Description	Schema
409	Entity not Found	⇄

```
▾configurationErrorDefault {
  error: string *
}
```

Code	Description	Schema
500	Failed to store new Configuration	⇄

```
▾configurationErrorDefault {
  error: string *
}
```

Try this operation

/parameterTemplates/{parameter_id}

GET /parameterTemplates/{parameter_id}

ParameterTemplate

Summary

Show a Parameter

Parameters

Name	Located in	Description	Required	Schema
parameter_id	path	Parameter Id	Yes	⇄ string
X-MODULE-TOKEN	header	Module Token	Yes	⇄ string

Responses

Code **Description** **Schema**

200 Show the Parameter data ⇄

```
▾parameterTemplateResponse {
  all of:
    ▾ {
      parameter_template_key: string *
      code:                   string *
      mandatory:              integer *
      parameter:              string
      description:            string
      value:                  string *
      currency:               string *
      position:               integer *
      use_filter:             integer *
      visible_in_list:        integer *
      visible:                integer *
    }
}
```

404 Parameter not Found ⇄

```
▾parameterTemplateErrorDefault {
  error: string *
}
```

500 Failed to retrieve Parameter ⇄

```
▾parameterTemplateErrorDefault {
  error: string *
}
```

Try this operation

/parameterTemplates

POST /parameterTemplates

ParameterTemplate

Summary

Creation of a Parameter Template

Parameters

<table>
<tr><th>Name</th><th>Located in</th><th>Description</th><th>Required</th><th>Schema</th></tr>
<tr><td>Body</td><td>body</td><td>Parameter Template data</td><td>Yes</td><td>⇄
<pre>
▾parameterTemplate {
  all of:
    ▾ {
      parameter_type_id: integer
                         *
      code: string
                         *
      mandatory: integer
                         *
      parameter: string
      description: string
      value: string
      currency: string
      position: integer
                         *
      use_filter: integer
                         *
      visible_in_list: integer
                         *
      visible: integer
                         *
    }
}
</pre></td></tr>
<tr><td>X-AUTH-TOKEN</td><td>header</td><td>User Auth Token</td><td>Yes</td><td>⇄ string</td></tr>
<tr><td>X-MODULE-TOKEN</td><td>header</td><td>Module Token</td><td>Yes</td><td>⇄ string</td></tr>
<tr><td>X-ENTITY-KEY</td><td>header</td><td>Entity Key</td><td>Yes</td><td>⇄ string</td></tr>
</table>

Responses

Code	Description	Schema

201 the newly created parameter ⇄

```
▾parameterTemplateResponse {
  all of:
    ▾ {
        parameter_template_key: string
                                *
        code:                   string
                                *
        mandatory:              integer
                                *
        parameter:              string
        description:            string
        value:                  string
                                *
        currency:               string
                                *
        position:               integer
                                *
        use_filter:             integer
                                *
        visible_in_list:        integer
                                *
        visible:                integer
                                *
      }
}
```

401 Unauthorized ⇄

```
▾parameterTemplateErrorDefault {
  error: string *
}
```

404 Unauthorized ⇄

```
▾parameterTemplateErrorDefault {
  error: string *
}
```

409 Entity not Found ⇄

```
▾parameterTemplateErrorDefault {
  error: string *
}
```

500 Failed to store new Parameter Template ⇄

```
▾parameterTemplateErrorDefault {
  error: string *
}
```

Try this operation

/parameterTemplate/{parameterId}

`PUT /parameterTemplate/{parameterId}`

ParameterTemplate

Summary

Update a Parameter

Parameters

Name	Located in	Description	Required	Schema
Body	body	Parameter Template Update data	Yes	⇄
parameterId	path	Parameter Template Key	Yes	⇄ `string`
X-AUTH-TOKEN	header	User Auth Token	Yes	⇄ `string`
X-MODULE-TOKEN	header	Module Token	Yes	⇄ `string`

```
▾parameterTemplate {
  all of:
     ▾ {
        parameter_type_id: integer
                           *
        code:              string
                           *
        mandatory:         integer
                           *
        parameter:         string
        description:       string
        value:             string
        currency:          string
        position:          integer
                           *
        use_filter:        integer
                           *
        visible_in_list:   integer
                           *
        visible:           integer
                           *
     }
}
```

Responses

Code	Description	Schema

200 The updated Parameter Template ⇄

```
▾parameterTemplateResponse {
 all of:
    ▾ {
     parameter_template_key: string
                             *
     code:                   string
                             *
     mandatory:              integer
                             *
     parameter:              string
     description:            string
     value:                  string
                             *
     currency:               string
                             *
     position:               integer
                             *
     use_filter:             integer
                             *
     visible_in_list:        integer
                             *
     visible:                integer
                             *
    }
}
```

401 Unauthorized ⇄

```
▾parameterTemplateErrorDefault {
  error: string *
}
```

404 Parameter Template not Found ⇄

```
▾parameterTemplateErrorDefault {
  error: string *
}
```

500 Failed to update Parameter Template ⇄

```
▾parameterTemplateErrorDefault {
  error: string *
}
```

Try this operation

DELETE /parameterTemplate/{parameterId}

ParameterTemplate

Summary

Delete a Parameter

Parameters

Name	Located in	Description	Required	Schema
parameterId	path	Parameter Key	Yes	⇄ string
X-MODULE-TOKEN	header	Module Token	Yes	⇄ string
X-AUTH-TOKEN	header	Authentication Token	Yes	⇄ string
X-ENTITY-KEY	header	Entity Key	Yes	⇄ string

Responses

Code	Description	Schema

200	OK	⇄ ▾**replyDeleteParameterTemplate {**
		string: string
		}

401	Unauthorized	⇄ ▾**parameterErrorDefault {**
		error: string *
		}

404	Parameter not Found	⇄ ▾**parameterErrorDefault {**
		error: string *
		}

500	Failed to delete Parameter	⇄ ▾**parameterErrorDefault {**
		error: string *
		}

Try this operation

/parameterTypes/{typeId}

`GET /parameterTypes/{typeId}`

ParameterType

Summary

Show a Parameter Type

Parameters

Name	Located in	Description	Required	Schema
typeId	path	Type Id	Yes	⇄ `string`
X-MODULE-TOKEN	header	Module Token	Yes	⇄ `string`

Responses

Code	Description	Schema
200	Show the Parameter Type data	⇄
404	Parameter Type not Found	⇄
500	Failed to retrieve Parameter Type	⇄

200

```
▾parameterType {
  all of:
    ▾ {
        name: string *
        code: string *
    }
}
```

404

```
▾parameterTypeErrorDefault {
  error: string *
}
```

500

```
▾parameterTypeErrorDefault {
  error: string *
}
```

Try this operation

`PUT /parameterTypes/{typeId}`

ParameterType

Summary

Update a User

Parameters

Name	Located in	Description	Required	Schema

Body	body	Parameter Type Update data	Yes	⇄ ▾parameterType {
				all of:
				▾ {
				name: string *
				code: string *
				}
				}

typeId	path	Parameter Type Id	Yes	⇄ string
X-AUTH-TOKEN	header	User Auth Token	Yes	⇄ string
X-MODULE-TOKEN	header	Module Token	Yes	⇄ string

Responses

Code	Description	Schema

200	The updated parameterType	⇄ ▾parameterType {
		all of:
		▾ {
		name: string *
		code: string *
		}
		}

401	Unauthorized	⇄ ▾parameterTypeErrorDefault {
		error: string *
		}

404	Parameter Type not Found	⇄ ▾parameterTypeErrorDefault {
		error: string *
		}

500	Failed to update Parameter Type	⇄ ▾parameterTypeErrorDefault {
		error: string *
		}

Try this operation

`DELETE /parameterTypes/{typeId}`

ParameterType

Summary

Delete a Parameter Type

Parameters

Name	Located in	Description	Required	Schema
typeId	path	Parameter Type Id	Yes	⇄ `string`
X-MODULE-TOKEN	header	Module Token	Yes	⇄ `string`
X-AUTH-TOKEN	header	Authentication Token	Yes	⇄ `string`
X-ENTITY-KEY	header	Entity Key	Yes	⇄ `string`

Responses

Code	Description	Schema

200	OK	⇄ ▾**replyDeleteParameterType** {
		`string: string`
		}

401	Unauthorized	⇄ ▾**parameterTypeErrorDefault** {
		`error: string *`
		}

404	Parameter Type not Found	⇄ ▾**parameterTypeErrorDefault** {
		`error: string *`
		}

500	Failed to delete Parameter Type	⇄ ▾**parameterTypeErrorDefault** {
		`error: string *`
		}

Try this operation

`/parameterTypes`

POST /parameterTypes

ParameterType

Summary

Creation of a Parameter Type

Parameters

Name	Located in	Description	Required	Schema

Body	body	Parameter Type data	Yes	⇄ ▾parameterType {
				all of:
				▾ {
				name: string *
				code: string *
				}
				}

X-AUTH-TOKEN	header	User Auth Token	Yes	⇄ string
X-MODULE-TOKEN	header	Module Token	Yes	⇄ string
X-ENTITY-KEY	header	Entity Key	Yes	⇄ string

Responses

Code	Description	Schema

201	the newly created parameter type	⇄ ▾parameterType {
		all of:
		▾ {
		name: string *
		code: string *
		}
		}

401	Unauthorized	⇄ ▾parameterTypeErrorDefault {
		error: string *
		}

404	Unauthorized	⇄ ▾parameterTypeErrorDefault {
		error: string *
		}

409	Entity not Found	⇄ ▾parameterTypeErrorDefault {
		error: string *
		}

500	Failed to store new Parameter Type	⇄ ▾parameterTypeErrorDefault {
		error: string *
		}

Try this operation

/parameters/{parameterId}

`GET /parameters/{parameterId}`

Parameter

Summary

Show a Parameter

Parameters

Name	Located in	Description	Required	Schema
parameterId	path	Parameter Id	Yes	⇄ `string`
X-MODULE-TOKEN	header	Module Token	Yes	⇄ `string`

Responses

Code **Description** **Schema**

200 Show the Parameter data ⇄

```
▾parameterResponse {
  all of:
     ▾ {
        code:            string *
        mandatory:       integer *
        parameter:       string
        description:     string
        value:           string *
        currency:        string *
        position:        integer *
        use_filter:      integer *
        visible_in_list: integer *
        visible:         integer *
     }
}
```

404 Parameter not Found ⇄

```
▾parameterErrorDefault {
  error: string *
}
```

500 Failed to retrieve Parameter ⇄

```
▾parameterErrorDefault {
  error: string *
}
```

Try this operation

PUT /parameters/{parameterId}

Parameter

Summary

Update a Parameter

Parameters

Name	Located in	Description	Required	Schema
Body	body	Parameter Update data	Yes	⇄
parameterId	path	Parameter Id	Yes	⇄ string
X-AUTH-TOKEN	header	User Auth Token	Yes	⇄ string
X-MODULE-TOKEN	header	Module Token	Yes	⇄ string

```
▾parameter {
  all of:
    ▾ {
      parameter_type_id: integer
                         *
      code:              string
                         *
      mandatory:         integer
                         *
      parameter:         string
      description:       string
      value:             string
      currency:          string
      position:          integer
                         *
      use_filter:        integer
                         *
      visible_in_list:   integer
                         *
      visible:           integer
                         *
    }
}
```

Responses

Code	Description	Schema

200 The updated Parameter ⇄

```
▾parameterResponse {
  all of:
    ▾ {
       code:             string *
       mandatory:        integer *
       parameter:        string
       description:      string
       value:            string *
       currency:         string *
       position:         integer *
       use_filter:       integer *
       visible_in_list:  integer *
       visible:          integer *
     }
}
```

401 Unauthorized ⇄

```
▾parameterErrorDefault {
  error: string *
}
```

404 Parameter not Found ⇄

```
▾parameterErrorDefault {
  error: string *
}
```

500 Failed to update Parameter ⇄

```
▾parameterErrorDefault {
  error: string *
}
```

Try this operation

`DELETE /parameters/{parameterId}`

Parameter

Summary

Delete a Parameter

Parameters

Name	Located in	Description	Required	Schema
parameterId	path	Parameter Id	Yes	⇄ `string`
X-MODULE-TOKEN	header	Module Token	Yes	⇄ `string`
X-AUTH-TOKEN	header	Authentication Token	Yes	⇄ `string`
X-ENTITY-KEY	header	Entity Key	Yes	⇄ `string`

Responses

Code	Description	Schema

200	OK	⇄ ▾**replyDeleteParameter {**
		`string: string`
		}

401	Unauthorized	⇄ ▾**parameterErrorDefault {**
		`error: string *`
		}

404	Parameter not Found	⇄ ▾**parameterErrorDefault {**
		`error: string *`
		}

500	Failed to delete Parameter	⇄ ▾**parameterErrorDefault {**
		`error: string *`
		}

Try this operation

`/parameters`

POST /parameters

Parameter

Summary

Creation of a Parameter

Parameters

Name	Located in	Description	Required	Schema
Body	body	Parameter data	Yes	⇄
X-AUTH-TOKEN	header	User Auth Token	Yes	⇄ string
X-MODULE-TOKEN	header	Module Token	Yes	⇄ string
X-ENTITY-KEY	header	Entity Key	Yes	⇄ string

Body:

```
▾parameter {
  all of:
     ▾ {
       parameter_type_id: integer
                          *
       code:              string
                          *
       mandatory:         integer
                          *
       parameter:         string
       description:       string
       value:             string
       currency:          string
       position:          integer
                          *
       use_filter:        integer
                          *
       visible_in_list:   integer
                          *
       visible:           integer
                          *
     }
}
```

Responses

Code	Description	Schema

201 the newly created parameter ⇄

```
▾parameterResponse {
  all of:
    ▾ {
        code:            string *
        mandatory:       integer *
        parameter:       string
        description:     string
        value:           string *
        currency:        string *
        position:        integer *
        use_filter:      integer *
        visible_in_list: integer *
        visible:         integer *
      }
}
```

401 Unauthorized ⇄

```
▾parameterErrorDefault {
  error: string *
}
```

404 Unauthorized ⇄

```
▾parameterErrorDefault {
  error: string *
}
```

409 Entity not Found ⇄

```
▾parameterErrorDefault {
  error: string *
}
```

500 Failed to store new Parameter ⇄

```
▾parameterErrorDefault {
  error: string *
}
```

Try this operation

/postabuse/{id}

GET /postabuse/{id}

PostAbuse

Summary

Show a Post Abuse

Parameters

Name	Located in	Description	Required	Schema
id	path	Post Abuse Id	Yes	⇄ string
X-MODULE-TOKEN	header	Module Token	Yes	⇄ string

Responses

Code **Description** **Schema**

200 Show the Site data ⇄

```
▾postAbuseResponse {
  all of:
    ▾ {
        id:         integer *
        post_id:    integer *
        type_id:    integer *
        comment:    string *
        created_by: string *
        processed:  integer *
        created_at: string *
        updated_at: string *
      }
}
```

404 Site not Found ⇄

```
▾postAbuseErrorDefault {
  error: string *
}
```

500 Failed to retrieve Site ⇄

```
▾postAbuseErrorDefault {
  error: string *
}
```

Try this operation

PUT /postabuse/{id}

PostAbuse

Summary

Update a Post Abuse

Parameters

Name	Located in	Description	Required	Schema
Body	body	Post Abuse Update data	Yes	⇄ postAbuse
id	path	Post Abuse Id	Yes	⇄ string
X-AUTH-TOKEN	header	User Auth Token	Yes	⇄ string
X-MODULE-TOKEN	header	Module Token	Yes	⇄ string

Body:

```
▾postAbuse {
   all of:
     ▾ {
         post_key:   string
                     *
         type_id:    integer
                     *
         comment:    string
                     *
         created_by: string
                     *
     }
 }
```

Responses

Code	Description	Schema
200	The updated Post Abuse	⇄ postAbuseResponse
401	Unauthorized	⇄ postAbuseErrorDefault
404	Post Abuse not Found	⇄ postAbuseErrorDefault
500	Failed to update Post Abuse	⇄ postAbuseErrorDefault

200:

```
▾postAbuseResponse {
   all of:
     ▾ {
         id:         integer *
         post_id:    integer *
         type_id:    integer *
         comment:    string *
         created_by: string *
         processed:  integer *
         created_at: string *
         updated_at: string *
     }
 }
```

401:

```
▾postAbuseErrorDefault {
   error: string *
 }
```

404:

```
▾postAbuseErrorDefault {
   error: string *
 }
```

500:

```
▾postAbuseErrorDefault {
   error: string *
 }
```

Try this operation

DELETE /postabuse/{id}

PostAbuse

Summary

Delete a Post Abuse

Parameters

Name	Located in	Description	Required	Schema
id	path	Post Abuse Id	Yes	⇄ string
X-MODULE-TOKEN	header	Module Token	Yes	⇄ string
X-AUTH-TOKEN	header	Authentication Token	Yes	⇄ string
X-ENTITY-KEY	header	Entity Key	Yes	⇄ string

Responses

Code	Description	Schema

200	OK	⇄ ▾**replyDeletePostAbuse** {
		string: string
		}

401	Unauthorized	⇄ ▾**postAbuseErrorDefault** {
		error: string *
		}

404	Post Abuse not Found	⇄ ▾**postAbuseErrorDefault** {
		error: string *
		}

500	Failed to delete Post Abuse	⇄ ▾**postAbuseErrorDefault** {
		error: string *
		}

Try this operation

/postabuse

POST /postabuse

PostAbuse

Summary

Creation of a Post Abuse

Parameters

Name	Located in	Description	Required	Schema
Body	body	Post Abuse data	Yes	⇄
X-AUTH-TOKEN	header	User Auth Token	Yes	⇄ string
X-MODULE-TOKEN	header	Module Token	Yes	⇄ string
X-ENTITY-KEY	header	Entity Key	Yes	⇄ string

```
▾postAbuse {
  all of:
    ▾ {
      post_key:   string
                  *
      type_id:    integer
                  *
      comment:    string
                  *
      created_by: string
                  *
    }
}
```

Responses

Code	Description	Schema
201	the newly created post abuse	⇄
401	Unauthorized	⇄
404	Unauthorized	⇄

```
▾postAbuseResponse {
  all of:
    ▾ {
      id:         integer *
      post_id:    integer *
      type_id:    integer *
      comment:    string *
      created_by: string *
      processed:  integer *
      created_at: string *
      updated_at: string *
    }
}
```

```
▾postAbuseErrorDefault {
  error: string *
}
```

```
▾postAbuseErrorDefault {
  error: string *
}
```

Code	Description	Schema

409	Entity not Found	⇄ ▾**postAbuseErrorDefault {**
		error: string *
		}

500	Failed to store new Post Abuse	⇄ ▾**postAbuseErrorDefault {**
		error: string *
		}

Try this operation

/postCommentTypes/{postCommentTypeKey}

GET /postCommentTypes/{postCommentTypeKey}

PostCommentType

Summary

Show a Post Comment Type

Parameters

Name	Located in	Description	Required	Schema
postCommentTypeKey	path	Post Comment Type Key	Yes	⇄ string
X-MODULE-TOKEN	header	Module Token	Yes	⇄ string

Responses

Code	Description	Schema
200	Show the Site data	⇄ ▾**postCommentTypeResponse {** **all of:** ▾ **{** post_comment_type_key: string * code: string * created_at: string * updated_at: string * } }

404	Post Comment Type not Found	⇄ ▾**postCommentTypeErrorDefault {**
		error: string *
		}

500	Failed to retrieve Post Comment Type	⇄ ▾**postCommentTypeErrorDefault {**
		error: string *
		}

Try this operation

/postCommentType

POST /postCommentType

PostCommentType

Summary

Creation of a Post Comment Type

Parameters

Name	Located in	Description	Required	Schema

Body	body	Post Comment Type data	Yes	⇄ ▾postCommentType {
				all of:
				▾ {
				code: string *
				}
				}

X-AUTH-TOKEN	header	User Auth Token	Yes	⇄ string
X-MODULE-TOKEN	header	Module Token	Yes	⇄ string
X-ENTITY-KEY	header	Entity Key	Yes	⇄ string

Responses

Code	Description	Schema

201	the newly created post comment type	⇄ ▾postCommentTypeResponse {
		all of:
		▾ {
		post_comment_type_key: string *
		code: string *
		created_at: string *
		updated_at: string *
		}
		}

401	Unauthorized	⇄ ▾postCommentTypeErrorDefault {
		error: string *
		}

404	Unauthorized	⇄ ▾postCommentTypeErrorDefault {
		error: string *
		}

409	Entity not Found	⇄ ▾postCommentTypeErrorDefault {
		error: string *
		}

500	Failed to store new Post Comment Type	⇄ ▾postCommentTypeErrorDefault {
		error: string *
		}

Try this operation

/postCommentType/{postCommentTypeKey}

PUT /postCommentType/{postCommentTypeKey}

PostCommentType

Summary

Update a Post Comment Type

Parameters

Name	Located in	Description	Required	Schema
Body	body	Post Comment Type Update data	Yes	⇄ `▾postCommentType { all of: ▾ { code: string * } }`
postCommentTypeKey	path	Post Comment Type Key	Yes	⇄ `string`
X-AUTH-TOKEN	header	User Auth Token	Yes	⇄ `string`
X-MODULE-TOKEN	header	Module Token	Yes	⇄ `string`

Responses

Code	Description	Schema
200	The updated Post Comment Type	⇄ `▾postCommentTypeResponse { all of: ▾ { post_comment_type_key: string * code: string * created_at: string * updated_at: string * } }`
401	Unauthorized	⇄ `▾postCommentTypeErrorDefault { error: string * }`
404	Post Comment Type not Found	⇄ `▾postCommentTypeErrorDefault { error: string * }`
500	Failed to update Post Comment Type	⇄ `▾postCommentTypeErrorDefault { error: string * }`

Try this operation

DELETE /postCommentType/{postCommentTypeKey}

PostCommentType

Summary

Delete a Post Comment Type

Parameters

Name	Located in	Description	Required	Schema
postCommentTypeKey	path	Post Comment Type Key	Yes	⇄ `string`
X-MODULE-TOKEN	header	Module Token	Yes	⇄ `string`
X-AUTH-TOKEN	header	Authentication Token	Yes	⇄ `string`
X-ENTITY-KEY	header	Entity Key	Yes	⇄ `string`

Responses

Code	Description	Schema

200	OK	⇄ ▾**`replyDeletePostCommentType {`**
		`string: string`
		`}`

401	Unauthorized	⇄ ▾**`postCommentTypeErrorDefault {`**
		`error: string *`
		`}`

404	Post Comment Type not Found	⇄ ▾**`postCommentTypeErrorDefault {`**
		`error: string *`
		`}`

500	Failed to delete Post Comment Type	⇄ ▾**`postCommentTypeErrorDefault {`**
		`error: string *`
		`}`

Try this operation

/postlike/{id}

GET /postlike/{id}

PostLike

Summary

Show a Post Like

Parameters

Name	Located in	Description	Required	Schema
id	path	Post Like Id	Yes	⇄ string
X-MODULE-TOKEN	header	Module Token	Yes	⇄ string

Responses

Code	Description	Schema
200	Show the Post Like data	⇄
404	Post Like not Found	⇄
500	Failed to retrieve Post Like	⇄

200

```
▾postLikeResponse {
  all of:
     ▾ {
        post_id:    integer *
        like:       string *
        created_at: string *
        updated_at: string *
     }
}
```

404

```
▾postLikeErrorDefault {
  error: string *
}
```

500

```
▾postLikeErrorDefault {
  error: string *
}
```

Try this operation

PUT /postlike/{id}

PostLike

Summary

Update a Post Like

Parameters

Name	Located in	Description	Required	Schema
Body	body	Post Like Update data	Yes	⇄
id	path	Post Like Id	Yes	⇄ string
X-AUTH-TOKEN	header	User Auth Token	Yes	⇄ string
X-MODULE-TOKEN	header	Module Token	Yes	⇄ string

Body

```
▾postLike {
  all of:
    ▾ {
        post_id: integer
                 *
        like:    string
                 *
      }
}
```

Responses

Code	Description	Schema
200	The updated Post Like	⇄
401	Unauthorized	⇄
404	Post Like not Found	⇄
500	Failed to update Post Like	⇄

200

```
▾postLikeResponse {
  all of:
    ▾ {
        post_id:    integer *
        like:       string *
        created_at: string *
        updated_at: string *
      }
}
```

401

```
▾postLikeErrorDefault {
  error: string *
}
```

404

```
▾postLikeErrorDefault {
  error: string *
}
```

500

```
▾postLikeErrorDefault {
  error: string *
}
```

Try this operation

`DELETE /postlike/{id}`

PostLike

Summary

Delete a Post Like

Parameters

Name	Located in	Description	Required	Schema
id	path	Post Like Id	Yes	⇄ `string`
X-MODULE-TOKEN	header	Module Token	Yes	⇄ `string`
X-AUTH-TOKEN	header	Authentication Token	Yes	⇄ `string`
X-ENTITY-KEY	header	Entity Key	Yes	⇄ `string`

Responses

Code	Description	Schema

200	OK	⇄ ▾**replyDeletePostLike** {
		string: string
		}

401	Unauthorized	⇄ ▾**postLikeErrorDefault** {
		error: string *
		}

404	Post Like not Found	⇄ ▾**postLikeErrorDefault** {
		error: string *
		}

500	Failed to delete Post Like	⇄ ▾**postLikeErrorDefault** {
		error: string *
		}

Try this operation

/postlike

POST /postlike

PostLike

Summary

Creation of a Post Like

Parameters

Name	Located in	Description	Required	Schema

Body	body	Post Like data	Yes	⇄ ▾postLike {
				all of:
				▾ {
				post_id: integer *
				like: string *
				}
				}

X-AUTH-TOKEN	header	User Auth Token	Yes	⇄ string
X-MODULE-TOKEN	header	Module Token	Yes	⇄ string
X-ENTITY-KEY	header	Entity Key	Yes	⇄ string

Responses

Code	Description	Schema

201	the newly created post like	⇄ ▾postLikeResponse {
		all of:
		▾ {
		post_id: integer *
		like: string *
		created_at: string *
		updated_at: string *
		}
		}

401	Unauthorized	⇄ ▾postLikeErrorDefault {
		error: string *
		}

404	Unauthorized	⇄ ▾postLikeErrorDefault {
		error: string *
		}

409	Entity not Found	⇄ ▾postLikeErrorDefault {
		error: string *
		}

500	Failed to store new Post Like	⇄ ▾postLikeErrorDefault {
		error: string *
		}

Try this operation

`/post`

POST /post

Post

Summary

Creation of a Post

Parameters

Name	Located in	Description	Required	Schema
Body	body	Post data	Yes	⇄
X-AUTH-TOKEN	header	User Auth Token	Yes	⇄ string
X-MODULE-TOKEN	header	Module Token	Yes	⇄ string
X-ENTITY-KEY	header	Entity Key	Yes	⇄ string

Body:

```
▾post {
  all of:
    ▾ {
      topic_key:              string *
      type_code:              string
      parent_id:              integer
      created_by:             string
      enabled:                integer
      contents:               string *
      post_comment_type_id:   integer
    }
}
```

Responses

Code	Description	Schema
201	the newly created post	⇄

201:

```
▾postResponse {
  all of:
    ▾ {
      post_key:               string *
      version:                integer *
      parent_id:              integer *
      topic_id:               integer *
      post_comment_type_id:   integer *
      enabled:                integer *
      blocked:                integer *
      active:                 integer *
      created_by:             string *
      contents:               string *
      status_id:              integer *
      created_at:             string *
      updated_at:             string *
    }
}
```

Code	Description	Schema

401	Unauthorized	⇄ ▾ **postErrorDefault {**
		error: string *
		}

404	Unauthorized	⇄ ▾ **postErrorDefault {**
		error: string *
		}

409	Entity not Found	⇄ ▾ **postErrorDefault {**
		error: string *
		}

500	Failed to store new Post	⇄ ▾ **postErrorDefault {**
		error: string *
		}

Try this operation

/post/{postKey}

GET /post/{postKey}

Post

Summary

Show a Post

Parameters

Name	Located in	Description	Required	Schema
postKey	path	Post Key	Yes	⇄ string
X-MODULE-TOKEN	header	Module Token	Yes	⇄ string

Responses

Code **Description** **Schema**

200 Show the Post data ⇄

```
▾postResponse {
  all of:
    ▾ {
        post_key:             string *
        version:              integer *
        parent_id:            integer *
        topic_id:             integer *
        post_comment_type_id: integer *
        enabled:              integer *
        blocked:              integer *
        active:               integer *
        created_by:           string *
        contents:             string *
        status_id:            integer *
        created_at:           string *
        updated_at:           string *
    }
}
```

404 Site not Found ⇄

```
▾postErrorDefault {
   error: string *
}
```

500 Failed to retrieve Site ⇄

```
▾postErrorDefault {
   error: string *
}
```

Try this operation

PUT /post/{postKey}

Post

Summary

Update a User

Parameters

Name	Located in	Description	Required	Schema
Body	body	Post Update data	Yes	⇄

```
▾post {
   all of:
     ▾ {
       topic_key:              string
                               *
       type_code:              string
       parent_id:              integer
       created_by:             string
       enabled:                integer
       contents:               string
                               *
       post_comment_type_id:   integer
     }
}
```

Name	Located in	Description	Required	Schema
postKey	path	Post Key	Yes	⇄ string
X-AUTH-TOKEN	header	User Auth Token	Yes	⇄ string
X-MODULE-TOKEN	header	Module Token	Yes	⇄ string

Responses

Code	Description	Schema
200	The updated Post	⇄

```
▾postResponse {
   all of:
     ▾ {
       post_key:               string *
       version:                integer *
       parent_id:              integer *
       topic_id:               integer *
       post_comment_type_id:   integer *
       enabled:                integer *
       blocked:                integer *
       active:                 integer *
       created_by:             string *
       contents:               string *
       status_id:              integer *
       created_at:             string *
       updated_at:             string *
     }
}
```

Code	Description	Schema

401	Unauthorized	⇄ ▾**postErrorDefault {**
		error: string *
		}

404	Post not Found	⇄ ▾**postErrorDefault {**
		error: string *
		}

500	Failed to update Post	⇄ ▾**postErrorDefault {**
		error: string *
		}

Try this operation

`DELETE /post/{postKey}`

Post

Summary

Delete a Post

Parameters

Name	Located in	Description	Required	Schema
postKey	path	Post Key	Yes	⇄ `string`
X-MODULE-TOKEN	header	Module Token	Yes	⇄ `string`
X-AUTH-TOKEN	header	Authentication Token	Yes	⇄ `string`
X-ENTITY-KEY	header	Entity Key	Yes	⇄ `string`

Responses

Code	Description	Schema

200	OK	⇄ ▾**replyDeletePost {**
		string: string
		}

401	Unauthorized	⇄ ▾**postErrorDefault {**
		error: string *
		}

404	Post not Found	⇄ ▾**postErrorDefault {**
		error: string *
		}

500	Failed to delete Post	⇄ ▾**postErrorDefault {**
		error: string *
		}

Try this operation

`/status/{statusKey}`

`GET /status/{statusKey}`

Status

Summary

Show a Status

Parameters

Name	Located in	Description	Required	Schema
statusKey	path	Status Key	Yes	⇄ `string`
X-MODULE-TOKEN	header	Module Token	Yes	⇄ `string`

Responses

Code	Description	Schema
200	Show the Status data	⇄

```
▾statusResponse {
  all of:
    ▾ {
        status_key:     string *
        status_type_id: integer *
        topic_id:       integer *
        active:         integer *
        created_at:     string *
        updated_at:     string *
    }
}
```

404	Status not Found	⇄

```
▾statusErrorDefault {
   error: string *
}
```

500	Failed to retrieve Status	⇄

```
▾statusErrorDefault {
   error: string *
}
```

Try this operation

`/status`

POST /status

Status

Summary

Creation of a Status

Parameters

Name	Located in	Description	Required	Schema

Body	body	Status data	Yes	⇄ ▾**status {**
				all of:
				▾ **{**
				`code: string *`
				`topic_key: string *`
				`status_type_id: integer`
				`comment: string`
				`}`
				`}`

X-AUTH-TOKEN	header	User Auth Token	Yes	⇄ `string`
X-MODULE-TOKEN	header	Module Token	Yes	⇄ `string`
X-ENTITY-KEY	header	Entity Key	Yes	⇄ `string`

Responses

Code	Description	Schema

201	the newly created status	⇄ ▾**statusResponse {**
		all of:
		▾ **{**
		`status_key: string *`
		`status_type_id: integer *`
		`topic_id: integer *`
		`active: integer *`
		`created_at: string *`
		`updated_at: string *`
		`}`
		`}`

401	Unauthorized	⇄ ▾**statusErrorDefault {**
		`error: string *`
		`}`

404	Unauthorized	⇄ ▾**statusErrorDefault {**
		`error: string *`
		`}`

409	Entity not Found	⇄ ▾**statusErrorDefault {**
		`error: string *`
		`}`

500	Failed to store new Status	⇄ ▾**statusErrorDefault {**
		`error: string *`
		`}`

Try this operation

/statusTypes/{statusTypeKey}

GET /statusTypes/{statusTypeKey}

StatusType

Summary

Show a Status Type

Parameters

Name	Located in	Description	Required	Schema
statusTypeKey	path	Status Type Key	Yes	⇄ string
X-MODULE-TOKEN	header	Module Token	Yes	⇄ string

Responses

Code	Description	Schema
200	Show the Status Type data	⇄
404	Status Type not Found	⇄
500	Failed to retrieve Status Type	⇄

200:

```
▾statusTypeResponse {
  all of:
    ▾ {
       status_type_key: string *
       code:            string *
       position:        integer *
       created_at:      string *
       updated_at:      string *
       translations:    ▾[
                          ▾ {
                             status_type_id: integer
                             language_code:  integer
                             name:           integer
                             description:    integer
                             created_at:     integer
                             updated_at:     integer
                          }
                        ]
    }
}
```

404:

```
▾statusTypeErrorDefault {
  error: string *
}
```

500:

```
▾statusTypeErrorDefault {
  error: string *
}
```

Try this operation

PUT /statusTypes/{statusTypeKey}

StatusType

Summary

Update a Status Type

Parameters

Name	Located in	Description	Required	Schema
Body	body	Status Type Update data	Yes	⇄

```
▾statusType {
  all of:
    ▾ {
      code:          string
                     *
      position:      integer
                     *
      translations:  ▸[]
    }
}
```

statusTypeKey	path	Status Type Key	Yes	⇄ `string`
X-AUTH-TOKEN	header	User Auth Token	Yes	⇄ `string`
X-MODULE-TOKEN	header	Module Token	Yes	⇄ `string`

Responses

Code	Description	Schema
200	The updated Status Type	⇄

```
▾statusTypeResponse {
  all of:
    ▾ {
      status_type_key: string *
      code:            string *
      position:        integer *
      created_at:      string *
      updated_at:      string *
      translations:     ▾[
                          ▾ {
                            status_type_id: integer
                            language_code:  integer
                            name:           integer
                            description:    integer
                            created_at:     integer
                            updated_at:     integer
                          }
                        ]
    }
}
```

401	Unauthorized	⇄

```
▾statusTypeErrorDefault {
  error: string *
}
```

Code	Description	Schema

404	Status Type not Found	⇄ ▾**statusTypeErrorDefault {**
		`error: string *`
		}

500	Failed to update Status Type	⇄ ▾**statusTypeErrorDefault {**
		`error: string *`
		}

Try this operation

DELETE /statusTypes/{statusTypeKey}

StatusType

Summary

Delete a Status Type

Parameters

Name	Located in	Description	Required	Schema
statusTypeKey	path	Status Type Key	Yes	⇄ `string`
X-MODULE-TOKEN	header	Module Token	Yes	⇄ `string`
X-AUTH-TOKEN	header	Authentication Token	Yes	⇄ `string`
X-ENTITY-KEY	header	Entity Key	Yes	⇄ `string`

Responses

Code	Description	Schema

200	OK	⇄ ▾**replyDeleteStatusType {**
		`string: string`
		}

401	Unauthorized	⇄ ▾**statusTypeErrorDefault {**
		`error: string *`
		}

404	User not Found	⇄ ▾**statusTypeErrorDefault {**
		`error: string *`
		}

500	Failed to delete Status Type	⇄ ▾**statusTypeErrorDefault {**
		`error: string *`
		}

Try this operation

/statusTypes

POST /statusTypes

StatusType

Summary

Creation of a Status Type

Parameters

Name	Located in	Description	Required	Schema
Body	body	Status Type data	Yes	⇄
X-AUTH-TOKEN	header	User Auth Token	Yes	⇄ string
X-MODULE-TOKEN	header	Module Token	Yes	⇄ string
X-ENTITY-KEY	header	Entity Key	Yes	⇄ string

Body:

```
▾statusType {
   all of:
      ▾ {
         code:          string
                        *
         position:      integer
                        *
         translations:  ▸[]
      }
}
```

Responses

Code	Description	Schema
201	the newly created Status Type	⇄
401	Unauthorized	⇄

201:

```
▾statusTypeResponse {
   all of:
      ▾ {
         status_type_key: string *
         code:            string *
         position:        integer *
         created_at:      string *
         updated_at:      string *
         translations:    ▾[
                             ▾ {
                                status_type_id: integer
                                language_code:  integer
                                name:           integer
                                description:    integer
                                created_at:     integer
                                updated_at:     integer
                             }
                          ]
      }
}
```

401:

```
▾statusTypeErrorDefault {
   error: string *
}
```

Code **Description** **Schema**

404 Unauthorized ⇄

```
▾statusTypeErrorDefault {
   error: string *
}
```

409 Entity not Found ⇄

```
▾statusTypeErrorDefault {
   error: string *
}
```

500 Failed to store new Status Type ⇄

```
▾statusTypeErrorDefault {
   error: string *
}
```

Try this operation

/topicReviewReplies/{topicReviewReplyKey}

GET /topicReviewReplies/{topicReviewReplyKey}

TopicReviewReply

Summary

Show a Topic Review Reply

Parameters

Name	Located in	Description	Required	Schema
topicReviewReplyKey	path	Topic Review Reply Key	Yes	⇄ string
X-MODULE-TOKEN	header	Module Token	Yes	⇄ string

Responses

Code **Description** **Schema**

200 Show the Site data ⇄

```
▾topicReviewReplyResponse {
   all of:
     ▾ {
        topic_review_reply_key: string *
        topic_review_id:        string *
        content:                string *
        created_by:             string *
        created_at:             string *
        updated_at:             string *
        topic_review:           undefined
     }
}
```

404 Site not Found ⇄

```
▾topicReviewReplyErrorDefault {
   error: string *
}
```

500 Failed to retrieve Site ⇄

```
▾topicReviewReplyErrorDefault {
   error: string *
}
```

Try this operation

PUT /topicReviewReplies/{topicReviewReplyKey}

TopicReviewReply

Summary

Update a Topic Review Reply

Parameters

Name	Located in	Description	Required	Schema
Body	body	Topic Review Reply Update data	Yes	⇄
topicReviewReplyKey	path	Topic Review Reply Key	Yes	⇄ string
X-AUTH-TOKEN	header	User Auth Token	Yes	⇄ string
X-MODULE-TOKEN	header	Module Token	Yes	⇄ string

Body:

```
▾
topicReviewReplyUpdate
{
   all of:
     ▾ {
        content: string
                 *
        code:    string
                 *
       }
}
```

Responses

Code	Description	Schema
200	The updated Topic Review Reply	⇄
401	Unauthorized	⇄
404	Topic Review Reply not Found	⇄
500	Failed to update Topic Review Reply	⇄

200:

```
▾topicReviewReplyResponse {
  all of:
    ▾ {
      topic_review_reply_key: string *
      topic_review_id:        string *
      content:                string *
      created_by:             string *
      created_at:             string *
      updated_at:             string *
      topic_review:           undefined
    }
}
```

401:

```
▾topicReviewReplyErrorDefault {
  error: string *
}
```

404:

```
▾topicReviewReplyErrorDefault {
  error: string *
}
```

500:

```
▾topicReviewReplyErrorDefault {
  error: string *
}
```

Try this operation

DELETE /topicReviewReplies/{topicReviewReplyKey}

TopicReviewReply

Summary

Delete a Topic Review Reply

Parameters

Name	Located in	Description	Required	Schema
topicReviewReplyKey	path	Topic Review Reply Key	Yes	⇄ string
X-MODULE-TOKEN	header	Module Token	Yes	⇄ string
X-AUTH-TOKEN	header	Authentication Token	Yes	⇄ string
X-ENTITY-KEY	header	Entity Key	Yes	⇄ string

Responses

Code	Description	Schema

200	OK	⇄ ▾**replyDeleteTopicReviewReply {**
		string: string
		}

401	Unauthorized	⇄ ▾**topicReviewReplyErrorDefault {**
		error: string *
		}

404	Topic Review Reply not Found	⇄ ▾**topicReviewReplyErrorDefault {**
		error: string *
		}

500	Failed to delete Topic Review Reply	⇄ ▾**topicReviewReplyErrorDefault {**
		error: string *
		}

Try this operation

/topicReviewReplies

POST /topicReviewReplies

TopicReviewReply

Summary

Creation of a Topic Review Reply

Parameters

Name	Located in	Description	Required	Schema
Body	body	Topic Review Reply data	Yes	⇄
X-AUTH-TOKEN	header	User Auth Token	Yes	⇄ string
X-MODULE-TOKEN	header	Module Token	Yes	⇄ string
X-ENTITY-KEY	header	Entity Key	Yes	⇄ string

Body:

```
▾topicReviewReply {
   all of:
      ▾ {
         topic_review_key: string
                           *
         content:          string
                           *
         code:             string
                           *
        }
   }
```

Responses

Code	Description	Schema
201	the newly created Topic Review Reply	⇄
401	Unauthorized	⇄
404	Unauthorized	⇄
409	Entity not Found	⇄

201:

```
▾topicReviewReplyResponse {
   all of:
      ▾ {
         topic_review_reply_key: string *
         topic_review_id:        string *
         content:                string *
         created_by:             string *
         created_at:             string *
         updated_at:             string *
         topic_review:           undefined
        }
   }
```

401:

```
▾topicReviewReplyErrorDefault {
   error: string *
}
```

404:

```
▾topicReviewReplyErrorDefault {
   error: string *
}
```

409:

```
▾topicReviewReplyErrorDefault {
   error: string *
}
```

Code	Description	Schema
500	Failed to store new Topic Review Reply	⇄

```
▾topicReviewReplyErrorDefault {
  error: string *
}
```

Try this operation

/topicReviewStatus/{topicReviewStatusKey}

GET /topicReviewStatus/{topicReviewStatusKey}

TopicReviewStatus

Summary

Show a Topic Review Status

Parameters

Name	Located in	Description	Required	Schema
topicReviewStatusKey	path	Topic Review Status Key	Yes	⇄ string
X-MODULE-TOKEN	header	Module Token	Yes	⇄ string

Responses

Code	Description	Schema
200	Show the Topic Review Status data	⇄

```
▾topicReviewStatusResponse {
  all of:
    ▾ {
        topic_review_status_key:     string *
        topic_review_status_type_id: integer *
        topic_review_id:             integer *
        active:                      integer *
        created_by:                  string *
        created_at:                  string *
        updated_at:                  string *
      }
}
```

Code	Description	Schema
404	Topic Review Status not Found	⇄

```
▾topicReviewStatusErrorDefault {
  error: string *
}
```

Code	Description	Schema
500	Failed to retrieve Topic Review Status	⇄

```
▾topicReviewStatusErrorDefault {
  error: string *
}
```

Try this operation

`DELETE /topicReviewStatus/{topicReviewStatusKey}`

TopicReviewStatus

Summary

Delete a Topic Review Status

Parameters

Name	Located in	Description	Required	Schema
topicReviewStatusKey	path	Topic Review Status Key	Yes	⇄ `string`
X-MODULE-TOKEN	header	Module Token	Yes	⇄ `string`
X-AUTH-TOKEN	header	Authentication Token	Yes	⇄ `string`
X-ENTITY-KEY	header	Entity Key	Yes	⇄ `string`

Responses

Code **Description** **Schema**

200 OK ⇄

```
▾replyDeleteTopicReviewStatus {
  string: string
}
```

401 Unauthorized ⇄

```
▾topicReviewStatusErrorDefault {
  error: string *
}
```

404 Topic Review Status not Found ⇄

```
▾topicReviewStatusErrorDefault {
  error: string *
}
```

500 Failed to delete Topic Review Status ⇄

```
▾topicReviewStatusErrorDefault {
  error: string *
}
```

Try this operation

`/topicReviewStatusTypes/{topicReviewStatusTypeKey}`

GET /topicReviewStatusTypes/{topicReviewStatusTypeKey}

TopicReviewStatusType

Summary

Show a Topic Review Status Type

Parameters

Name	Located in	Description	Required	Schema
topicReviewStatusTypeKey	path	Topic Review Status Type Key	Yes	⇄ string
LANG_CODE	header	Language Code	Yes	⇄ string
X-MODULE-TOKEN	header	Module Token	Yes	⇄ string

Responses

Code **Description** **Schema**

200 Show the Topic Review Status Type data ⇄

```
▾topicReviewStatusTypeResponse {
  all of:
    ▾ {
      topic_review_status_type_key: string
                                    *
      code:                         string
                                    *
      position:                     integer
                                    *
      created_at:                   string
                                    *
      updated_at:                   string
                                    *
      translations:                 ▸[]
    }
}
```

404 Topic Review Status Type not Found ⇄

```
▾topicReviewStatusTypeErrorDefault {
  error: string *
}
```

500 Failed to retrieve Topic Review Status Type ⇄

```
▾topicReviewStatusTypeErrorDefault {
  error: string *
}
```

Try this operation

`DELETE /topicReviewStatusTypes/{topicReviewStatusTypeKey}`

TopicReviewStatusType

Summary

Delete a Topic Review Status Type

Parameters

Name	Located in	Description	Required	Schema
topicReviewStatusTypeKey	path	Topic Review Status Type Key	Yes	⇄ `string`
X-MODULE-TOKEN	header	Module Token	Yes	⇄ `string`
X-AUTH-TOKEN	header	Authentication Token	Yes	⇄ `string`
X-ENTITY-KEY	header	Entity Key	Yes	⇄ `string`

Responses

Code	Description	Schema

200	OK	⇄ ▾**replyDeleteTopicReviewStatusType**
		`{`
		`string: string`
		`}`

401	Unauthorized	⇄ ▾**topicReviewStatusTypeErrorDefault**
		`{`
		`error: string *`
		`}`

404	Topic Review Status Type not Found	⇄ ▾**topicReviewStatusTypeErrorDefault**
		`{`
		`error: string *`
		`}`

500	Failed to delete Topic Review Status Type	⇄ ▾**topicReviewStatusTypeErrorDefault**
		`{`
		`error: string *`
		`}`

Try this operation

`/topicReviewStatusTypes`

`POST /topicReviewStatusTypes`

TopicReviewStatusType

Summary

Creation of a Topic Review Status Type

Parameters

Name	Located in	Description	Required	Schema
Body	body	Topic Review Status Type data	Yes	⇄
X-AUTH-TOKEN	header	User Auth Token	Yes	⇄ `string`
X-MODULE-TOKEN	header	Module Token	Yes	⇄ `string`
X-ENTITY-KEY	header	Entity Key	Yes	⇄ `string`

Body:

```
▾topicReviewStatusType {
  all of:
    ▾ {
      code:          string
                     *
      position:      integer
                     *
      translations:  ▸[]
    }
}
```

Responses

Code	Description	Schema
201	the newly created user	⇄
401	Unauthorized	⇄

201:

```
▾topicReviewStatusTypeResponse {
  all of:
    ▾ {
      topic_review_status_type_key: string *
      code:                         string *
      position:                     integer *
      created_at:                   string *
      updated_at:                   string *
      translations:                 ▾[
                                      ▾ {
                                        language_code:
                                        topic_review_status
                                        name:
                                        description:
                                        created_at:
                                        updated_at:
                                      }
                                    ]
    }
}
```

401:

```
▾topicReviewStatusTypeErrorDefault {
  error: string *
}
```

Code	Description		Schema

404	Unauthorized	⇄	▾**`topicReviewStatusTypeErrorDefault {`**
			`  error: string *`
			`}`

409	Entity not Found	⇄	▾**`topicReviewStatusTypeErrorDefault {`**
			`  error: string *`
			`}`

500	Failed to store new Topic Review Status Type	⇄	▾**`topicReviewStatusTypeErrorDefault {`**
			`  error: string *`
			`}`

Try this operation

/topicReviews/{topicReviewKey}

GET /topicReviews/{topicReviewKey}

TopicReview

Summary

Show a Topic Review

Parameters

Name	Located in	Description	Required	Schema
topicReviewKey	path	Topic Review Key	Yes	⇄ `string`
X-MODULE-TOKEN	header	Module Token	Yes	⇄ `string`

Responses

Code **Description** **Schema**

200 Show the Topic Review data ⇄

```
▾topicReviewResponse {
  all of:
    ▾ {
        topic_review_key: string *
        topic_id:         integer *
        description:      string *
        subject:          string *
        created_by:       string *
        created_at:       string *
        updated_at:       string *
      }
}
```

404 Topic Review not Found ⇄

```
▾topicReviewErrorDefault {
  error: string *
}
```

500 Failed to retrieve Topic Review ⇄

```
▾topicReviewErrorDefault {
  error: string *
}
```

Try this operation

PUT /topicReviews/{topicReviewKey}

TopicReview

Summary

Update a Topic Review

Parameters

Name	Located in	Description	Required	Schema
Body	body	Topic Review Update data	Yes	⇄
topicReviewKey	path	Topic Review Key	Yes	⇄ string
X-AUTH-TOKEN	header	User Auth Token	Yes	⇄ string
X-MODULE-TOKEN	header	Module Token	Yes	⇄ string

```
▾topicReview {
  all of:
    ▾ {
      topic_review_key: string
                        *
      created_by:       string
                        *
      topic_id:         string
                        *
      reviewer_key:     string
      description:      string
                        *
      subject:          string
                        *
      topic_key:        string
      code:             string
      reviewers:        ▸[]
    }
}
```

Responses

Code	Description	Schema
200	The updated Topic Review	⇄

```
▾topicReviewResponse {
  all of:
    ▾ {
      topic_review_key: string *
      topic_id:         integer *
      description:      string *
      subject:          string *
      created_by:       string *
      created_at:       string *
      updated_at:       string *
    }
}
```

Code	Description	Schema

401	Unauthorized	⇄ ▾**topicReviewErrorDefault {**
		`error: string *`
		}

404	Topic Review not Found	⇄ ▾**topicReviewErrorDefault {**
		`error: string *`
		}

500	Failed to update Topic Review	⇄ ▾**topicReviewErrorDefault {**
		`error: string *`
		}

Try this operation

`DELETE /topicReviews/{topicReviewKey}`

TopicReview

Summary

Delete a Topic Review

Parameters

Name	Located in	Description	Required	Schema
topicReviewKey	path	Topic Review Key	Yes	⇄ `string`
X-MODULE-TOKEN	header	Module Token	Yes	⇄ `string`
X-AUTH-TOKEN	header	Authentication Token	Yes	⇄ `string`
X-ENTITY-KEY	header	Entity Key	Yes	⇄ `string`

Responses

Code	Description	Schema

200	OK	⇄ ▾**replyDeleteTopicReview {**
		`string: string`
		}

401	Unauthorized	⇄ ▾**topicReviewErrorDefault {**
		`error: string *`
		}

404	Topic Review not Found	⇄ ▾**topicReviewErrorDefault {**
		`error: string *`
		}

500	Failed to delete Topic Review	⇄ ▾**topicReviewErrorDefault {**
		`error: string *`
		}

Try this operation

`/topicReviews`

POST /topicReviews

TopicReview

Summary

Creation of a Topic Review

Parameters

Name	Located in	Description	Required	Schema
Body	body	Topic Review data	Yes	⇄
X-AUTH-TOKEN	header	User Auth Token	Yes	⇄ `string`
X-MODULE-TOKEN	header	Module Token	Yes	⇄ `string`
X-ENTITY-KEY	header	Entity Key	Yes	⇄ `string`

Body:

```
▾topicReview {
  all of:
    ▾ {
      topic_review_key: string *
      created_by:       string *
      topic_id:         string *
      reviewer_key:     string
      description:      string *
      subject:          string *
      topic_key:        string
      code:             string
      reviewers:        ▸[]
    }
}
```

Responses

Code	Description	Schema
201	the newly created topic review	⇄
401	Unauthorized	⇄

201:

```
▾topicReviewResponse {
  all of:
    ▾ {
      topic_review_key: string *
      topic_id:         integer *
      description:      string *
      subject:          string *
      created_by:       string *
      created_at:       string *
      updated_at:       string *
    }
}
```

401:

```
▾topicReviewErrorDefault {
  error: string *
}
```

Code	Description		Schema

404	Unauthorized	⇄	▾**`topicReviewErrorDefault {`**
			`error: string *`
			`}`

409	Entity not Found	⇄	▾**`topicReviewErrorDefault {`**
			`error: string *`
			`}`

500	Failed to store new topic review	⇄	▾**`topicReviewErrorDefault {`**
			`error: string *`
			`}`

Try this operation

/topic/{topicKey}

GET /topic/{topicKey}

Topic

Summary

Show a Topic

Parameters

Name	Located in	Description	Required	Schema
topicKey	path	Topic Key	Yes	⇄ string
X-MODULE-TOKEN	header	Module Token	Yes	⇄ string

Responses

Code	Description	Schema
200	Show the Topic data	⇄

```
▾topicResponse {
  all of:
   ▾ {
       topic_key:     string *
       cb_id:         string *
       created_by:    string *
       title:         string *
       contents:      string *
       blocked:       string *
       q_key:         string *
       topic_number:  string *
       start_date:    string *
       end_date:      string *
       created_at:    string *
       updated_at:    string *
       first_post:   ▾firstPost {
                        all of:
                         ▾ {
                             post_key:             string
                                                   *
                             version:              integer
                                                   *
                             parent_id:            integer
                                                   *
                             topic_id:             integer
                                                   *
                             post_comment_type_id: integer
                                                   *
                             created_by:           string
                                                   *
                             contents:             string
                                                   *
                             status_id:            integer
                                                   *
                             created_at:           string
                                                   *
                             updated_at:           string
                                                   *
                             enabled:              integer
                                                   *
```

Code **Description** **Schema**

```
                                                    blocked:              integer
                                                                          *

                                                    active:               integer
                                                                          *
                                                }
                                            }
                          cb:               ▾cbTopic {
                                              all of:
                                                ▾ {
                                                    cb_key:       string *
                                                    parent_cb_id: integer *
                                                    title:        string *
                                                    contents:     string *
                                                    created_by:   string *
                                                    blocked:      integer *
                                                    status_id:    string *
                                                    layout_code:  string *
                                                    start_date:   string *
                                                    end_date:     string *
                                                    created_at:   string *
                                                    updated_at:   string *
                                                }
                                            }
                          configurations: ▾configurationsTopic {
                                              all of:
                                                ▾ {
                                                    cm_key: string *
                                                }
                                            }
                      }
                  }
```

Code	Description		Schema

404	Topic not Found	⇄	▾**topicErrorDefault {**
			error: string *
			}

500	Failed to retrieve Topic	⇄	▾**topicErrorDefault {**
			error: string *
			}

Try this operation

PUT /topic/{topicKey}

Topic

Summary

Update a Topic

Parameters

Name	Located in	Description	Required	Schema
Body	body	User Update data	Yes	⇄
topicKey	path	Topic Key	Yes	⇄ string
X-AUTH-TOKEN	header	User Auth Token	Yes	⇄ string
X-MODULE-TOKEN	header	Module Token	Yes	⇄ string

```
▾topic {
  all of:
    ▾ {
        cb_key:      string *
        title:       string *
        contents:    string *
        parameters:  ▸
                     parametersTopic
                     { }
    }
}
```

Responses

Code **Description** **Schema**

```
▾topicResponse {
  all of:
    ▾ {
        topic_key:     string *
        cb_id:         string *
        created_by:    string *
        title:         string *
        contents:      string *
        blocked:       string *
        q_key:         string *
        topic_number:  string *
        start_date:    string *
        end_date:      string *
        created_at:    string *
        updated_at:    string *
        first_post:    ▾firstPost {
                         all of:
                           ▾ {
                               post_key:            string
                                                    *
                               version:             integer
                                                    *
                               parent_id:           integer
                                                    *
```

Code **Description** **Schema**

200 The updated Topic ⇄

```
               topic_id:            integer *
               post_comment_type_id: integer *
               created_by:          string *
               contents:            string *
               status_id:           integer *
               created_at:          string *
               updated_at:          string *
               enabled:             integer *
               blocked:             integer *
               active:              integer *
              }
            }
   cb:             ▾cbTopic {
             all of:
              ▾ {
               cb_key:       string *
               parent_cb_id: integer *
               title:        string *
               contents:     string *
               created_by:   string *
               blocked:      integer *
               status_id:    string *
               layout_code:  string *
               start_date:   string *
               end_date:     string *
               created_at:   string *
               updated_at:   string *
              }
            }
   configurations: ▾configurationsTopic {
             all of:
              ▾ {
               cm_key: string *
              }
            }
  }
}
```

401 Unauthorized ⇄

```
▾topicErrorDefault {
  error: string *
}
```

404 Topic not Found ⇄

```
▾topicErrorDefault {
  error: string *
}
```

500 Failed to update Topic ⇄

```
▾topicErrorDefault {
  error: string *
}
```

Try this operation

`DELETE /topic/{topicKey}`

Topic

Summary

Delete a Topic

Parameters

Name	Located in	Description	Required	Schema
topicKey	path	Topic Key	Yes	⇄ `string`
X-MODULE-TOKEN	header	Module Token	Yes	⇄ `string`
X-AUTH-TOKEN	header	Authentication Token	Yes	⇄ `string`
X-ENTITY-KEY	header	Entity Key	Yes	⇄ `string`

Responses

Code	Description	Schema

200	OK	⇄ ▾**replyDeleteTopic** {
		string: string
		}

401	Unauthorized	⇄ ▾**topicErrorDefault** {
		error: string *
		}

404	Topic not Found	⇄ ▾**topicErrorDefault** {
		error: string *
		}

500	Failed to delete Topic	⇄ ▾**topicErrorDefault** {
		error: string *
		}

Try this operation

`/topic`

`POST /topic`

Topic

Summary

Creation of a Topic

Parameters

Name	Located in	Description	Required	Schema
Body	body	Topic data	Yes	⇄
X-AUTH-TOKEN	header	User Auth Token	Yes	⇄ string
X-MODULE-TOKEN	header	Module Token	Yes	⇄ string
X-ENTITY-KEY	header	Entity Key	Yes	⇄ string

```
▾topic {
  all of:
    ▾ {
        cb_key:       string *
        title:        string *
        contents:     string *
        parameters:   ▸
                      parametersTopic
                      { }
      }
}
```

Responses

Code	Description	Schema

```
▾topicResponse {
  all of:
    ▾ {
        topic_key:      string *
        cb_id:          string *
        created_by:     string *
        title:          string *
        contents:       string *
        blocked:        string *
        q_key:          string *
        topic_number:   string *
        start_date:     string *
        end_date:       string *
        created_at:     string *
        updated_at:     string *
        first_post:     ▾firstPost {
                          all of:
                            ▾ {
                                post_key:             string
                                                      *
                                version:              integer
                                                      *
                                parent_id:
```

Code **Description** **Schema**

201 the newly created topic ⇄

```
                                                    integer *
                              topic_id:             integer *
                              post_comment_type_id: integer *
                              created_by:           string *
                              contents:             string *
                              status_id:            integer *
                              created_at:           string *
                              updated_at:           string *
                              enabled:              integer *
                              blocked:              integer *
                              active:               integer *
                            }
                       }
       cb:             ▾cbTopic {
                         all of:
                            ▾ {
                              cb_key:       string *
                              parent_cb_id: integer *
                              title:        string *
                              contents:     string *
                              created_by:   string *
                              blocked:      integer *
                              status_id:    string *
                              layout_code:  string *
                              start_date:   string *
                              end_date:     string *
                              created_at:   string *
                              updated_at:   string *
                            }
                       }
       configurations: ▾configurationsTopic {
                         all of:
                            ▾ {
                              cm_key: string *
                            }
                       }
     }
}
```

401 Unauthorized ⇄

```
▾topicErrorDefault {
  error: string *
}
```

404 Unauthorized ⇄

```
▾topicErrorDefault {
  error: string *
}
```

409 Entity not Found ⇄

```
▾topicErrorDefault {
  error: string *
}
```

Code	Description	Schema
500	Failed to store new Topic	⇄

```
▾topicErrorDefault {
   error: string *
}
```

Try this operation

/voteConfigurations/{voteConfigurationKey}

GET /voteConfigurations/{voteConfigurationKey}

VoteConfiguration

Summary

Show a Vote Configuration

Parameters

Name	Located in	Description	Required	Schema
voteConfigurationKey	path	Vote Configuration Key	Yes	⇄ string
X-MODULE-TOKEN	header	Module Token	Yes	⇄ string

Responses

Code	Description	Schema
200	Show the Vote Configuration data	⇄

```
▾voteConfigurationResponse {
  all of:
     ▾ {
        vote_configuration_key: string
                                *
        code:                   string
                                *
        created_at:             string
                                *
        updated_at:             string
                                *
        translations:           ▸[]
     }
}
```

Code	Description	Schema
404	Vote Configuration not Found	⇄

```
▾voteConfigurationErrorDefault {
   error: string *
}
```

Code	Description	Schema
500	Failed to retrieve Vote Configuration	⇄

```
▾voteConfigurationErrorDefault {
   error: string *
}
```

Try this operation

PUT /voteConfigurations/{voteConfigurationKey}

VoteConfiguration

Summary

Update a Vote Configuration

Parameters

Name	Located in	Description	Required	Schema
Body	body	Vote Configuration Update data	Yes	⇄
voteConfigurationKey	path	Vote Configuration Key	Yes	⇄ string
X-AUTH-TOKEN	header	User Auth Token	Yes	⇄ string
X-MODULE-TOKEN	header	Module Token	Yes	⇄ string

```
▾voteConfiguration {
  all of:
    ▾ {
       code:           string
                       *
       translations:  ▸[]
    }
}
```

Responses

Code	Description	Schema
200	The updated Vote Configuration	⇄
401	Unauthorized	⇄
404	Vote Configuration not Found	⇄
500	Failed to update Vote Configuration	⇄

```
▾voteConfigurationResponse {
  all of:
    ▾ {
      vote_configuration_key: string
                              *
      code:                   string
                              *
      created_at:             string
                              *
      updated_at:             string
                              *
      translations:          ▸[]
    }
}
```

```
▾voteConfigurationErrorDefault {
   error: string *
}
```

```
▾voteConfigurationErrorDefault {
   error: string *
}
```

```
▾voteConfigurationErrorDefault {
   error: string *
}
```

Try this operation

DELETE /voteConfigurations/{voteConfigurationKey}

VoteConfiguration

Summary

Delete a Vote Configuration

Parameters

Name	Located in	Description	Required	Schema
voteConfigurationKey	path	Vote Configuration Key	Yes	⇄ string
X-MODULE-TOKEN	header	Module Token	Yes	⇄ string
X-AUTH-TOKEN	header	Authentication Token	Yes	⇄ string
X-ENTITY-KEY	header	Entity Key	Yes	⇄ string

Responses

Code	Description	Schema

200	OK	⇄ ▾**replyDeleteVoteConfiguration** {
		string: string
		}

401	Unauthorized	⇄ ▾**voteConfigurationErrorDefault** {
		error: string *
		}

404	VoteConfiguration not Found	⇄ ▾**voteConfigurationErrorDefault** {
		error: string *
		}

500	Failed to delete VoteConfiguration	⇄ ▾**voteConfigurationErrorDefault** {
		error: string *
		}

Try this operation

/voteConfigurations

POST /voteConfigurations

VoteConfiguration

Summary

Creation of a Vote Configuration

Parameters

Name	**Located in**	**Description**	**Required**	**Schema**

Body	body	Vote Configuration data	Yes	⇄ ▾voteConfiguration {
				all of:
				▾ {
				code: string *
				translations: ▸[]
				}
				}

X-AUTH-TOKEN	header	User Auth Token	Yes	⇄ string
X-MODULE-TOKEN	header	Module Token	Yes	⇄ string
X-ENTITY-KEY	header	Entity Key	Yes	⇄ string

Responses

Code	**Description**	**Schema**

201	the newly created vote configuration	⇄ ▾voteConfigurationResponse {
		all of:
		▾ {
		vote_configuration_key: string *
		code: string *
		created_at: string *
		updated_at: string *
		translations: ▸[]
		}
		}

401	Unauthorized	⇄ ▾voteConfigurationErrorDefault {
		error: string *
		}

404	Unauthorized	⇄ ▾voteConfigurationErrorDefault {
		error: string *
		}

409	Entity not Found	⇄ ▾voteConfigurationErrorDefault {
		error: string *
		}

500	Failed to store new Vote Configuraton	⇄ ▾voteConfigurationErrorDefault {
		error: string *
		}

Try this operation

Models

annotationsErrorDefault

```
▾annotationsErrorDefault {
   error: string *
}
```

annotations

```
▾annotations {
  all of:
    ▾ {
      text:   string *
      quote:  string *
      ranges: ▾[
                ▾ {
                  start:        string
                  end:          string
                  start_offset: string
                  end_offset:   string
                }
              ]
      tags:   ▾[
                ▾ {
                  code: string
                }
              ]
    }
}
```

annotationsUpdate

```
▾annotationsUpdate {
  all of:
    ▾ {
      text: string *
      tags: ▾[
              ▾ {
                code: string
              }
            ]
    }
}
```

annotationsResponse

```
▾annotationsResponse {
  all of:
    ▾ {
      annotation_key: string *
      text:           string
      quote:          string
    }
}
```

annotationsRanges

```
▾annotationsRanges {
  all of:
    ▾ {
      ranges: ▾[
                ▾ {
                  start:        string
                  end:          string
                  start_offset: string
                  end_offset:   string
                }
              ]
    }
}
```

replyDeleteAnnotation

```
▾replyDeleteAnnotation {
  string: string
}
```

cbErrorDefault

```
▾cbErrorDefault {
  error: string *
}
```

cb

```
    ▾cb {
      all of:
        ▾ {
          title:        string *
          contents:     string *
          created_by:   string *
          blocked:      string *
⇄         status_id:    string *
          layout_code:  string *
          parent_cb_id: string *
          start_date:   boolean *
          end_date:     boolean *
        }
    }
```

cbResponse

```
    ▾cbResponse {
      all of:
        ▾ {
          cbKey:        string *
          title:        string *
          contents:     string *
          created_by:   string *
⇄         blocked:      string *
          status_id:    string *
          layout_code:  string *
          parent_cb_id: string *
          start_date:   boolean *
          end_date:     boolean *
        }
    }
```

replyDeleteCb

```
    ▾replyDeleteCb {
⇄      string: string
    }
```

commentErrorDefault

```
    ▾commentErrorDefault {
⇄      error: string *
    }
```

commentsResponse

```
▾commentsResponse {
  all of:
    ▾ {
      comment_key: string *
      title:       string *
      content:     string *
    }
}
```

comments

```
▾comments {
  all of:
    ▾ {
      public:     string *
      title:      string *
      content:    string *
      created_by: string *
    }
}
```

commentsUpdate

```
▾commentsUpdate {
  all of:
    ▾ {
      public:  string *
      title:   string *
      content: string *
    }
}
```

replyDeleteComment

```
▾replyDeleteComment {
   string: string
}
```

configOptsErrorDefault

```
▾configOptsErrorDefault {
   error: string *
}
```

congOpts

```
▾congOpts {
  all of:
    ▾ {
      cm_key: string *
    }
}
```

congOptsResponse

```
▾congOptsResponse {
  all of:
    ▾ {
      conf_opt_key: string *
    }
}
```

configurationTypeErrorDefault

```
▾configurationTypeErrorDefault {
  error: string *
}
```

configurationType

```
▾configurationType {
  all of:
    ▾ {
      code:         string *
      translations: ▾[
                      ▾ {
                        configuration_type_id: integer
                        language_code:         string
                        title:                 string
                        description:           string
                        created_at:            string
                        updated_at:            string
                      }
                    ]
    }
}
```

configurationTypeResponse

```
▾configurationTypeResponse {
  all of:
    ▾ {
      code:         string *
      created_by:   string *
      created_at:   string *
      updated_at:   string *
      translations: ▾[
                      ▾ {
                        configuration_type_id: integer
                        language_code:         string
                        title:                 string
                        description:           string
                        created_at:            string
                        updated_at:            string
                      }
                    ]
    }
}
```

replyDeleteConfigurationType

```
▾replyDeleteConfigurationType {
  string: string
}
```

configurationErrorDefault

```
▾configurationErrorDefault {
  error: string *
}
```

configurationResponse

```
▾configurationResponse {
  all of:
    ▾ {
      code: string *
    }
}
```

configurations

```
    ▾configurations {
      all of:
        ▾ {
          configuration_type_id: string *
          code:                  string *
          created_by:            string *
          translations:         ▾[
⇄                                ▾ {
                                   language_code: string
                                   ttitle:        string
                                   description:   string
                                 }
                               ]
        }
    }
```

replyDeleteConfiguration

```
    ▾replyDeleteConfiguration {
⇄     string: string
    }
```

parameterTemplateErrorDefault

```
    ▾parameterTemplateErrorDefault {
⇄     error: string *
    }
```

parameterTemplateResponse

```
    ▾parameterTemplateResponse {
      all of:
        ▾ {
          parameter_template_key: string *
          code:                   string *
          mandatory:              integer *
          parameter:              string
          description:            string
⇄         value:                  string *
          currency:               string *
          position:               integer *
          use_filter:             integer *
          visible_in_list:        integer *
          visible:                integer *
        }
    }
```

parameterTemplate

```
▾parameterTemplate {
  all of:
    ▾ {
      parameter_type_id: integer *
      code:              string *
      mandatory:         integer *
      parameter:         string
      description:       string
      value:             string
      currency:          string
      position:          integer *
      use_filter:        integer *
      visible_in_list:   integer *
      visible:           integer *
    }
}
```

replyDeleteParameterTemplate

```
▾replyDeleteParameterTemplate {
  string: string
}
```

parameterTypeErrorDefault

```
▾parameterTypeErrorDefault {
  error: string *
}
```

parameterType

```
▾parameterType {
  all of:
    ▾ {
      name: string *
      code: string *
    }
}
```

replyDeleteParameterType

```
▾replyDeleteParameterType {
  string: string
}
```

parameterErrorDefault

```
▾parameterErrorDefault {
  error: string *
}
```

parameterResponse

```
    ▾parameterResponse {
      all of:
        ▾ {
          code:            string *
          mandatory:       integer *
          parameter:       string
          description:     string
⇄         value:           string *
          currency:        string *
          position:        integer *
          use_filter:      integer *
          visible_in_list: integer *
          visible:         integer *
        }
    }
```

parameter

```
    ▾parameter {
      all of:
        ▾ {
          parameter_type_id: integer *
          code:              string *
          mandatory:         integer *
          parameter:         string
          description:       string
⇄         value:             string
          currency:          string
          position:          integer *
          use_filter:        integer *
          visible_in_list:   integer *
          visible:           integer *
        }
    }
```

replyDeleteParameter

```
    ▾replyDeleteParameter {
⇄     string: string
    }
```

postAbuseErrorDefault

```
    ▾postAbuseErrorDefault {
⇄     error: string *
    }
```

postAbuse

```
▾postAbuse {
  all of:
    ▾ {
      post_key:   string *
      type_id:    integer *
      comment:    string *
      created_by: string *
    }
}
```

postAbuseResponse

```
▾postAbuseResponse {
  all of:
    ▾ {
      id:         integer *
      post_id:    integer *
      type_id:    integer *
      comment:    string *
      created_by: string *
      processed:  integer *
      created_at: string *
      updated_at: string *
    }
}
```

replyDeletePostAbuse

```
▾replyDeletePostAbuse {
  string: string
}
```

postCommentTypeErrorDefault

```
▾postCommentTypeErrorDefault {
  error: string *
}
```

postCommentType

```
▾postCommentType {
  all of:
    ▾ {
      code: string *
    }
}
```

postCommentTypeResponse

```
   ▾postCommentTypeResponse {
     all of:
        ▾ {
           post_comment_type_key: string *
⇄          code:                  string *
           created_at:            string *
           updated_at:            string *
        }
   }
```

replyDeletePostCommentType

```
   ▾replyDeletePostCommentType {
⇄     string: string
   }
```

postLikeErrorDefault

```
   ▾postLikeErrorDefault {
⇄     error: string *
   }
```

postLikeResponse

```
   ▾postLikeResponse {
     all of:
        ▾ {
           post_id:    integer *
⇄          like:       string *
           created_at: string *
           updated_at: string *
        }
   }
```

postLike

```
   ▾postLike {
     all of:
        ▾ {
⇄          post_id: integer *
           like:    string *
        }
   }
```

replyDeletePostLike

```
   ▾replyDeletePostLike {
⇄     string: string
   }
```

postErrorDefault

```
   ▾postErrorDefault {
⇄     error: string *
   }
```

post

```
▾post {
  all of:
    ▾ {
      topic_key:            string *
      type_code:            string
      parent_id:            integer
      created_by:           string
      enabled:              integer
      contents:             string *
      post_comment_type_id: integer
    }
}
```

postResponse

```
▾postResponse {
  all of:
    ▾ {
      post_key:             string *
      version:              integer *
      parent_id:            integer *
      topic_id:             integer *
      post_comment_type_id: integer *
      enabled:              integer *
      blocked:              integer *
      active:               integer *
      created_by:           string *
      contents:             string *
      status_id:            integer *
      created_at:           string *
      updated_at:           string *
    }
}
```

replyDeletePost

```
▾replyDeletePost {
  string: string
}
```

statusErrorDefault

```
▾statusErrorDefault {
  error: string *
}
```

status

```
    ▾status {
      all of:
        ▾ {
          code:           string *
⇄         topic_key:      string *
          status_type_id: integer
          comment:        string
        }
    }
```

statusResponse

```
    ▾statusResponse {
      all of:
        ▾ {
          status_key:     string *
          status_type_id: integer *
⇄         topic_id:       integer *
          active:         integer *
          created_at:     string *
          updated_at:     string *
        }
    }
```

statusTypeErrorDefault

```
    ▾statusTypeErrorDefault {
⇄     error: string *
    }
```

statusTypeResponse

```
  ▾statusTypeResponse {
    all of:
      ▾ {
        status_type_key: string *
        code:            string *
        position:        integer *
        created_at:      string *
        updated_at:      string *
        translations:    ▾[
                           ▾ {
⇄                            status_type_id: integer
                             language_code:  integer
                             name:           integer
                             description:    integer
                             created_at:     integer
                             updated_at:     integer
                           }
                         ]
      }
  }
```

statusType

```
  ▾statusType {
    all of:
      ▾ {
        code:         string *
        position:     integer *
        translations: ▾[
                        ▾ {
                          status_type_id: integer
⇄                         language_code:  integer
                          name:           integer
                          description:    integer
                          created_at:     integer
                          updated_at:     integer
                        }
                      ]
      }
  }
```

replyDeleteStatusType

```
  ▾replyDeleteStatusType {
⇄    string: string
  }
```

topicReviewReplyErrorDefault

```
    ▾topicReviewReplyErrorDefault {
 ⇄    error: string *
    }
```

topicReviewReplyResponse

```
    ▾topicReviewReplyResponse {
      all of:
        ▾ {
          topic_review_reply_key: string *
          topic_review_id:        string *
          content:                string *
 ⇄        created_by:             string *
          created_at:             string *
          updated_at:             string *
          topic_review:           undefined
        }
    }
```

topicReviewReply

```
    ▾topicReviewReply {
      all of:
        ▾ {
          topic_review_key: string *
 ⇄        content:          string *
          code:             string *
        }
    }
```

topicReviewReplyUpdate

```
    ▾topicReviewReplyUpdate {
      all of:
        ▾ {
 ⇄        content: string *
          code:    string *
        }
    }
```

replyDeleteTopicReviewReply

```
    ▾replyDeleteTopicReviewReply {
 ⇄    string: string
    }
```

topicReviewStatusErrorDefault

```
    ▾topicReviewStatusErrorDefault {
 ⇄    error: string *
    }
```

topicReviewStatus

```
▾topicReviewStatus {
  all of:
    ▾ {
      cm_key: string *
    }
}
```

topicReviewStatusResponse

```
▾topicReviewStatusResponse {
  all of:
    ▾ {
      topic_review_status_key:     string *
      topic_review_status_type_id: integer *
      topic_review_id:             integer *
      active:                      integer *
      created_by:                  string *
      created_at:                  string *
      updated_at:                  string *
    }
}
```

replyDeleteTopicReviewStatus

```
▾replyDeleteTopicReviewStatus {
  string: string
}
```

topicReviewStatusTypeErrorDefault

```
▾topicReviewStatusTypeErrorDefault {
  error: string *
}
```

topicReviewStatusType

```
▾topicReviewStatusType {
  all of:
    ▾ {
      code:          string *
      position:      integer *
      translations: ▾[
                       ▾ {
                         language_code: string
                         name:          string
                         description:   string
                       }
                     ]
    }
}
```

topicReviewStatusTypeResponse

```
   ▾topicReviewStatusTypeResponse {
     all of:
       ▾ {
         topic_review_status_type_key: string *
         code:                         string *
         position:                     integer *
         created_at:                   string *
         updated_at:                   string *
         translations:                 ▾[
                                         ▾ {
⇄                                          language_code:                string
                                           topic_review_status_type_id: integer
                                           name:                         string
                                           description:                  string
                                           created_at:                   string
                                           updated_at:                   string
                                         }
                                       ]
       }
   }
```

replyDeleteTopicReviewStatusType

```
   ▾replyDeleteTopicReviewStatusType {
⇄    string: string
   }
```

topicReviewErrorDefault

```
   ▾topicReviewErrorDefault {
⇄    error: string *
   }
```

topicReviewResponse

```
   ▾topicReviewResponse {
     all of:
       ▾ {
         topic_review_key: string *
         topic_id:         integer *
         description:      string *
⇄        subject:          string *
         created_by:       string *
         created_at:       string *
         updated_at:       string *
       }
   }
```

topicReview

```
    ▾topicReview {
      all of:
        ▾ {
          topic_review_key: string *
          created_by:       string *
          topic_id:         string *
          reviewer_key:     string
          description:      string *
          subject:          string *
⇄         topic_key:        string
          code:             string
          reviewers:        ▾[
                              ▾ {
                                key:      string
                                is_group: integer
                              }
                             ]
        }
    }
```

replyDeleteTopicReview

```
    ▾replyDeleteTopicReview {
⇄     string: string
    }
```

topicErrorDefault

```
    ▾topicErrorDefault {
⇄     error: string *
    }
```

configurationsTopic

```
    ▾configurationsTopic {
      all of:
        ▾ {
⇄         cm_key: string *
        }
    }
```

cbTopic

```
▾cbTopic {
  all of:
    ▾ {
      cb_key:       string *
      parent_cb_id: integer *
      title:        string *
      contents:     string *
      created_by:   string *
      blocked:      integer *
      status_id:    string *
      layout_code:  string *
      start_date:   string *
      end_date:     string *
      created_at:   string *
      updated_at:   string *
    }
}
```

firstPost

```
▾firstPost {
  all of:
    ▾ {
      post_key:             string *
      version:              integer *
      parent_id:            integer *
      topic_id:             integer *
      post_comment_type_id: integer *
      created_by:           string *
      contents:             string *
      status_id:            integer *
      created_at:           string *
      updated_at:           string *
      enabled:              integer *
      blocked:              integer *
      active:               integer *
    }
}
```

topicResponse

```
▾topicResponse {
  all of:
    ▾ {
      topic_key:      string *
      cb_id:          string *
      created_by:     string *
      title:          string *
```

```
contents:       string *
blocked:        string *
q_key:          string *
topic_number:   string *
start_date:     string *
end_date:       string *
created_at:     string *
updated_at:     string *
first_post:     ▾firstPost {
                  all of:
                     ▾ {
                       post_key:             string *
                       version:              integer *
                       parent_id:            integer *
                       topic_id:             integer *
                       post_comment_type_id: integer *
                       created_by:           string *
                       contents:             string *
                       status_id:            integer *
                       created_at:           string *
                       updated_at:           string *
⇄                      enabled:              integer *
                       blocked:              integer *
                       active:               integer *
                     }
                }
cb:             ▾cbTopic {
                  all of:
                     ▾ {
                       cb_key:       string *
                       parent_cb_id: integer *
                       title:        string *
                       contents:     string *
                       created_by:   string *
                       blocked:      integer *
                       status_id:    string *
                       layout_code:  string *
                       start_date:   string *
                       end_date:     string *
                       created_at:   string *
                       updated_at:   string *
                     }
                }
configurations: ▾configurationsTopic {
                  all of:
```

```
                                      ▾ {
                                          cm_key: string *
                                        }
                                   }
                }
            }
```

parametersTopic

```
      ▾ parametersTopic {
          all of:
            ▾ {
⇄              parameter_id: integer *
               value:        string *
              }
        }
```

topic

```
      ▾ topic {
          all of:
            ▾ {
               cb_key:     string *
               title:      string *
               contents:   string *
               parameters: ▾ parametersTopic {
⇄                              all of:
                                 ▾ {
                                    parameter_id: integer *
                                    value:        string *
                                   }
                               }
              }
        }
```

replyDeleteTopic

```
      ▾ replyDeleteTopic {
⇄         string: string
        }
```

voteConfigurationErrorDefault

```
      ▾ voteConfigurationErrorDefault {
⇄         error: string *
        }
```

voteConfigurationResponse

```
  ▾voteConfigurationResponse {
    all of:
      ▾ {
        vote_configuration_key: string *
        code:                   string *
        created_at:             string *
        updated_at:             string *
        translations:          ▾[
                                 ▾ {
⇄                                  vote_configuration_id: integer
                                   language_code:         string
                                   name:                  string
                                   description:           string
                                   created_at:            string
                                   updated_at:            string
                                 }
                               ]
      }
  }
```

voteConfiguration

```
  ▾voteConfiguration {
    all of:
      ▾ {
        code:         string *
        translations: ▾[
                        ▾ {
                          vote_configuration_id: integer
                          language_code:         string
⇄                         name:                  string
                          description:           string
                          created_at:            string
                          updated_at:            string
                        }
                      ]
      }
  }
```

replyDeleteVoteConfiguration

```
  ▾replyDeleteVoteConfiguration {
⇄   string: string
  }
```

CM Module

Version 1.0.0

Contact information
pvalente@onesource.pt (mailto:pvalente@onesource.pt)

Paths

`/content/{content_key}`

`GET /content/{content_key}`

Content Method

Summary

Show Content Method

Parameters

Name	Located in	Description	Required	Schema
content_key	path	Content Key	Yes	⇄ string
X-MODULE-TOKEN	header	Module Token	Yes	⇄ string
LANG-CODE-DEFAULT	header	Module Token	Yes	⇄ string

Responses

Code	Description	Schema
200	Show the Content data	⇄

```
▾contentReply {
  all of:
    ▾ {
      id:           integer
      content_key:  string
      type_id:      integer
      fixed:        integer
      clean:        integer
      published:    integer
      start_date:   string
      end_date:     string
      publish_date: string
      title:        string
      summary:      string
      content:      string
      content_id:   integer
      version:      integer
      link:         string
      enabled:      integer
      created_by:   string
      docs_main:    integer
      docs_side:    integer
      highlight:    integer
      slideshow:    integer
    }
}
```

Code	Description	Schema
401	Unauthorized	⇄

```
▾contentsMethodErrorDefault {
  error: string *
}
```

Code	Description	Schema
404	Content not Found | No translation found	⇄

```
▾contentsMethodErrorDefault {
  error: string *
}
```

Code	Description	Schema
500	Failed to retrieve the content	⇄

```
▾contentsMethodErrorDefault {
  error: string *
}
```

Try this operation

`PUT /content/{content_key}`

Content Method

Summary

Update a Content Method

Parameters

Name	Located in	Description	Required	Schema
content	body	Content Method data	Yes	⇄

```
▾contentUpdate {
  all of:
    ▾ {
        type:          string
        fixed:         integer
        clean:         integer
        published:     integer
        translations: ▾[
                        ▸
                        translations
                        { }
                       ]
      }
}
```

content_key	path	Content Key	Yes	⇄ string
X-AUTH-TOKEN	header	User Auth Token	Yes	⇄ string
X-MODULE-TOKEN	header	Module Token	Yes	⇄ string

Responses

Code	Description	Schema
200	The updated Content Method	⇄

```
▾contentUpdateReply {
  all of:
    ▾ {
        content: ▾[
                   ▸content { }
                  ]
        version: integer
      }
}
```

401	Unauthorized	⇄ ▾contentsMethodErrorDefault {
		error: string *
		}

404	Content not Found	⇄ ▾contentsMethodErrorDefault {
		error: string *
		}

500	Failed to update a Content	⇄ ▾contentsMethodErrorDefault {
		error: string *
		}

Try this operation

`DELETE /content/{content_key}`

Content Method

Summary

Delete Content Method

Parameters

Name	Located in	Description	Required	Schema
content_key	path	Content Key	Yes	⇄ `string`
X-MODULE-TOKEN	header	Module Token	Yes	⇄ `string`
X-AUTH-TOKEN	header	User Auth Token	Yes	⇄ `string`

Responses

Code	Description	Schema

200	OK	⇄ ▾**`contentDeleteReply {`**
		`string: string`
		`}`

401	Unauthorized	⇄ ▾**`contentsMethodErrorDefault {`**
		`error: string *`
		`}`

404	Content not Found	⇄ ▾**`contentsMethodErrorDefault {`**
		`error: string *`
		`}`

500	Failed to delete Content	⇄ ▾**`contentsMethodErrorDefault {`**
		`error: string *`
		`}`

Try this operation

`/content`

`POST /content`

Content Method

Summary

Create a Content Method

Parameters

Name	Located in	Description	Required	Schema
content	body	Content Method data	Yes	⇄
X-AUTH-TOKEN	header	User Auth Token	Yes	⇄ `string`
X-MODULE-TOKEN	header	Module Token	Yes	⇄ `string`

```
▾contentCreate {
 all of:
   ▾ {
      type:          string *
      fixed:         integer
      clean:         integer
      published:     integer
      translations:  ▾[
                        ▸
                        translations
                        { }
                     ]
   }
}
```

Responses

Code	Description	Schema
201	the newly created Content Method	⇄
401	Unauthorized	⇄
404	Content Type not Found	⇄

```
▾contentCreateReply {
 all of:
   ▾ {
      id:            integer
      content_key:   string
      type_id:       integer
      fixed:         integer
      clean:         integer
      start_date:    string
      end_date:      string
      publish_date:  string
   }
}
```

```
▾contentsMethodErrorDefault {
  error: string *
}
```

```
▾contentsMethodErrorDefault {
  error: string *
}
```

Code **Description** **Schema**

500 Failed to store new Content ⇄

```
▾contentsMethodErrorDefault {
   error: string *
}
```

Try this operation

/menutype/{menu_type_id}

GET /menutype/{menu_type_id}

Menu Type Method

Summary

Show a Menu Type Method

Parameters

Name	Located in	Description	Required	Schema
menu_type_id	path	Menu Type Method Id	Yes	⇄ `integer`
X-MODULE-TOKEN	header	Module Token	Yes	⇄ `string`

Responses

Code **Description** **Schema**

200 Show the Menu Type data ⇄

```
▾menuTypesReply {
   all of:
     ▾ {
        id:         integer
        type:       string
        module:     string
        title:      string
        created_at: string
        updated_at: string
     }
}
```

401 Unauthorized ⇄

```
▾menuTypesMethodErrorDefault {
   error: string *
}
```

404 Menu type not Found ⇄

```
▾menuTypesMethodErrorDefault {
   error: string *
}
```

Try this operation

/menu/{menu_key}

GET /menu/{menu_key}

Menu Method

Summary

Show a Menu Method

Parameters

Name	Located in	Description	Required	Schema
menu_key	path	Menu Method Key	Yes	⇄ string
X-MODULE-TOKEN	header	Module Token	Yes	⇄ string
LANG-CODE-DEFAULT	header	Default Laguage Code	Yes	⇄ string

Responses

Code **Description** **Schema**

200 Show the Menu data ⇄

```
▾menusShowReply {
  all of:
    ▾ {
      id:         integer
      menu_key:   string
      access_id:  integer
      parent_id:  integer
      position:   integer
      type_id:    integer
      page_id:    integer
      type:       string
      value:      string
      title:      string
      link:       string
      created_at: string
      updated_at: string
      menu_types: ▾[
                    ▸menuTypesReply
                    { }
                  ]
    }
}
```

401 Unauthorized ⇄

```
▾menusErrorDefault {
  error: string *
}
```

404 Menu not Found | No translation found ⇄

```
▾menusErrorDefault {
  error: string *
}
```

Try this operation

`PUT /menu/{menu_key}`

Menu Method

Summary

Update a Menu Method

Parameters

Name	Located in	Description	Required	Schema
menu	body	Menu Method data	Yes	⇄

```
▾menuCreateUpdate {
   all of:
     ▾ {
        parent_id:     integer *
        access_id:     integer *
        type_id:       integer
        type:          string
        value:         string
        translations:  ▾[
                          ▸
                         menuTranslations
                         { }
                        ]
     }
}
```

Name	Located in	Description	Required	Schema
menu_key	path	Menu Key	Yes	⇄ `string`
X-AUTH-TOKEN	header	User Auth Token	Yes	⇄ `string`
X-MODULE-TOKEN	header	Module Token	Yes	⇄ `string`

Responses

Code	Description	Schema
200	The updated Menu	⇄

```
▾menusCreateUpdateReply {
   all of:
     ▾ {
        id:          integer
        menu_key:    string
        access_id:   integer
        parent_id:   integer
        position:    integer
        type_id:     integer
        type:        string
        value:       string
        created_at:  string
        updated_at:  string
     }
}
```

Code	Description	Schema
400	Failed to update Menu	⇄

```
▾menusErrorDefault {
   error: string *
}
```

Code	Description	Schema

401	Unauthorized	⇄ ▾**menusErrorDefault** {
		error: string *
		}

404	Menu not Found	⇄ ▾**menusErrorDefault** {
		error: string *
		}

Try this operation

`DELETE /menu/{menu_key}`

Menu Method

Summary

Delete menu Method

Parameters

Name	Located in	Description	Required	Schema
menu_key	path	Menu Key	Yes	⇄ string
X-MODULE-TOKEN	header	Module Token	Yes	⇄ string
X-AUTH-TOKEN	header	User Auth Token	Yes	⇄ string

Responses

Code	Description	Schema

200	OK	⇄ ▾**menuDeleteReply** {
		string: string
		}

401	Unauthorized	⇄ ▾**menusErrorDefault** {
		error: string *
		}

404	Menu not Found	⇄ ▾**menusErrorDefault** {
		error: string *
		}

500	Failed to delete Menu	⇄ ▾**menusErrorDefault** {
		error: string *
		}

Try this operation

/menu

POST /menu

Menu Method

Summary

Create a Menu Method

Parameters

Name	Located in	Description	Required	Schema
menu	body	Menu Method data	Yes	⇄
X-AUTH-TOKEN	header	User Auth Token	Yes	⇄ string
X-MODULE-TOKEN	header	Module Token	Yes	⇄ string

```
▾menuCreateUpdate {
  all of:
    ▾ {
      parent_id:     integer *
      access_id:     integer *
      type_id:       integer
      type:          string
      value:         string
      translations:  ▾[
                       ▸
                      menuTranslations
                      { }
                     ]
    }
}
```

Responses

Code	Description	Schema
201	the newly created Menu Method	⇄
400	Failed to store new Menu translations	⇄

```
▾menusCreateUpdateReply {
  all of:
    ▾ {
      id:          integer
      menu_key:    string
      access_id:   integer
      parent_id:   integer
      position:    integer
      type_id:     integer
      type:        string
      value:       string
      created_at:  string
      updated_at:  string
    }
}
```

```
▾menusErrorDefault {
  error: string *
}
```

Code	Description	Schema

401	Unauthorized	⇄ ▾**menusErrorDefault {**
		error: string *
		}

500	Failed to store new Menu	⇄ ▾**menusErrorDefault {**
		error: string *
		}

Try this operation

`/text/{text_id}`

`GET /text/{text_id}`

Text Method

Summary

Show a text Method

Parameters

Name	Located in	Description	Required	Schema
text_id	path	Text Method Id	Yes	⇄ `integer`
X-MODULE-TOKEN	header	Module Token	Yes	⇄ `string`

Responses

Code	Description	Schema

200	Show the Text data	⇄ ▾**textsReply {**
		all of:
		▾ {
		id: integer
		key: string
		created_at: string
		updated_at: string
		}
		}

401	Unauthorized	⇄ ▾**textsMethodErrorDefault {**
		error: string *
		}

404	Text not Found	⇄ ▾**textsMethodErrorDefault {**
		error: string *
		}

500	Failed to retrieve the Text	⇄ ▾**textsMethodErrorDefault {**
		error: string *
		}

Try this operation

PUT /text/{text_id}

Text Method

Summary

Update a text Method

Parameters

Name	Located in	Description	Required	Schema
text	body	Text Method data	Yes	⇄
text_id	path	Text Id	Yes	⇄ integer
X-AUTH-TOKEN	header	User Auth Token	Yes	⇄ string
X-MODULE-TOKEN	header	Module Token	Yes	⇄ string

text

```
▾textUpdate {
  all of:
    ▾ {
        entity_id:     integer *
        translations: ▾[
                          ▸
                          textTranslations
                          { }
                       ]
      }
}
```

Responses

Code	Description	Schema
200	The updated text	⇄
401	Unauthorized	⇄
404	Text not Found	⇄
500	Failed to update text	⇄

200

```
▾textsReply {
  all of:
    ▾ {
        id:         integer
        key:        string
        created_at: string
        updated_at: string
      }
}
```

401

```
▾textsMethodErrorDefault {
   error: string *
}
```

404

```
▾textsMethodErrorDefault {
   error: string *
}
```

500

```
▾textsMethodErrorDefault {
   error: string *
}
```

Try this operation

`DELETE /text/{text_id}`

Text Method

Summary

Delete text Method

Parameters

Name	Located in	Description	Required	Schema
text_id	path	Text Id	Yes	⇄ `integer`
X-MODULE-TOKEN	header	Module Token	Yes	⇄ `string`
X-AUTH-TOKEN	header	User Auth Token	Yes	⇄ `string`

Responses

Code	Description	Schema

200	OK	⇄ ▾**`textDeleteReply {`**
		`string: string`
		`}`

401	Unauthorized	⇄ ▾**`textsMethodErrorDefault {`**
		`error: string *`
		`}`

404	text not Found	⇄ ▾**`textsMethodErrorDefault {`**
		`error: string *`
		`}`

500	Failed to delete text	⇄ ▾**`textsMethodErrorDefault {`**
		`error: string *`
		`}`

Try this operation

`/text`

`POST /text`

Text Method

Summary

Create a text Method

Parameters

Name	Located in	Description	Required	Schema

text	body	text Method data	Yes	⇄ ▾**textCreate {**
				all of:
				▾ **{**
				entity_id: integer *
				translations: ▾**[**
				▸ **textTranslations { }**
				]
				}
				}

X-AUTH-TOKEN	header	User Auth Token	Yes	⇄ `string`
X-MODULE-TOKEN	header	Module Token	Yes	⇄ `string`

Responses

Code	Description	Schema
201	the newly created text Method	⇄ ▾**textsReply {** **all of:** ▾ **{** id: integer key: string created_at: string updated_at: string } }

401	Unauthorized	⇄ ▾**textsMethodErrorDefault {**
		error: string *
		}

500	Failed to store new text	⇄ ▾**textsMethodErrorDefault {**
		error: string *
		}

Try this operation

Models

contentsMethodErrorDefault

```
    ▾contentsMethodErrorDefault {
⇄     error: string *
    }
```

contentCreate

```
    ▾contentCreate {
      all of:
        ▾ {
          type:          string *
          fixed:         integer
          clean:         integer
          published:     integer
          translations: ▾[
                          ▾translations {
                            all of:
                              ▾ {
                                language_code: string *
                                title:         string *
⇄                               summary:       string
                                content:       string
                                version:       integer
                                enabled:       integer
                                link:          string
                                docs_main:     integer
                                docs_side:     integer
                                highlight:     integer
                                slideshow:     integer
                              }
                          }
                        ]
        }
    }
```

contentUpdate

```
▾contentUpdate {
  all of:
    ▾ {
      type:          string
      fixed:         integer
      clean:         integer
      published:     integer
      translations: ▾[
                      ▾translations {
                        all of:
                          ▾ {
                            language_code: string *
                            title:         string *
                            summary:       string
                            content:       string
                            version:       integer
                            enabled:       integer
                            link:          string
                            docs_main:     integer
                            docs_side:     integer
                            highlight:     integer
                            slideshow:     integer
                          }
                      }
                    ]
    }
}
```

contentReply

```
  ▾contentReply {
    all of:
      ▾ {
         id:           integer
         content_key:  string
         type_id:      integer
         fixed:        integer
         clean:        integer
         published:    integer
         start_date:   string
         end_date:     string
         publish_date: string
         title:        string
⇄        summary:      string
         content:      string
         content_id:   integer
         version:      integer
         link:         string
         enabled:      integer
         created_by:   string
         docs_main:    integer
         docs_side:    integer
         highlight:    integer
         slideshow:    integer
      }
  }
```

contentCreateReply

```
  ▾contentCreateReply {
    all of:
      ▾ {
         id:           integer
         content_key:  string
         type_id:      integer
⇄        fixed:        integer
         clean:        integer
         start_date:   string
         end_date:     string
         publish_date: string
      }
  }
```

contentDeleteReply

```
  ▾contentDeleteReply {
⇄    string: string
  }
```

contentUpdateReply

```
   ▾contentUpdateReply {
     all of:
        ▾ {
          content: ▾[
                      ▾content {
                        all of:
                           ▾ {
                             id:           integer
                             content_key:  string
                             type_id:      integer
⇄                            fixed:        integer
                             clean:        integer
                             start_date:   string
                             end_date:     string
                             publish_date: string
                           }
                      }
                    ]
          version: integer
        }
   }
```

content

```
   ▾content {
     all of:
        ▾ {
          id:           integer
          content_key:  string
          type_id:      integer
⇄         fixed:        integer
          clean:        integer
          start_date:   string
          end_date:     string
          publish_date: string
        }
   }
```

translations

```
  ▾translations {
    all of:
      ▾ {
        language_code: string *
        title:         string *
        summary:       string
        content:       string
        version:       integer
⇄       enabled:       integer
        link:          string
        docs_main:     integer
        docs_side:     integer
        highlight:     integer
        slideshow:     integer
      }
  }
```

menuTypesMethodErrorDefault

```
  ▾menuTypesMethodErrorDefault {
⇄   error: string *
  }
```

menuTypesReply

```
  ▾menuTypesReply {
    all of:
      ▾ {
        id:         integer
        type:       string
⇄       module:     string
        title:      string
        created_at: string
        updated_at: string
      }
  }
```

menusErrorDefault

```
  ▾menusErrorDefault {
⇄   error: string *
  }
```

menusShowReply

```
▾menusShowReply {
  all of:
    ▾ {
      id:          integer
      menu_key:    string
      access_id:   integer
      parent_id:   integer
      position:    integer
      type_id:     integer
      page_id:     integer
      type:        string
      value:       string
      title:       string
      link:        string
      created_at:  string
      updated_at:  string
      menu_types:  ▾[
                     ▾menuTypesReply {
                       all of:
                         ▾ {
                           id:          integer
                           type:        string
                           module:      string
                           title:       string
                           created_at:  string
                           updated_at:  string
                         }
                     }
                   ]
    }
}
```

menuCreateUpdate

```
  ▾menuCreateUpdate {
    all of:
      ▾ {
        parent_id:    integer *
        access_id:    integer *
        type_id:      integer
        type:         string
        value:        string
        translations: ▾[
⇄                        ▾menuTranslations {
                           all of:
                             ▾ {
                               language_code: string *
                               title:         string *
                               link:          string
                             }
                         }
                       ]
      }
  }
```

menuDeleteReply

```
  ▾menuDeleteReply {
⇄    string: string
  }
```

menusCreateUpdateReply

```
  ▾menusCreateUpdateReply {
    all of:
      ▾ {
        id:         integer
        menu_key:   string
        access_id:  integer
        parent_id:  integer
⇄       position:   integer
        type_id:    integer
        type:       string
        value:      string
        created_at: string
        updated_at: string
      }
  }
```

menuTranslations

```
    ▾menuTranslations {
      all of:
         ▾ {
           language_code: string *
⇄          title:         string *
           link:          string
         }
    }
```

textsMethodErrorDefault

```
    ▾textsMethodErrorDefault {
⇄     error: string *
    }
```

textsReply

```
    ▾textsReply {
      all of:
         ▾ {
           id:         integer
⇄          key:        string
           created_at: string
           updated_at: string
         }
    }
```

textCreate

```
    ▾textCreate {
      all of:
         ▾ {
           entity_id:    integer *
           translations: ▾[
                            ▾textTranslations {
                              all of:
                                 ▾ {
⇄                                  language_code: string *
                                   title:         string *
                                   content:       string
                                   tag:           string
                                 }
                            }
                         ]
         }
    }
```

textUpdate

```
   ▾textUpdate {
     all of:
       ▾ {
         entity_id:    integer *
         translations: ▾[
                         ▾textTranslations {
                           all of:
                             ▾ {
⇄                              language_code: string *
                               title:         string *
                               content:       string
                               tag:           string
                             }
                         }
                       ]
       }
   }
```

textDeleteReply

```
   ▾textDeleteReply {
⇄     string: string
   }
```

textTranslations

```
   ▾textTranslations {
     all of:
       ▾ {
         language_code: string *
⇄        title:         string *
         content:       string
         tag:           string
       }
   }
```

Events Module

Version 1.0.0

Contact information
pvalente@onesource.pt (mailto:pvalente@onesource.pt)

Paths

`/event/{event_key}`

`GET /event/{event_key}`

Event Method

Summary

Shows a Event Method

Parameters

Name	Located in	Description	Required	Schema
event_key	path	Event Method Key	Yes	⇄ `string`
X-MODULE-TOKEN	header	Module Token	Yes	⇄ `string`

Responses

Code	Description	Schema
200	Shows the Event data	⇄

```
▾eventsReplyWithTranslations {
  all of:
    ▾ {
      event_key:     string
      file_id:       string
      created_by:    string
      start_date:    string
      end_date:      string
      created_at:    string
      updated_at:    string
      translations: ▾[
                       ▾eventTranslations {
                         all of:
                           ▾ {
                             language_code: string
                                            *
                             name:          string
                                            *
                             description:   string
                             footer:        string
                           }
                       }
                     ]
    }
}
```

Code	Description	Schema
401	Unauthorized	⇄

```
▾eventsMethodErrorDefault {
  error: string *
}
```

Code	Description	Schema
404	Event not Found	⇄

```
▾eventsMethodErrorDefault {
  error: string *
}
```

Code	Description	Schema
500	Failed to retrieve Event	⇄

```
▾eventsMethodErrorDefault {
  error: string *
}
```

Try this operation

PUT /event/{event_key}

Event Method

Summary

Update a event Method

Parameters

Name	Located in	Description	Required	Schema
event	body	Event Method data	Yes	⇄
event_key	path	Event Key	Yes	⇄ string
X-AUTH-TOKEN	header	User Auth Token	Yes	⇄ string
X-MODULE-TOKEN	header	Module Token	Yes	⇄ string

event:

```
▾eventCreateUpdate {
  all of:
    ▾ {
      file_id:       string
      start_date:    string
                     *
      end_date:      string
                     *
      translations:  ▸[]
    }
}
```

Responses

Code	Description	Schema
200	The updated Event	⇄

200:

```
▾eventsReplyWithTranslations {
  all of:
    ▾ {
      event_key:     string
      file_id:       string
      created_by:    string
      start_date:    string
      end_date:      string
      created_at:    string
      updated_at:    string
      translations:  ▾[
                       ▾eventTranslations {
                         all of:
                           ▾ {
                             language_code: string
                                            *
                             name:          string
                                            *
                             description:   string
                             footer:        string
                           }
                       }
                     ]
    }
}
```

Code	Description	Schema

401	Unauthorized	⇄ ▾**eventsMethodErrorDefault {**
		`error: string *`
		}

404	event not Found	⇄ ▾**eventsMethodErrorDefault {**
		`error: string *`
		}

500	Failed to Update Event	⇄ ▾**eventsMethodErrorDefault {**
		`error: string *`
		}

Try this operation

`DELETE /event/{event_key}`

Event Method

Summary

Delete Event Method

Parameters

Name	Located in	Description	Required	Schema
event_key	path	Event Key	Yes	⇄ `string`
X-MODULE-TOKEN	header	Module Token	Yes	⇄ `string`
X-AUTH-TOKEN	header	User Auth Token	Yes	⇄ `string`

Responses

Code	Description	Schema

200	OK	⇄ ▾**eventDeleteReply {**
		`string: string`
		}

401	Unauthorized	⇄ ▾**eventsMethodErrorDefault {**
		`error: string *`
		}

404	Event not Found	⇄ ▾**eventsMethodErrorDefault {**
		`error: string *`
		}

500	Failed to delete Event	⇄ ▾**eventsMethodErrorDefault {**
		`error: string *`
		}

Try this operation

`/event`

`POST /event`

Event Method

Summary

Create a event Method

Parameters

Name	Located in	Description	Required	Schema
event	body	Event Method data	Yes	⇄ eventCreateUpdate (below)
X-AUTH-TOKEN	header	User Auth Token	Yes	⇄ string
X-MODULE-TOKEN	header	Module Token	Yes	⇄ string

event:

```
▾eventCreateUpdate {
  all of:
    ▾ {
       file_id:       string
       start_date:    string *
       end_date:      string *
       translations: ▾[
                        ▸
                        eventTranslations
                        { }
                      ]
    }
}
```

Responses

Code	Description	Schema

201 the newly created Event ⇄

```
▾eventsReplyWithTranslations {
  all of:
    ▾ {
      event_key:    string
      file_id:      string
      created_by:   string
      start_date:   string
      end_date:     string
      created_at:   string
      updated_at:   string
      translations: ▾[
                      ▾eventTranslations {
                        all of:
                          ▾ {
                            language_code: string
                                           *
                            name:          string
                                           *
                            description:   string
                            footer:        string
                          }
                      }
                    ]
    }
}
```

401 Unauthorized ⇄

```
▾eventsMethodErrorDefault {
  error: string *
}
```

500 Failed to store new Event ⇄

```
▾eventsMethodErrorDefault {
  error: string *
}
```

Try this operation

/item/{item_key}

`GET /item/{item_key}`

Item Method

Summary

Shows a Item Method

Parameters

Name	Located in	Description	Required	Schema
item_key	path	Item Method Id	Yes	⇄ `string`
X-MODULE-TOKEN	header	Module Token	Yes	⇄ `string`

Responses

Code	Description	Schema
200	Shows the Item data	⇄
401	Unauthorized	⇄
404	Item not Found	⇄
500	Failed to retrieve Item	⇄

200

```
▾itemsReply {
  all of:
     ▾ {
       id:         integer
       item_key:   string
       sessions:   string
       value:      string
       event_id:   integer
       created_at: string
       updated_at: string
     }
}
```

401

```
▾itemsMethodErrorDefault {
  error: string *
}
```

404

```
▾itemsMethodErrorDefault {
  error: string *
}
```

500

```
▾itemsMethodErrorDefault {
  error: string *
}
```

Try this operation

`PUT /item/{item_key}`

Item Method

Summary

Update a Item Method

Parameters

Name	Located in	Description	Required	Schema
item	body	Item Method data	Yes	⇄
item_key	path	Item Key	Yes	⇄ string
X-AUTH-TOKEN	header	User Auth Token	Yes	⇄ string
X-MODULE-TOKEN	header	Module Token	Yes	⇄ string

item:

```
▾itemUpdate {
  all of:
    ▾ {
        sessions:  ▾[
                     string
                   ]
        value:     string *
    }
}
```

Responses

Code	Description	Schema
200	The Updated Item	⇄
401	Unauthorized	⇄
500	Failed to Update Item	⇄

200:

```
▾itemsReply {
  all of:
    ▾ {
        id:         integer
        item_key:   string
        sessions:   string
        value:      string
        event_id:   integer
        created_at: string
        updated_at: string
    }
}
```

401:

```
▾itemsMethodErrorDefault {
  error: string *
}
```

500:

```
▾itemsMethodErrorDefault {
  error: string *
}
```

Try this operation

`DELETE /item/{item_key}`

Item Method

Summary

Delete Item Method

Parameters

Name	Located in	Description	Required	Schema
item_key	path	Item Key	Yes	⇄ `string`
X-MODULE-TOKEN	header	Module Token	Yes	⇄ `string`
X-AUTH-TOKEN	header	User Auth Token	Yes	⇄ `string`

Responses

Code	Description	Schema

200	OK	⇄ ▾**itemDeleteReply** {
		`string: string`
		}

401	Unauthorized	⇄ ▾**itemsMethodErrorDefault** {
		`error: string *`
		}

404	Item not Found	⇄ ▾**itemsMethodErrorDefault** {
		`error: string *`
		}

500	Failed to delete Item	⇄ ▾**itemsMethodErrorDefault** {
		`error: string *`
		}

Try this operation

`/item`

POST /item

Item Method

Summary

Create a Item Method

Parameters

Name	Located in	Description	Required	Schema
item	body	Item Method data	Yes	⇄

```
▾itemCreate {
  all of:
    ▾ {
        event_key: string *
        sessions:   ▾[
                      string
                    ]
        value:      string *
      }
  }
```

Name	Located in	Description	Required	Schema
X-AUTH-TOKEN	header	User Auth Token	Yes	⇄ string
X-MODULE-TOKEN	header	Module Token	Yes	⇄ string

Responses

Code	Description	Schema
201	The newly created Item	⇄

```
▾itemsReply {
  all of:
    ▾ {
        id:         integer
        item_key:   string
        sessions:   string
        value:      string
        event_id:   integer
        created_at: string
        updated_at: string
      }
  }
```

Code	Description	Schema
401	Unauthorized	⇄

```
▾itemsMethodErrorDefault {
  error: string *
}
```

Code	Description	Schema
500	Failed to store new Item	⇄

```
▾itemsMethodErrorDefault {
  error: string *
}
```

Try this operation

/payment/{payment_key}

`GET /payment/{payment_key}`

Payment Method

Summary

Shows a payment Method

Parameters

Name	Located in	Description	Required	Schema
payment_key	path	Payment Method Key	Yes	⇄ string
X-MODULE-TOKEN	header	Module Token	Yes	⇄ string

Responses

Code	Description	Schema
200	Shows the Payment data	⇄

```
▾paymentsReply {
  all of:
     ▾ {
        payment_key:      string
        registration_id: integer
        value:           integer
        paid:            integer
        payment_date:    string
        payment_method:  string
        created_at:      string
        updated_at:      string
     }
}
```

Code	Description	Schema
401	Unauthorized	⇄

```
▾paymentsMethodErrorDefault {
   error: string *
}
```

Code	Description	Schema
404	Payment not Found	⇄

```
▾paymentsMethodErrorDefault {
   error: string *
}
```

Code	Description	Schema
500	Failed to retrieve Payment	⇄

```
▾paymentsMethodErrorDefault {
   error: string *
}
```

Try this operation

`PUT /payment/{payment_key}`

Payment Method

Summary

Update a payment Method

Parameters

Name	Located in	Description	Required	Schema
payment	body	Payment Method data	Yes	⇄ paymentUpdate
payment_key	path	payment Key	Yes	⇄ string
X-AUTH-TOKEN	header	User Auth Token	Yes	⇄ string
X-MODULE-TOKEN	header	Module Token	Yes	⇄ string

payment:

```
▾paymentUpdate {
  all of:
    ▾ {
        registration_id: integer
                         *
        value:           integer
                         *
        paid:            integer
                         *
        payment_date:    string
                         *
        payment_method:  string
                         *
      }
  }
```

Responses

Code	Description	Schema
200	The Updated Payment	⇄ paymentsReply
401	Unauthorized	⇄ paymentsMethodErrorDefault
500	Failed to Update Payment	⇄ paymentsMethodErrorDefault

200

```
▾paymentsReply {
  all of:
    ▾ {
        payment_key:     string
        registration_id: integer
        value:           integer
        paid:            integer
        payment_date:    string
        payment_method:  string
        created_at:      string
        updated_at:      string
      }
  }
```

401

```
▾paymentsMethodErrorDefault {
   error: string *
}
```

500

```
▾paymentsMethodErrorDefault {
   error: string *
}
```

Try this operation

`/payment`

`POST /payment`

Payment Method

Summary

Create a Payment Method

Parameters

Name	Located in	Description	Required	Schema
payment	body	Payment Method data	Yes	⇄

```
▾paymentCreate {
  all of:
    ▾ {
      registration_key: string
                        *
      value:            integer
                        *
      paid:             integer
                        *
      payment_date:     string
                        *
      payment_method:   string
                        *
    }
}
```

Name	Located in	Description	Required	Schema
X-AUTH-TOKEN	header	User Auth Token	Yes	⇄ `string`
X-MODULE-TOKEN	header	Module Token	Yes	⇄ `string`

Responses

Code	Description	Schema
201	The newly created Payment	⇄

```
▾paymentsReply {
  all of:
    ▾ {
      payment_key:     string
      registration_id: integer
      value:           integer
      paid:            integer
      payment_date:    string
      payment_method:  string
      created_at:      string
      updated_at:      string
    }
}
```

Code	Description	Schema
401	Unauthorized	⇄

```
▾paymentsMethodErrorDefault {
  error: string *
}
```

Code	Description	Schema
500	Failed to store new Payment	⇄

```
▾paymentsMethodErrorDefault {
  error: string *
}
```

Try this operation

/payment/{payment_id}

DELETE /payment/{payment_id}

Payment Method

Summary

Delete Payment Method

Parameters

Name	Located in	Description	Required	Schema
payment_id	path	Payment Id	Yes	⇄ integer
X-MODULE-TOKEN	header	Module Token	Yes	⇄ string
X-AUTH-TOKEN	header	User Auth Token	Yes	⇄ string

Responses

Code	Description	Schema

200	OK	⇄ ▾**paymentDeleteReply {**
		string: string
		}

401	Unauthorized	⇄ ▾**paymentsMethodErrorDefault {**
		error: string *
		}

404	Payment not Found	⇄ ▾**paymentsMethodErrorDefault {**
		error: string *
		}

500	Failed to delete Payment	⇄ ▾**paymentsMethodErrorDefault {**
		error: string *
		}

Try this operation

/registration/{registration_key}

GET /registration/{registration_key}

Registration Method

Summary

Shows a registration Method

Parameters

Name	Located in	Description	Required	Schema
registration_key	path	Registration Method Key	Yes	⇄ string
X-MODULE-TOKEN	header	Module Token	Yes	⇄ string

Responses

Code **Description** **Schema**

200 Shows the Registration data ⇄

```
▾registrationsShowReply {
  all of:
    ▾ {
      id:               integer
      registration_key: string
      user_key:         string
      event_id:         integer
      name:             string
      email:            string
      vat:              integer
      created_at:       string
      updated_at:       string
      event:            ▾ {
                          all of:
                            ▾eventsReply {
                              all of:
                                ▾ {
                                  event_key:  string
                                  file_id:    string
                                  created_by: string
                                  start_date: string
                                  end_date:   string
                                  created_at: string
                                  updated_at: string
                                }
                            }
                        }
      sessions:         ▾[
                          string
                        ]
    }
}
```

401 Unauthorized ⇄

```
▾registrationsMethodErrorDefault {
   error: string *
}
```

404 Registration not Found ⇄

```
▾registrationsMethodErrorDefault {
   error: string *
}
```

Code	Description	Schema
500	Failed to retrieve Registration	⇄

```
▾registrationsMethodErrorDefault {
   error: string *
}
```

Try this operation

/registration

`POST /registration`

Registration Method

Summary

Create a Registration Method

Parameters

Name	Located in	Description	Required	Schema
registration	body	Registration Method data	Yes	⇄ registrationCreate
X-AUTH-TOKEN	header	User Auth Token	No	⇄ `string`
X-MODULE-TOKEN	header	Module Token	Yes	⇄ `string`

registration:

```
▾registrationCreate {
   all of:
       ▾ {
         event_key: string
         name:      string
         email:     string
         vat:       integer
         sessions:  ▾[
                       string
                    ]
       }
}
```

Responses

Code	Description	Schema

201 The newly created Registration ⇄

```
▾registrationsCreateReply {
  all of:
    ▾ {
      registration_key: string
      user_key:         string
      event_id:         integer
      name:             string
      email:            string
      vat:              integer
      created_at:       string
      updated_at:       string
      event:            ▾ {
                          all of:
                            ▾eventsReply {
                              all of:
                                ▾ {
                                  event_key:  string
                                  file_id:    string
                                  created_by: string
                                  start_date: string
                                  end_date:   string
                                  created_at: string
                                  updated_at: string
                                }
                            }
                        }
    }
}
```

401 Unauthorized ⇄

```
▾registrationsMethodErrorDefault {
  error: string *
}
```

500 Failed to store new Registration ⇄

```
▾registrationsMethodErrorDefault {
  error: string *
}
```

Try this operation

/registration/{registration_id}

`DELETE /registration/{registration_id}`

Registration Method

Summary

Delete a Registration Method

Parameters

Name	Located in	Description	Required	Schema
registration_id	path	Registration Id	Yes	⇄ `integer`
X-MODULE-TOKEN	header	Module Token	Yes	⇄ `string`
X-AUTH-TOKEN	header	User Auth Token	Yes	⇄ `string`

Responses

Code	Description	Schema

200	OK	⇄ ▾`registrationDeleteReply {`
		`string: string`
		`}`

401	Unauthorized	⇄ ▾`registrationsMethodErrorDefault {`
		`error: string *`
		`}`

404	Registration not Found	⇄ ▾`registrationsMethodErrorDefault {`
		`error: string *`
		`}`

500	Failed to delete Registration	⇄ ▾`registrationsMethodErrorDefault {`
		`error: string *`
		`}`

Try this operation

`/session/{session_key}`

`GET /session/{session_key}`

Session Method

Summary

Shows a session Method

Parameters

Name	Located in	Description	Required	Schema
session_key	path	Session Method Key	Yes	⇄ string
X-MODULE-TOKEN	header	Module Token	Yes	⇄ string

Responses

Code	Description	Schema
200	Shows the Session data	⇄

```
▾sessionsReply {
  all of:
    ▾ {
      session_key:  string
      event_id:     integer
      created_by:   string
      created_at:   string
      updated_at:   string
      translations: ▾[
                      ▾sessionTranslations {
                        all of:
                          ▾ {
                            session_id:    integer
                            language_code: string
                                           *
                            name:          string
                                           *
                            description:   string
                            created_at:    string
                            updated_at:    string
                          }
                      }
                    ]
      schedules:    ▾[
                      ▾schedules {
                        all of:
                          ▾ {
                            schedule_key: string
                            session_id:   integer
                            start_date:   string *
                            end_date:     string *
                            start_time:   string *
                            end_time:     string *
                            created_at:   string
                            updated_at:   string
                          }
                      }
                    ]
    }
}
```

Code	Description		Schema

401	Unauthorized	⇄	▾ **`sessionsMethodErrorDefault {`**
			`error: string *`
			`}`

404	Session not Found	⇄	▾ **`sessionsMethodErrorDefault {`**
			`error: string *`
			`}`

500	Failed to retrieve Session	⇄	▾ **`sessionsMethodErrorDefault {`**
			`error: string *`
			`}`

Try this operation

PUT /session/{session_key}

Session Method

Summary

Update a session Method

Parameters

Name	Located in	Description	Required	Schema
session	body	Session Method data	Yes	⇄
session_key	path	Session Key	Yes	⇄ string
X-AUTH-TOKEN	header	User Auth Token	Yes	⇄ string
X-MODULE-TOKEN	header	Module Token	Yes	⇄ string

session:

```
▾sessionUpdate {
  all of:
    ▾ {
        translations:  ▸
                       []
        schedules:     ▸
                       []
      }
}
```

Responses

Code	Description	Schema
200	The Updated Session	⇄
401	Unauthorized	⇄
500	Failed to Update Session	⇄

200:

```
▾sessionsReply {
  all of:
    ▾ {
        session_key:   string
        event_id:      integer
        created_by:    string
        created_at:    string
        updated_at:    string
        translations:  ▾[
                          ▸sessionTranslations {
                         }
                       ]
        schedules:     ▾[
                          ▸schedules { }
                       ]
      }
}
```

401:

```
▾sessionsMethodErrorDefault {
  error: string *
}
```

500:

```
▾sessionsMethodErrorDefault {
  error: string *
}
```

Try this operation

`DELETE /session/{session_key}`

Session Method

Summary

Delete session Method

Parameters

Name	Located in	Description	Required	Schema
session_key	path	Session Key	Yes	⇄ `string`
X-MODULE-TOKEN	header	Module Token	Yes	⇄ `string`
X-AUTH-TOKEN	header	User Auth Token	Yes	⇄ `string`

Responses

Code	Description	Schema

200	OK	⇄ ▾**`sessionDeleteReply {`**
		`string: string`
		`}`

401	Unauthorized	⇄ ▾**`sessionsMethodErrorDefault {`**
		`error: string *`
		`}`

404	Session not Found	⇄ ▾**`sessionsMethodErrorDefault {`**
		`error: string *`
		`}`

500	Failed to delete Session	⇄ ▾**`sessionsMethodErrorDefault {`**
		`error: string *`
		`}`

Try this operation

`/session`

`POST /session`

Session Method

Summary

Create a Session Method

Parameters

Name	Located in	Description	Required	Schema
session	body	Session Method data	Yes	⇄
X-AUTH-TOKEN	header	User Auth Token	Yes	⇄ `string`
X-MODULE-TOKEN	header	Module Token	Yes	⇄ `string`

session:

```
▾sessionCreate {
  all of:
    ▾ {
        event_key:     string
                       *
        translations:  ▸[]
        schedules:     ▸[]
      }
}
```

Responses

Code	Description	Schema
201	The newly created Session	⇄
401	Unauthorized	⇄
500	Failed to store new Session	⇄

201:

```
▾sessionsReply {
  all of:
    ▾ {
        session_key:  string
        event_id:     integer
        created_by:   string
        created_at:   string
        updated_at:   string
        translations: ▾[
                        ▸sessionTranslations {
                        }
                      ]
        schedules:    ▾[
                        ▸schedules { }
                      ]
      }
}
```

401:

```
▾sessionsMethodErrorDefault {
   error: string *
}
```

500:

```
▾sessionsMethodErrorDefault {
   error: string *
}
```

Try this operation

`/speaker/{speaker_key}`

`GET /speaker/{speaker_key}`

Speaker Method

Summary

Shows a speaker Method

Parameters

Name	Located in	Description	Required	Schema
speaker_key	path	Speaker Method Key	Yes	⇄ `string`
X-MODULE-TOKEN	header	Module Token	Yes	⇄ `string`

Responses

Code **Description** **Schema**

200 Shows the Speaker data ⇄

```
▾speakersReply {
  all of:
    ▾ {
       speaker_key: string
       session_id:  integer
       name:        string
       company:     string
       nationality: string
       profession:  string
       age:         integer
       created_at:  string
       updated_at:  string
     }
}
```

401 Unauthorized ⇄

```
▾speakersMethodErrorDefault {
   error: string *
}
```

404 Speaker not Found ⇄

```
▾speakersMethodErrorDefault {
   error: string *
}
```

500 Failed to retrieve Speaker ⇄

```
▾speakersMethodErrorDefault {
   error: string *
}
```

Try this operation

PUT /speaker/{speaker_key}

Speaker Method

Summary

Update a speaker Method

Parameters

Name	Located in	Description	Required	Schema
speaker	body	Speaker Method data	Yes	⇄
speaker_key	path	speaker Key	Yes	⇄ `string`
X-AUTH-TOKEN	header	User Auth Token	Yes	⇄ `string`
X-MODULE-TOKEN	header	Module Token	Yes	⇄ `string`

speaker

```
▾speakerUpdate {
  all of:
    ▾ {
        name:         string
                      *
        company:      string
        nationality:  string
        profession:   string
        age:          integer
      }
}
```

Responses

Code	Description	Schema
200	The Updated Speaker	⇄
401	Unauthorized	⇄
500	Failed to Update Speaker	⇄

200

```
▾speakersReply {
  all of:
    ▾ {
        speaker_key: string
        session_id:  integer
        name:        string
        company:     string
        nationality: string
        profession:  string
        age:         integer
        created_at:  string
        updated_at:  string
      }
}
```

401

```
▾speakersMethodErrorDefault {
   error: string *
}
```

500

```
▾speakersMethodErrorDefault {
   error: string *
}
```

Try this operation

`DELETE /speaker/{speaker_key}`

Speaker Method

Summary

Delete speaker Method

Parameters

Name	Located in	Description	Required	Schema
speaker_key	path	Speaker Key	Yes	⇄ `string`
X-MODULE-TOKEN	header	Module Token	Yes	⇄ `string`
X-AUTH-TOKEN	header	User Auth Token	Yes	⇄ `string`

Responses

Code	Description	Schema

200	OK	⇄ ▾**speakerDeleteReply** {
		`string: string`
		}

401	Unauthorized	⇄ ▾**speakersMethodErrorDefault** {
		`error: string *`
		}

404	Speaker not Found	⇄ ▾**speakersMethodErrorDefault** {
		`error: string *`
		}

500	Failed to delete Speaker	⇄ ▾**speakersMethodErrorDefault** {
		`error: string *`
		}

Try this operation

`/speaker`

`POST /speaker`

Speaker Method

Summary

Create a Speaker Method

Parameters

Name	Located in	Description	Required	Schema
speaker	body	Speaker Method data	Yes	⇄ speakerCreate
X-AUTH-TOKEN	header	User Auth Token	Yes	⇄ string
X-MODULE-TOKEN	header	Module Token	Yes	⇄ string

speaker:

```
▾speakerCreate {
  all of:
    ▾ {
      session_key: string
                   *
      name:        string
                   *
      company:     string
      nationality: string
      profession:  string
      age:         integer
    }
}
```

Responses

Code	Description	Schema
201	The newly created Speaker	⇄ speakersReply
401	Unauthorized	⇄ speakersMethodErrorDefault
500	Failed to store new Speaker	⇄ speakersMethodErrorDefault

201:

```
▾speakersReply {
  all of:
    ▾ {
      speaker_key: string
      session_id:  integer
      name:        string
      company:     string
      nationality: string
      profession:  string
      age:         integer
      created_at:  string
      updated_at:  string
    }
}
```

401:

```
▾speakersMethodErrorDefault {
  error: string *
}
```

500:

```
▾speakersMethodErrorDefault {
  error: string *
}
```

Try this operation

```
/sponsor/{sponsor_key}
```

`GET /sponsor/{sponsor_key}`

Sponsor Method

Summary

Shows a sponsor Method

Parameters

Name	Located in	Description	Required	Schema
sponsor_key	path	Sponsor Method Key	Yes	⇄ `string`
X-MODULE-TOKEN	header	Module Token	Yes	⇄ `string`

Responses

Code	Description	Schema
200	Shows the Sponsor data	⇄

```
▾sponsorsShowReply {
  all of:
    ▾ {
      sponsor_key: string
      event_id:    integer
      file_id:     string
      name:        string
      description: string
      email:       string
      web_page:    integer
      created_at:  string
      updated_at:  string
      event:       ▾ {
                     all of:
                       ▾eventsReply {
                         all of:
                           ▾ {
                             event_key:  string
                             file_id:    string
                             created_by: string
                             start_date: string
                             end_date:   string
                             created_at: string
                             updated_at: string
                           }
                       }
                   }
    }
}
```

Code	Description	Schema
401	Unauthorized	⇄

```
▾sponsorsMethodErrorDefault {
  error: string *
}
```

Code	Description	Schema
404	Sponsor not Found	⇄

```
▾sponsorsMethodErrorDefault {
  error: string *
}
```

Code	Description	Schema
500	Failed to retrieve Sponsor	⇄

```
▾sponsorsMethodErrorDefault {
  error: string *
}
```

Try this operation

`PUT /sponsor/{sponsor_key}`

Sponsor Method

Summary

Update a sponsor Method

Parameters

Name	Located in	Description	Required	Schema
sponsor	body	Sponsor Method data	Yes	⇄

```
▾sponsorUpdate {
  all of:
    ▾ {
       name:        string
                    *
       file_id:     string
       description: string
       email:       string
       web_page:    integer
     }
 }
```

Name	Located in	Description	Required	Schema
sponsor_key	path	sponsor Key	Yes	⇄ `string`
X-AUTH-TOKEN	header	User Auth Token	Yes	⇄ `string`
X-MODULE-TOKEN	header	Module Token	Yes	⇄ `string`

Responses

Code	Description	Schema
200	The Updated Sponsor	⇄

```
▾sponsorsReply {
  all of:
    ▾ {
       sponsor_key: string
       event_id:    integer
       file_id:     string
       name:        string
       description: string
       email:       string
       web_page:    integer
       created_at:  string
       updated_at:  string
     }
 }
```

Code	Description	Schema
401	Unauthorized	⇄

```
▾sponsorsMethodErrorDefault {
   error: string *
 }
```

Code	Description	Schema
500	Failed to Update Sponsor	⇄

```
▾sponsorsMethodErrorDefault {
   error: string *
 }
```

Try this operation

`DELETE /sponsor/{sponsor_key}`

Sponsor Method

Summary

Delete sponsor Method

Parameters

Name	Located in	Description	Required	Schema
sponsor_key	path	Sponsor Key	Yes	⇄ `string`
X-MODULE-TOKEN	header	Module Token	Yes	⇄ `string`
X-AUTH-TOKEN	header	User Auth Token	Yes	⇄ `string`

Responses

Code **Description** **Schema**

200 OK ⇄

```
▾sponsorDeleteReply {
   string: string
}
```

401 Unauthorized ⇄

```
▾sponsorsMethodErrorDefault {
   error: string *
}
```

404 Sponsor not Found ⇄

```
▾sponsorsMethodErrorDefault {
   error: string *
}
```

500 Failed to delete Sponsor ⇄

```
▾sponsorsMethodErrorDefault {
   error: string *
}
```

Try this operation

/sponsor

POST /sponsor

Sponsor Method

Summary

Create a Sponsor Method

Parameters

Name	Located in	Description	Required	Schema
sponsor	body	Sponsor Method data	Yes	⇄ sponsorCreate
X-AUTH-TOKEN	header	User Auth Token	Yes	⇄ `string`
X-MODULE-TOKEN	header	Module Token	Yes	⇄ `string`

sponsor:

```
▾sponsorCreate {
  all of:
    ▾ {
       event_key:   string
                    *
       name:        string
                    *
       file_id:     string
       description: string
       email:       string
       web_page:    integer
     }
}
```

Responses

Code	Description	Schema
201	The newly created Sponsor	⇄ sponsorsReply
401	Unauthorized	⇄ sponsorsMethodErrorDefault
500	Failed to store new Sponsor	⇄ sponsorsMethodErrorDefault

201:

```
▾sponsorsReply {
  all of:
    ▾ {
       sponsor_key: string
       event_id:    integer
       file_id:     string
       name:        string
       description: string
       email:       string
       web_page:    integer
       created_at:  string
       updated_at:  string
     }
}
```

401:

```
▾sponsorsMethodErrorDefault {
   error: string *
}
```

500:

```
▾sponsorsMethodErrorDefault {
   error: string *
}
```

Try this operation

Models

eventsMethodErrorDefault

```
▾eventsMethodErrorDefault {
   error: string *
}
```

eventsReplyWithTranslations

```
▾eventsReplyWithTranslations {
   all of:
     ▾ {
       event_key:    string
       file_id:      string
       created_by:   string
       start_date:   string
       end_date:     string
       created_at:   string
       updated_at:   string
       translations: ▾[
                        ▾eventTranslations {
                          all of:
                            ▾ {
                              language_code: string *
                              name:          string *
                              description:   string
                              footer:        string
                            }
                        }
                     ]
     }
}
```

eventsReply

```
▾eventsReply {
   all of:
     ▾ {
       event_key:  string
       file_id:    string
       created_by: string
       start_date: string
       end_date:   string
       created_at: string
       updated_at: string
     }
}
```

eventCreateUpdate

```
   ▾eventCreateUpdate {
     all of:
        ▾ {
          file_id:      string
          start_date:   string *
          end_date:     string *
          translations: ▾[
                           ▾eventTranslations {
                             all of:
⇄                               ▾ {
                                  language_code: string *
                                  name:          string *
                                  description:   string
                                  footer:        string
                                }
                           }
                         ]
        }
   }
```

eventDeleteReply

```
   ▾eventDeleteReply {
⇄    string: string
   }
```

eventTranslations

```
   ▾eventTranslations {
     all of:
        ▾ {
          language_code: string *
⇄         name:          string *
          description:   string
          footer:        string
        }
   }
```

itemsMethodErrorDefault

```
   ▾itemsMethodErrorDefault {
⇄    error: string *
   }
```

itemsReply

```
    ▾itemsReply {
      all of:
        ▾ {
          id:         integer
          item_key:   string
          sessions:   string
⇄         value:      string
          event_id:   integer
          created_at: string
          updated_at: string
        }
    }
```

itemCreate

```
    ▾itemCreate {
      all of:
        ▾ {
          event_key: string *
          sessions:  ▾[
⇄                        string
                       ]
          value:     string *
        }
    }
```

itemUpdate

```
    ▾itemUpdate {
      all of:
        ▾ {
          sessions: ▾[
⇄                       string
                    ]
          value:    string *
        }
    }
```

itemDeleteReply

```
    ▾itemDeleteReply {
⇄     string: string
    }
```

paymentsMethodErrorDefault

```
    ▾paymentsMethodErrorDefault {
⇄     error: string *
    }
```

paymentsReply

```
  ▾paymentsReply {
    all of:
      ▾ {
        payment_key:     string
        registration_id: integer
        value:           integer
⇄       paid:            integer
        payment_date:    string
        payment_method:  string
        created_at:      string
        updated_at:      string
      }
  }
```

paymentCreate

```
  ▾paymentCreate {
    all of:
      ▾ {
        registration_key: string *
        value:            integer *
⇄       paid:             integer *
        payment_date:     string *
        payment_method:   string *
      }
  }
```

paymentUpdate

```
  ▾paymentUpdate {
    all of:
      ▾ {
        registration_id: integer *
        value:           integer *
⇄       paid:            integer *
        payment_date:    string *
        payment_method:  string *
      }
  }
```

paymentDeleteReply

```
  ▾paymentDeleteReply {
⇄   string: string
  }
```

registrationsMethodErrorDefault

```
  ▾registrationsMethodErrorDefault {
⇄   error: string *
  }
```

registrationsCreateReply

```
▾registrationsCreateReply {
   all of:
     ▾ {
        registration_key: string
        user_key:         string
        event_id:         integer
        name:             string
        email:            string
        vat:              integer
        created_at:       string
        updated_at:       string
        event:            ▾ {
                            all of:
                              ▾eventsReply {
                                 all of:
                                   ▾ {
                                      event_key:  string
                                      file_id:    string
                                      created_by: string
                                      start_date: string
                                      end_date:   string
                                      created_at: string
                                      updated_at: string
                                   }
                              }
                          }
     }
}
```

registrationsShowReply

```
▾registrationsShowReply {
  all of:
    ▾ {
      id:               integer
      registration_key: string
      user_key:         string
      event_id:         integer
      name:             string
      email:            string
      vat:              integer
      created_at:       string
      updated_at:       string
      event:            ▾ {
                          all of:
                            ▾eventsReply {
                              all of:
                                ▾ {
                                  event_key:  string
                                  file_id:    string
                                  created_by: string
                                  start_date: string
                                  end_date:   string
                                  created_at: string
                                  updated_at: string
                                }
                            }
                        }
      sessions:         ▾[
                          string
                        ]
    }
}
```

registrationCreate

```
▾registrationCreate {
  all of:
    ▾ {
      event_key: string
      name:      string
      email:     string
      vat:       integer
      sessions:  ▾[
                   string
                 ]
    }
}
```

registrationDeleteReply

```
▾registrationDeleteReply {
⇄   string: string
}
```

sessionsMethodErrorDefault

```
▾sessionsMethodErrorDefault {
⇄   error: string *
}
```

sessionsReply

```
▾sessionsReply {
  all of:
    ▾ {
      session_key:  string
      event_id:     integer
      created_by:   string
      created_at:   string
      updated_at:   string
      translations: ▾[
                      ▾sessionTranslations {
                        all of:
                          ▾ {
                            session_id:    integer
                            language_code: string *
                            name:          string *
                            description:   string
                            created_at:    string
                            updated_at:    string
                          }
⇄                     }
                    ]
      schedules:    ▾[
                      ▾schedules {
                        all of:
                          ▾ {
                            schedule_key: string
                            session_id:   integer
                            start_date:   string *
                            end_date:     string *
                            start_time:   string *
                            end_time:     string *
                            created_at:   string
                            updated_at:   string
                          }
                      }
                    ]
    }
}
```

sessionTranslations

```
   ▾sessionTranslations {
     all of:
        ▾ {
          session_id:    integer
          language_code: string *
⇄         name:          string *
          description:   string
          created_at:    string
          updated_at:    string
        }
   }
```

sessionTranslationsCreate

```
   ▾sessionTranslationsCreate {
     all of:
        ▾ {
          language_code: string *
⇄         name:          string *
          description:   string *
        }
   }
```

schedules

```
   ▾schedules {
     all of:
        ▾ {
          schedule_key: string
          session_id:   integer
          start_date:   string *
⇄         end_date:     string *
          start_time:   string *
          end_time:     string *
          created_at:   string
          updated_at:   string
        }
   }
```

schedulesCreate

```
   ▾schedulesCreate {
     all of:
        ▾ {
          start_date: string *
⇄         end_date:   string *
          start_time: string *
          end_time:   string *
        }
   }
```

sessionCreate

```
▾sessionCreate {
  all of:
     ▾ {
        event_key:    string *
        translations: ▾[
                        ▾sessionTranslationsCreate {
                          all of:
                             ▾ {
                                language_code: string *
                                name:          string *
                                description:   string *
                             }
                        }
⇄                     ]
        schedules:    ▾[
                        ▾schedulesCreate {
                          all of:
                             ▾ {
                                start_date: string *
                                end_date:   string *
                                start_time: string *
                                end_time:   string *
                             }
                        }
                      ]
     }
}
```

sessionUpdate

```
▾sessionUpdate {
  all of:
    ▾ {
      translations: ▾[
                       ▾sessionTranslationsCreate {
                         all of:
                           ▾ {
                             language_code: string *
                             name:          string *
                             description:   string *
                           }
                       }
                     ]
      schedules:     ▾[
                       ▾schedulesCreate {
                         all of:
                           ▾ {
                             start_date: string *
                             end_date:   string *
                             start_time: string *
                             end_time:   string *
                           }
                       }
                     ]
    }
}
```

sessionDeleteReply

```
▾sessionDeleteReply {
  string: string
}
```

speakersMethodErrorDefault

```
▾speakersMethodErrorDefault {
  error: string *
}
```

speakersReply

```
▾speakersReply {
  all of:
    ▾ {
      speaker_key: string
      session_id:  integer
      name:        string
      company:     string
      nationality: string
      profession:  string
      age:         integer
      created_at:  string
      updated_at:  string
    }
}
```

speakerCreate

```
▾speakerCreate {
  all of:
    ▾ {
      session_key: string *
      name:        string *
      company:     string
      nationality: string
      profession:  string
      age:         integer
    }
}
```

speakerUpdate

```
▾speakerUpdate {
  all of:
    ▾ {
      name:        string *
      company:     string
      nationality: string
      profession:  string
      age:         integer
    }
}
```

speakerDeleteReply

```
▾speakerDeleteReply {
  string: string
}
```

sponsorsMethodErrorDefault

```
▾sponsorsMethodErrorDefault {
⇄   error: string *
 }
```

sponsorsReply

```
▾sponsorsReply {
  all of:
     ▾ {
        sponsor_key: string
        event_id:    integer
        file_id:     string
        name:        string
⇄       description: string
        email:       string
        web_page:    integer
        created_at:  string
        updated_at:  string
     }
 }
```

sponsorsShowReply

```
▾sponsorsShowReply {
  all of:
    ▾ {
      sponsor_key: string
      event_id:    integer
      file_id:     string
      name:        string
      description: string
      email:       string
      web_page:    integer
      created_at:  string
      updated_at:  string
      event:       ▾ {
                     all of:
                       ▾eventsReply {
                         all of:
                           ▾ {
                             event_key:  string
                             file_id:    string
                             created_by: string
                             start_date: string
                             end_date:   string
                             created_at: string
                             updated_at: string
                           }
                       }
                   }
    }
}
```

sponsorCreate

```
▾sponsorCreate {
  all of:
    ▾ {
      event_key:   string *
      name:        string *
      file_id:     string
      description: string
      email:       string
      web_page:    integer
    }
}
```

sponsorUpdate

```
▾sponsorUpdate {
  all of:
    ▾ {
        name:        string *
        file_id:     string
        description: string
        email:       string
        web_page:    integer
    }
}
```

sponsorDeleteReply

```
▾sponsorDeleteReply {
   string: string
}
```

MP Module

Version 1.0.0

Contact information
pvalente@onesource.pt (mailto:pvalente@onesource.pt)

Paths

`/mp`

`POST /mp`

MP

Summary

Create a MP

Parameters

Name	Located in	Description	Required	Schema
mp	body	MP Data	Yes	⇄
X-AUTH-TOKEN	header	User Auth Token	Yes	⇄ string
X-ENTITY-KEY	header	Entity Key	Yes	⇄ string
X-MODULE-TOKEN	header	Module Token	Yes	⇄ string
LANG-CODE	header	User Language	Yes	⇄ string
LANG-CODE-DEFAULT	header	Entity default Language	Yes	⇄ string

```
▾mpCreate {
  all of:
      ▾ {
        diagram_code: string *
        translations: ▾[
                        ▾
                        mpTranslation
                        {
                          all of:
                              ▸ { }
                        }
                      ]
        operators:    ▸ { }
      }
}
```

Responses

Code	Description	Schema

201 the newly created MP ⇄

```
▾mpReplyDefault {
  all of:
    ▾ {
        mp_key:       string
        entity_key:   string
        diagram_code: string
        finished:     boolean
        start_date:   string
        end_date:     string
        created_at:   string
        updated_at:   string
        translations: ▾[
                        ▾mpTranslationShow {
                           all of:
                             ▾ {
                                 mp_id:         integer
                                 language_code: string
                                 name:          string
                                 description:   string
                                 created_at:    string
                                 updated_at:    string
                               }
                         }
                      ]
      }
}
```

401 Unauthorized ⇄

```
▾mpErrorDefault {
   error: string
}
```

404 Mp not found ⇄

```
▾mpErrorDefault {
   error: string
}
```

500 Failed to store new Mp ⇄

```
▾mpErrorDefault {
   error: string
}
```

Try this operation

/mp/{mp_key}

`GET /mp/{mp_key}`

MP

Summary

Show a MP

Parameters

Name	Located in	Description	Required	Schema
mp_key	path	MP Key	Yes	⇄ `string`
X-MODULE-TOKEN	header	Module Token	Yes	⇄ `string`
X-ENTITY-KEY	header	Entity Token	Yes	⇄ `string`
X-AUTH-TOKEN	header	User Auth Token	Yes	⇄ `string`
LANG-CODE	header	User Language	Yes	⇄ `string`
LANG-CODE-DEFAULT	header	Entity default Language	Yes	⇄ `string`

Responses

Code	Description	Schema

Code	Description	Schema
200	Show the MP data	⇄

```
▾mpShowReply {
  all of:
    ▾ {
      mp_key:       string
      entity_key:   string
      diagram_code: string
      finished:     boolean
      start_date:   string
      end_date:     string
      created_at:   string
      updated_at:   string
      translations: ▾[
                      ▾mpTranslationShow {
                        all of:
                          ▾ {
                            mp_id:         integer
                            language_code: string
                            name:          string
                            description:   string
                            created_at:    string
                            updated_at:    string
                          }
                      }
                    ]
      operators:    ▾[
                      ▾operatorShow {
                        all of:
                          ▾ {
                            operator_key:     string
                            mp_id:            integer
                            operator_type_id: integer
                            component_key:    string
                            pad_operator_id:  integer
                            created_at:       string
                            updated_at:       string
                            operator_type:    ▸ { }
                            configurable:     boolean
                          }
                      }
                    ]
    }
}
```

Code	Description	Schema
401	Unauthorized	⇄

```
▾mpErrorDefault {
   error: string
}
```

Code	Description	Schema
404	Mp not Found	⇄

```
▾mpErrorDefault {
   error: string
}
```

Code	Description	Schema
500	Failed to retrieve mp	⇄

```
▾mpErrorDefault {
   error: string
}
```

Try this operation

`PUT /mp/{mp_key}`

MP

Summary

Update a MP

Parameters

Name	Located in	Description	Required	Schema
Body	body	MP Update Data	Yes	⇄
mp_key	path	MP Key	Yes	⇄ `string`
X-AUTH-TOKEN	header	User Auth Token	Yes	⇄ `string`
X-ENTITY-KEY	header	Entity Token	Yes	⇄ `string`
X-MODULE-TOKEN	header	Module Token	Yes	⇄ `string`
LANG-CODE	header	User Language	Yes	⇄ `string`
LANG-CODE-DEFAULT	header	Entity default Language	Yes	⇄ `string`

Body schema:

```
▾mpCreate {
   all of:
      ▾ {
         diagram_code: string *
         translations: ▾[
                           ▸
                          mpTranslation
                          { }
                        ]
         operators:     ▸ { }
       }
}
```

Responses

Code	Description	Schema

200 The updated MP ⇄

```
▾mpReplyDefault {
  all of:
    ▾ {
      mp_key:        string
      entity_key:    string
      diagram_code:  string
      finished:      boolean
      start_date:    string
      end_date:      string
      created_at:    string
      updated_at:    string
      translations: ▾[
                      ▾mpTranslationShow {
                        all of:
                          ▾ {
                            mp_id:          integer
                            language_code:  string
                            name:           string
                            description:    string
                            created_at:     string
                            updated_at:     string
                          }
                      }
                    ]
    }
}
```

401 Unauthorized ⇄

```
▾mpErrorDefault {
  error: string
}
```

404 MP not Found ⇄

```
▾mpErrorDefault {
  error: string
}
```

500 Failed to update MP ⇄

```
▾mpErrorDefault {
  error: string
}
```

Try this operation

`DELETE /mp/{mp_key}`

MP

Summary

Delete a MP

Parameters

Name	Located in	Description	Required	Schema
mp_key	path	MP Key	Yes	⇄ `integer`
X-ENTITY-KEY	header	Entity Token	Yes	⇄ `string`
X-MODULE-TOKEN	header	Module Token	Yes	⇄ `string`
X-AUTH-TOKEN	header	User Auth Token	Yes	⇄ `string`

Responses

Code	Description	Schema

200	OK	⇄ ▾ **`mpDeleteReply {`**
		`string: string`
		`}`

401	Unauthorized	⇄ ▾ **`mpErrorDefault {`**
		`error: string`
		`}`

404	MP not Found	⇄ ▾ **`mpErrorDefault {`**
		`error: string`
		`}`

500	Failed to delete MP	⇄ ▾ **`mpErrorDefault {`**
		`error: string`
		`}`

Try this operation

`/mp/{mp_key}/updateState`

`POST /mp/{mp_key}/updateState`

MP

Summary

Update the MP state

Parameters

Name	Located in	Description	Required	Schema
mpUpdateState	body	MP Finished state	Yes	⇄ ▾**mpUpdateState** `{ all of: ▸ { } }`
mp_key	path	MP Key	Yes	⇄ `string`
X-AUTH-TOKEN	header	User Auth Token	Yes	⇄ `string`
X-ENTITY-KEY	header	Entity Key	Yes	⇄ `string`
X-MODULE-TOKEN	header	Module Token	Yes	⇄ `string`
LANG-CODE	header	User Language	Yes	⇄ `string`
LANG-CODE-DEFAULT	header	Entity default Language	Yes	⇄ `string`

Responses

Code	Description	Schema

200 The Mp updated ⇄

```
▾mpReplyDefault {
  all of:
    ▾ {
      mp_key:       string
      entity_key:   string
      diagram_code: string
      finished:     boolean
      start_date:   string
      end_date:     string
      created_at:   string
      updated_at:   string
      translations: ▾[
                      ▾mpTranslationShow {
                        all of:
                          ▾ {
                            mp_id:         integer
                            language_code: string
                            name:          string
                            description:   string
                            created_at:    string
                            updated_at:    string
                          }
                      }
                    ]
    }
}
```

401 Unauthorized ⇄

```
▾mpErrorDefault {
  error: string
}
```

404 Mp not found ⇄

```
▾mpErrorDefault {
  error: string
}
```

500 Failed to update Mp finished ⇄

```
▾mpErrorDefault {
  error: string
}
```

Try this operation

`/mp/{mp_key}/updateDates`

`POST /mp/{mp_key}/updateDates`

MP

Summary

Update the MP dates

Parameters

Name	Located in	Description	Required	Schema

mpDates	body	MP dates	Yes	⇄ ▾**mpUpdateDates {**
				all of:
				▾ {
				start_date: string *
				end_date: string *
				}
				}

mp_key	path	MP Key	Yes	⇄ `string`
X-AUTH-TOKEN	header	User Auth Token	Yes	⇄ `string`
X-ENTITY-KEY	header	Entity Key	Yes	⇄ `string`
X-MODULE-TOKEN	header	Module Token	Yes	⇄ `string`
LANG-CODE	header	User Language	Yes	⇄ `string`
LANG-CODE-DEFAULT	header	Entity default Language	Yes	⇄ `string`

Responses

Code	Description	Schema

200 The Mp updated ⇄

```
▾mpReplyDefault {
  all of:
    ▾ {
      mp_key:       string
      entity_key:   string
      diagram_code: string
      finished:     boolean
      start_date:   string
      end_date:     string
      created_at:   string
      updated_at:   string
      translations: ▾[
                      ▾mpTranslationShow {
                        all of:
                          ▾ {
                            mp_id:         integer
                            language_code: string
                            name:          string
                            description:   string
                            created_at:    string
                            updated_at:    string
                          }
                      }
                    ]
    }
}
```

---

401 Unauthorized ⇄

```
▾mpErrorDefault {
   error: string
}
```

---

404 Mp not found ⇄

```
▾mpErrorDefault {
   error: string
}
```

---

500 Failed to update Mp dates ⇄

```
▾mpErrorDefault {
   error: string
}
```

Try this operation

/operator/{operator_key}

`GET /operator/{operator_key}`

Operator

Summary

Show a Operator

Parameters

Name	Located in	Description	Required	Schema
operator_key	path	Operator Key	Yes	⇄ `string`
X-MODULE-TOKEN	header	Module Token	Yes	⇄ `string`
X-ENTITY-KEY	header	Entity Token	Yes	⇄ `string`
X-AUTH-TOKEN	header	User Auth Token	Yes	⇄ `string`
LANG-CODE	header	User Language	Yes	⇄ `string`
LANG-CODE-DEFAULT	header	Entity default Language	Yes	⇄ `string`

Responses

Code	Description	Schema
200	Show the Operator data	⇄
401	Unauthorized	⇄
404	Operator not Found	⇄

200 Show the Operator data ⇄

```
▾operatorShowReply {
  all of:
    ▾ {
      operator_key:      string
      mp_id:             integer
      operator_type_id:  integer
      component_key:     string
      pad_operator_id:   integer
      created_at:        string
      updated_at:        string
      operator_type:     ▾ {
                           all of:
                             ▾operatorTypeShow
                             {
                               all of:
                                 ▸ { }
                             }
                         }
      mp:                ▾ {
                           all of:
                             ▾mpShow {
                               all of:
                                 ▸ { }
                             }
                         }
    }
}
```

401 Unauthorized ⇄

```
▾operatorErrorDefault {
  error: string
}
```

404 Operator not Found ⇄

```
▾operatorErrorDefault {
  error: string
}
```

Code	Description	Schema
500	Failed to retrieve operator	⇄

```
▾operatorErrorDefault {
   error: string
}
```

Try this operation

PUT /operator/{operator_key}

Operator

Summary

Update a Operator

Parameters

Name	Located in	Description	Required	Schema

Body	body	Operator Update Data	Yes	⇄ `▾operatorUpdate {`
				`all of:`
				`▸ { }`
				`}`

operator_key	path	Operator Key	Yes	⇄ `integer`
X-AUTH-TOKEN	header	User Auth Token	Yes	⇄ `string`
X-ENTITY-KEY	header	Entity Token	Yes	⇄ `string`
X-MODULE-TOKEN	header	Module Token	Yes	⇄ `string`
LANG-CODE	header	User Language	Yes	⇄ `string`
LANG-CODE-DEFAULT	header	Entity default Language	Yes	⇄ `string`

Responses

Code	Description	Schema

200	The updated Operator	⇄ `▾operatorReply {`
		`all of:`
		`▾ {`
		`operator_key: string`
		`mp_id: integer`
		`operator_type_id: integer`
		`component_key: string`
		`pad_operator_id: integer`
		`created_at: string`
		`updated_at: string`
		`}`
		`}`

401	Unauthorized	⇄ `▾operatorErrorDefault {`
		`error: string`
		`}`

404	Operator not Found	⇄ `▾operatorErrorDefault {`
		`error: string`
		`}`

500	Failed to update Operator	⇄ `▾operatorErrorDefault {`
		`error: string`
		`}`

Try this operation

Models

mpErrorDefault

```
  ▾mpErrorDefault {
⇄   error: string
  }
```

mpUpdateState

```
  ▾mpUpdateState {
    all of:
      ▾ {
⇄       mp_finished: boolean *
      }
  }
```

mpUpdateDates

```
  ▾mpUpdateDates {
    all of:
      ▾ {
⇄       start_date: string *
        end_date:   string *
      }
  }
```

mpTranslation

```
  ▾mpTranslation {
    all of:
      ▾ {
        language_code: string *
⇄       name:          string *
        description:   string *
      }
  }
```

mpTranslationShow

```
  ▾mpTranslationShow {
    all of:
      ▾ {
        mp_id:         integer
        language_code: string
⇄       name:          string
        description:   string
        created_at:    string
        updated_at:    string
      }
  }
```

operatorTypeShow

```
▾operatorTypeShow {
  all of:
    ▾ {
      operator_type_key: string
      code:              string
      inputs:            integer
      outputs:           integer
      created_at:        string
      updated_at:        string
      name:              string
      description:       string
    }
}
```

operator

```
▾operator {
  all of:
    ▾ {
      type:         string *
      parents:      ▾[
                      string
                    ]
      children:     ▾[
                      string
                    ]
      operator_key: string *
      pad_parent:   string
    }
}
```

operatorShow

```
▾operatorShow {
  all of:
    ▾ {
      operator_key:     string
      mp_id:            integer
      operator_type_id: integer
      component_key:    string
      pad_operator_id:  integer
      created_at:       string
      updated_at:       string
      operator_type:    ▾ {
                          all of:
                            ▾operatorTypeShow {
                              all of:
⇄                               ▾ {
                                  operator_type_key: string
                                  code:              string
                                  inputs:            integer
                                  outputs:           integer
                                  created_at:        string
                                  updated_at:        string
                                  name:              string
                                  description:       string
                                }
                            }
                        }
      configurable:     boolean
    }
}
```

mpCreate

```
▾mpCreate {
  all of:
    ▾ {
      diagram_code: string *
      translations: ▾[
                      ▾mpTranslation {
                        all of:
                          ▾ {
                            language_code: string *
                            name:          string *
                            description:   string *
                          }
                      }
                    ]
      operators:    ▾ {
⇄                     operator_js_key: ▾ {
                                         all of:
                                           ▾operator {
                                             all of:
                                               ▾ {
                                                 type:         string *
                                                 parents:      ▸[]
                                                 children:     ▸[]
                                                 operator_key: string *
                                                 pad_parent:   string
                                               }
                                           }
                                       }
                    }
    }
}
```

mpReplyDefault

```
  ▾mpReplyDefault {
    all of:
       ▾ {
         mp_key:       string
         entity_key:   string
         diagram_code: string
         finished:     boolean
         start_date:   string
         end_date:     string
         created_at:   string
         updated_at:   string
         translations: ▾[
⇄                         ▾mpTranslationShow {
                            all of:
                               ▾ {
                                 mp_id:         integer
                                 language_code: string
                                 name:          string
                                 description:   string
                                 created_at:    string
                                 updated_at:    string
                               }
                          }
                        ]
       }
  }
```

mpShowReply

```
▾mpShowReply {
  all of:
    ▾ {
      mp_key:       string
      entity_key:   string
      diagram_code: string
      finished:     boolean
      start_date:   string
      end_date:     string
      created_at:   string
      updated_at:   string
      translations: ▾[
                      ▾mpTranslationShow {
                        all of:
                          ▾ {
                            mp_id:         integer
                            language_code: string
                            name:          string
                            description:   string
                            created_at:    string
                            updated_at:    string
                          }
⇄                     }
                    ]
      operators:    ▾[
                      ▾operatorShow {
                        all of:
                          ▾ {
                            operator_key:     string
                            mp_id:            integer
                            operator_type_id: integer
                            component_key:    string
                            pad_operator_id:  integer
                            created_at:       string
                            updated_at:       string
                            operator_type:    ▾ {
                                                all of:
                                                  ▸operatorTypeShow { }
                                              }
                            configurable:     boolean
                          }
                      }
                    ]
    }
}
```

mpDeleteReply

```
  ▾mpDeleteReply {
⇄    string: string
  }
```

operatorErrorDefault

```
  ▾operatorErrorDefault {
⇄    error: string
  }
```

operatorUpdate

```
  ▾operatorUpdate {
    all of:
      ▾ {
⇄       component_key: string *
      }
  }
```

mpShow

```
  ▾mpShow {
    all of:
      ▾ {
        mp_key:       string
        entity_key:   string
        diagram_code: string
⇄       finished:     boolean
        start_date:   string
        end_date:     string
        created_at:   string
        updated_at:   string
      }
  }
```

operatorReply

```
  ▾operatorReply {
    all of:
      ▾ {
        operator_key:     string
        mp_id:            integer
        operator_type_id: integer
⇄       component_key:    string
        pad_operator_id:  integer
        created_at:       string
        updated_at:       string
      }
  }
```

operatorShowReply

```
▾operatorShowReply {
  all of:
    ▾ {
      operator_key:     string
      mp_id:            integer
      operator_type_id: integer
      component_key:    string
      pad_operator_id:  integer
      created_at:       string
      updated_at:       string
      operator_type:    ▾ {
                          all of:
                            ▾operatorTypeShow {
                              all of:
                                ▾ {
                                  operator_type_key: string
                                  code:              string
                                  inputs:            integer
                                  outputs:           integer
                                  created_at:        string
                                  updated_at:        string
                                  name:              string
                                  description:       string
                                }
                            }
                        }
      mp:               ▾ {
                          all of:
                            ▾mpShow {
                              all of:
                                ▾ {
                                  mp_key:       string
                                  entity_key:   string
                                  diagram_code: string
                                  finished:     boolean
                                  start_date:   string
                                  end_date:     string
                                  created_at:   string
                                  updated_at:   string
                                }
                            }
                        }
    }
}
```

Notify Module

Version 1.0.0

Contact information
pvalente@onesource.pt (mailto:pvalente@onesource.pt)

Paths

`/emailGroup/{email_group_key}`

`GET /emailGroup/{email_group_key}`

Email Group

Summary

Show a Email Group

Parameters

Name	Located in	Description	Required	Schema
email_group_key	path	Email Group Key	Yes	⇄ `string`
X-MODULE-TOKEN	header	Module Token	Yes	⇄ `string`

Responses

Code	Description	Schema
200	Show the Email Group data	⇄
401	Unauthorized	⇄
404	Email Group not Found	⇄
500	Failed to retrieve Email Group	⇄

200

```
▾emailGroupReply {
  all of:
    ▾ {
        id:              integer
        email_group_key: string
        site_key:        string
        created_at:      string
        updated_at:      string
        deleted_at:      string
      }
}
```

401

```
▾emailGroupErrorDefault {
  error: string
}
```

404

```
▾emailGroupErrorDefault {
  error: string
}
```

500

```
▾emailGroupErrorDefault {
  error: string
}
```

Try this operation

PUT /emailGroup/{email_group_key}

Email Group

Summary

Update an Email Group

Parameters

Name	Located in	Description	Required	Schema
Body	body	Email Group Data	Yes	⇄
email_group_key	path	Email Group Key	Yes	⇄ string
X-AUTH-TOKEN	header	User Auth Token	Yes	⇄ string
X-MODULE-TOKEN	header	Module Token	Yes	⇄ string

```
▾emailGroupCreate {
  all of:
     ▾ {
         site_key: string
                         *
       }
}
```

Responses

Code	Description	Schema
200	The updated Email Group	⇄
401	Unauthorized	⇄
404	Email Group not Found	⇄
500	Failed to update Email Group	⇄

```
▾emailGroupReply {
  all of:
     ▾ {
         id:              integer
         email_group_key: string
         site_key:        string
         created_at:      string
         updated_at:      string
         deleted_at:      string
       }
}
```

```
▾emailGroupErrorDefault {
   error: string
}
```

```
▾emailGroupErrorDefault {
   error: string
}
```

```
▾emailGroupErrorDefault {
   error: string
}
```

Try this operation

`DELETE /emailGroup/{email_group_key}`

Email Group

Summary

Delete an Email Group

Parameters

Name	Located in	Description	Required	Schema
email_group_key	path	Email Group Key	Yes	⇄ `string`
X-MODULE-TOKEN	header	Module Token	Yes	⇄ `string`
X-AUTH-TOKEN	header	User Auth Token	Yes	⇄ `string`

Responses

Code	Description	Schema

200	OK	⇄ ▾ **`emailGroupDeleteReply {`**
		`string: string`
		`}`

401	Unauthorized	⇄ ▾ **`emailGroupErrorDefault {`**
		`error: string`
		`}`

404	Email Group not Found	⇄ ▾ **`emailGroupErrorDefault {`**
		`error: string`
		`}`

500	Failed to delete Email Group	⇄ ▾ **`emailGroupErrorDefault {`**
		`error: string`
		`}`

Try this operation

`/emailGroup`

POST /emailGroup

Email Group

Summary

Create a Email Group

Parameters

Name	Located in	Description	Required	Schema
EmailGroup	body	Email Group Data	Yes	⇄
X-AUTH-TOKEN	header	User Auth Token	Yes	⇄ string
X-MODULE-TOKEN	header	Module Token	Yes	⇄ string

EmailGroup:

```
▾emailGroupCreate {
  all of:
    ▾ {
        site_key: string
                *
    }
}
```

Responses

Code	Description	Schema
201	the newly created Email Group	⇄
401	Unauthorized	⇄
404	Email Group not found	⇄
500	Failed to store Email Group	⇄

201:

```
▾emailGroupReply {
  all of:
    ▾ {
        id:              integer
        email_group_key: string
        site_key:        string
        created_at:      string
        updated_at:      string
        deleted_at:      string
    }
}
```

401:

```
▾emailGroupErrorDefault {
   error: string
}
```

404:

```
▾emailGroupErrorDefault {
   error: string
}
```

500:

```
▾emailGroupErrorDefault {
   error: string
}
```

Try this operation

/emailTemplate/{email_template_key}

GET /emailTemplate/{email_template_key}

Email Template

Summary

Show a Email Template

Parameters

Name	Located in	Description	Required	Schema
email_template_key	path	Email Template Key	Yes	⇄ string
LANG-CODE	header	User Language	Yes	⇄ string
LANG-CODE-DEFAULT	header	Entity default Language	Yes	⇄ string
X-MODULE-TOKEN	header	Module Token	Yes	⇄ string

Responses

Code **Description** **Schema**

200 Show the Email Template data ⇄

```
▾emailTemplateReply {
  all of:
    ▾ {
      id:                 integer
      email_template_key: string
      type_id:            integer
      email_group_id:     integer
      subject:            string
      header:             string
      content:            string
      footer:             string
      type:               ▾ {
                            all of:
                              ▾typeReply {
                                all of:
                                  ▾ {
                                    id:         integer
                                    type_key:   string
                                    code:       string
                                    name:       string
                                    created_at: string
                                    updated_at: string
                                    deleted_at: string
                                  }
                              }
                          }
      site_key:           string
      created_at:         string
      updated_at:         string
      deleted_at:         string
    }
}
```

401 Unauthorized ⇄

```
▾emailTemplateErrorDefault {
  error: string
}
```

Code	Description	Schema

404	Email Template not Found	⇄ ▾ **`emailTemplateErrorDefault {`**
		`error: string`
		`}`

500	Failed to retrieve Email Template	⇄ ▾ **`emailTemplateErrorDefault {`**
		`error: string`
		`}`

Try this operation

PUT /emailTemplate/{email_template_key}

Email Template

Summary

Update an Email Template

Parameters

Name	Located in	Description	Required	Schema
Body	body	Email Template Data	Yes	⇄

```
▾emailTemplateCreate {
   all of:
     ▾ {
        type_key:     string
                        *
        site_key:     string
        translations: ▸[]
     }
}
```

Name	Located in	Description	Required	Schema
email_template_key	path	Email Template Key	Yes	⇄ string
X-AUTH-TOKEN	header	User Auth Token	Yes	⇄ string
X-MODULE-TOKEN	header	Module Token	Yes	⇄ string

Responses

Code	Description	Schema
200	The updated Email Template	⇄

```
▾emailGroupReply {
   all of:
     ▾ {
        id:              integer
        email_group_key: string
        site_key:        string
        created_at:      string
        updated_at:      string
        deleted_at:      string
     }
}
```

Code	Description	Schema
401	Unauthorized	⇄

```
▾emailTemplateErrorDefault {
   error: string
}
```

Code	Description	Schema
404	Email Template not Found	⇄

```
▾emailTemplateErrorDefault {
   error: string
}
```

Code	Description	Schema
500	Failed to update Email Template	⇄

```
▾emailTemplateErrorDefault {
   error: string
}
```

Try this operation

`DELETE /emailTemplate/{email_template_key}`

Email Template

Summary

Delete an Email Template

Parameters

Name	Located in	Description	Required	Schema
email_template_key	path	Email Template Key	Yes	⇄ `string`
X-MODULE-TOKEN	header	Module Token	Yes	⇄ `string`
X-AUTH-TOKEN	header	User Auth Token	Yes	⇄ `string`

Responses

Code	Description	Schema

200	OK	⇄ ▾**emailTemplateDeleteReply** {
		string: string
		}

401	Unauthorized	⇄ ▾**emailTemplateErrorDefault** {
		error: string
		}

404	Email Template not Found	⇄ ▾**emailTemplateErrorDefault** {
		error: string
		}

500	Failed to delete Email Template	⇄ ▾**emailTemplateErrorDefault** {
		error: string
		}

Try this operation

`/emailTemplate`

`POST /emailTemplate`

Email Template

Summary

Create an Email Template

Parameters

Name	Located in	Description	Required	Schema
Body	body	Email Template Data	Yes	⇄
X-AUTH-TOKEN	header	User Auth Token	Yes	⇄ `string`
X-MODULE-TOKEN	header	Module Token	Yes	⇄ `string`

Body:

```
▾emailTemplateCreate {
  all of:
     ▾ {
        type_key:      string
                       *
        site_key:      string
        translations:  ▸[]
     }
}
```

Responses

Code	Description	Schema

Code **Description** **Schema**

201 the newly created Email Template ⇄

```
▾emailTemplateReply {
  all of:
   ▾ {
     id:                 integer
     email_template_key: string
     type_id:            integer
     email_group_id:     integer
     subject:            string
     header:             string
     content:            string
     footer:             string
     type:               ▾ {
                           all of:
                            ▾typeReply {
                              all of:
                               ▾ {
                                 id:         integer
                                 type_key:   string
                                 code:       string
                                 name:       string
                                 created_at: string
                                 updated_at: string
                                 deleted_at: string
                               }
                            }
                         }
     site_key:           string
     created_at:         string
     updated_at:         string
     deleted_at:         string
   }
}
```

401 Unauthorized ⇄

```
▾emailTemplateErrorDefault {
  error: string
}
```

404 Email Template not found ⇄

```
▾emailTemplateErrorDefault {
  error: string
}
```

500 Failed to store Email Template ⇄

```
▾emailTemplateErrorDefault {
  error: string
}
```

Try this operation

/email/send/{email_template_key}

`POST /email/send/{email_template_key}`

Email

Summary

Send an Email given a template

Parameters

Name	Located in	Description	Required	Schema
Body	body	Email Data	Yes	⇄
email_template_key	path	Email Template Key	Yes	⇄ `string`
LANG-CODE	header	User Language	Yes	⇄ `string`
LANG-CODE-DEFAULT	header	Entity default Language	Yes	⇄ `string`
X-AUTH-TOKEN	header	User Auth Token	No	⇄ `string`
X-MODULE-TOKEN	header	Module Token	Yes	⇄ `string`

```
▾emailCreate {
  all of:
    ▾ {
       user_key:    string
       recipient:   string *
       content:     ▾[
                       string
                    ]
       no_reply:    string
       sender_name: string
    }
}
```

Responses

Code	Description	Schema

200 The Email Sent ⇄

```
▾emailReply {
  all of:
    ▾ {
        id:           integer
        email_key:    string
        module_token: string
        sent:         string
        recipient:    string
        content:      string
        created_by:   string
        updated_by:   string
        created_at:   string
        updated_at:   string
        deleted_at:   string
     }
}
```

401 Unauthorized ⇄

```
▾emailErrorDefault {
   error: string
}
```

404 Email Template not Found ⇄

```
▾emailErrorDefault {
   error: string
}
```

500 Failed to send Email ⇄

```
▾emailErrorDefault {
   error: string
}
```

Try this operation

`/email/emailSend/{type_code}`

`POST /email/emailSend/{type_code}`

Email

Summary

Send an Email given a type

Parameters

Name	Located in	Description	Required	Schema
Body	body	Email Data	Yes	⇄
type_code	path	Type Code	Yes	⇄ `string`
X-SITE-KEY	header	Site Key	Yes	⇄ `string`
LANG-CODE	header	User Language	Yes	⇄ `string`
LANG-CODE-DEFAULT	header	Entity default Language	Yes	⇄ `string`
X-AUTH-TOKEN	header	User Auth Token	No	⇄ `string`
X-MODULE-TOKEN	header	Module Token	Yes	⇄ `string`

Body:

```
▾emailCreate {
   all of:
     ▾ {
         user_key:    string
         recipient:   string *
         content:     ▾[
                        string
                      ]
         no_reply:    string
         sender_name: string
       }
 }
```

Responses

Code	Description	Schema
200	The Email Sent	⇄

200:

```
▾emailReply {
   all of:
     ▾ {
         id:           integer
         email_key:    string
         module_token: string
         sent:         string
         recipient:    string
         content:      string
         created_by:   string
         updated_by:   string
         created_at:   string
         updated_at:   string
         deleted_at:   string
       }
 }
```

Code	Description	Schema

401	Unauthorized	⇄ ▾ **emailErrorDefault {**
		error: string
		}

404	Email Template not Found	⇄ ▾ **emailErrorDefault {**
		error: string
		}

500	Failed to send Email	⇄ ▾ **emailErrorDefault {**
		error: string
		}

Try this operation

/type/{type_key}

GET /type/{type_key}

Type

Summary

Show a Type

Parameters

Name	Located in	Description	Required	Schema
type_key	path	Type Key	Yes	⇄ `string`
LANG-CODE	header	User Language	Yes	⇄ `string`
LANG-CODE-DEFAULT	header	Entity default Language	Yes	⇄ `string`
X-MODULE-TOKEN	header	Module Token	Yes	⇄ `string`

Responses

Code **Description** **Schema**

200 Show the Type data ⇄

```
▾typeReply {
   all of:
     ▾ {
        id:         integer
        type_key:   string
        code:       string
        name:       string
        created_at: string
        updated_at: string
        deleted_at: string
     }
}
```

401 Unauthorized ⇄

```
▾typeErrorDefault {
   error: string
}
```

404 Type not Found ⇄

```
▾typeErrorDefault {
   error: string
}
```

500 Failed to retrieve Type ⇄

```
▾typeErrorDefault {
   error: string
}
```

Try this operation

`PUT /type/{type_key}`

Type

Summary

Update a Type

Parameters

Name	Located in	Description	Required	Schema

Body	body	Type Data	Yes	⇄ ▾typeCreate {
				all of:
				▾ {
				code: string *
				translations: ▾[
				▸ typeTranslations { }
				]
				}
				}

type_key	path	Type Key	Yes	⇄ string
X-AUTH-TOKEN	header	User Auth Token	Yes	⇄ string
X-MODULE-TOKEN	header	Module Token	Yes	⇄ string

Responses

Code	Description	Schema

200	The updated Type	⇄ ▾typeReply {
		all of:
		▾ {
		id: integer
		type_key: string
		code: string
		name: string
		created_at: string
		updated_at: string
		deleted_at: string
		}
		}

401	Unauthorized	⇄ ▾typeErrorDefault {
		error: string
		}

404	Type not Found	⇄ ▾typeErrorDefault {
		error: string
		}

500	Failed to update Type	⇄ ▾typeErrorDefault {
		error: string
		}

Try this operation

`DELETE /type/{type_key}`

Type

Summary

Delete an Type

Parameters

Name	Located in	Description	Required	Schema
type_key	path	Type Key	Yes	⇄ `string`
X-MODULE-TOKEN	header	Module Token	Yes	⇄ `string`
X-AUTH-TOKEN	header	User Auth Token	Yes	⇄ `string`

Responses

Code	Description	Schema

200	OK	⇄ ▾**typeDeleteReply** {
		string: string
		}

401	Unauthorized	⇄ ▾**typeErrorDefault** {
		error: string
		}

404	Type not Found	⇄ ▾**typeErrorDefault** {
		error: string
		}

500	Failed to delete Type	⇄ ▾**typeErrorDefault** {
		error: string
		}

Try this operation

`/type`

POST /type

Type

Summary

Create a Type

Parameters

Name	Located in	Description	Required	Schema

Body	body	Type Data	Yes	⇄ ▾typeCreate {
				all of:
				▾ {
				code: string *
				translations: ▾[
				▸ typeTranslations { }
				]
				}
				}

X-AUTH-TOKEN	header	User Auth Token	Yes	⇄ string
X-MODULE-TOKEN	header	Module Token	Yes	⇄ string

Responses

Code	Description	Schema

201	the newly created Type	⇄ ▾typeReply {
		all of:
		▾ {
		id: integer
		type_key: string
		code: string
		name: string
		created_at: string
		updated_at: string
		deleted_at: string
		}
		}

401	Unauthorized	⇄ ▾typeErrorDefault {
		error: string
		}

404	Type not found	⇄ ▾typeErrorDefault {
		error: string
		}

500	Failed to store Type	⇄ ▾typeErrorDefault {
		error: string
		}

Try this operation

Models

emailGroupErrorDefault

```
▾emailGroupErrorDefault {
   error: string
}
```

emailGroupCreate

```
▾emailGroupCreate {
  all of:
    ▾ {
      site_key: string *
    }
}
```

emailGroupReply

```
▾emailGroupReply {
  all of:
    ▾ {
      id:              integer
      email_group_key: string
      site_key:        string
      created_at:      string
      updated_at:      string
      deleted_at:      string
    }
}
```

emailGroupDeleteReply

```
▾emailGroupDeleteReply {
   string: string
}
```

emailTemplateErrorDefault

```
▾emailTemplateErrorDefault {
   error: string
}
```

emailTemplateTranslations

```
    ▾emailTemplateTranslations {
      all of:
        ▾ {
          language_code: string *
          subject:       string *
⇄         header:        string *
          content:       string *
          footer:        string *
        }
    }
```

emailTemplateCreate

```
    ▾emailTemplateCreate {
      all of:
        ▾ {
          type_key:     string *
          site_key:     string
          translations: ▾[
                          ▾emailTemplateTranslations {
                            all of:
                              ▾ {
⇄                               language_code: string *
                                subject:       string *
                                header:        string *
                                content:       string *
                                footer:        string *
                              }
                          }
                        ]
        }
    }
```

emailTemplateReply

```
  ▾emailTemplateReply {
    all of:
      ▾ {
        id:                 integer
        email_template_key: string
        type_id:            integer
        email_group_id:     integer
        subject:            string
        header:             string
        content:            string
        footer:             string
        type:               ▾ {
                              all of:
                                ▾typeReply {
                                  all of:
                                    ▾ {
⇄                                     id:         integer
                                      type_key:   string
                                      code:       string
                                      name:       string
                                      created_at: string
                                      updated_at: string
                                      deleted_at: string
                                    }
                                }
                            }
        site_key:           string
        created_at:         string
        updated_at:         string
        deleted_at:         string
      }
  }
```

emailTemplateDeleteReply

```
  ▾emailTemplateDeleteReply {
⇄   string: string
  }
```

emailErrorDefault

```
  ▾emailErrorDefault {
⇄   error: string
  }
```

emailCreate

```
▾emailCreate {
  all of:
    ▾ {
      user_key:    string
      recipient:   string *
      content:     ▾[
                     string
                   ]
      no_reply:    string
      sender_name: string
    }
}
```

emailReply

```
▾emailReply {
  all of:
    ▾ {
      id:            integer
      email_key:     string
      module_token:  string
      sent:          string
      recipient:     string
      content:       string
      created_by:    string
      updated_by:    string
      created_at:    string
      updated_at:    string
      deleted_at:    string
    }
}
```

typeErrorDefault

```
▾typeErrorDefault {
   error: string
}
```

typeTranslations

```
▾typeTranslations {
  all of:
    ▾ {
      language_code: string *
      name:          string
      subject:       string *
      header:        string *
      content:       string *
      footer:        string *
    }
}
```

typeCreate

```
▾typeCreate {
  all of:
    ▾ {
      code:         string *
      translations: ▾[
                      ▾typeTranslations {
                        all of:
                          ▾ {
                            language_code: string *
                            name:          string
                            subject:       string *
                            header:        string *
                            content:       string *
                            footer:        string *
                          }
                      }
                    ]
    }
}
```

typeReply

```
▾typeReply {
  all of:
    ▾ {
      id:         integer
      type_key:   string
      code:       string
      name:       string
      created_at: string
      updated_at: string
      deleted_at: string
    }
}
```

typeDeleteReply

```
  ▾typeDeleteReply {
⇄   string: string
  }
```

Orchestrator Module

Version 1.0.0

Contact information
pvalente@onesource.pt (mailto:pvalente@onesource.pt)

Paths

`/accessmenu/{access_menu_id}`

GET /accessmenu/{access_menu_id}

Access Menu

Summary

Show a AccessMenu

Parameters

Name	Located in	Description	Required	Schema
access_menu_id	path	Access Menu id	Yes	⇄ integer
X-MODULE-TOKEN	header	Module Token	Yes	⇄ string

Responses

Code	Description	Schema
200	Show the Access Menu data	⇄
404	AccessMenu not Found	⇄
500	Failed to retrieve AccessMenu	⇄

200 Show the Access Menu data

```
▾accessMenuShowReply {
  all of:
    ▾ {
       entity_id:   integer *
       site_id:     integer *
       name:        string *
       description: string *
       active:      boolean *
       created_by:  string *
       updated_by:  string *
       site:        ▾[
                      ▾site {
                        all of:
                          ▾ {
                             cm_key:        string
                             site_key:      string
                             entity_id:     string
                             name:          string
                             description:   string
                             link:          string
                             partial_link:  boolean
                             active:        boolean
                             no_reply_email: string
                             start_date:    string
                             end_date:      string
                          }
                      }
                    ]
    }
}
```

404 AccessMenu not Found

```
▾accessMenuErrorDefault {
  error: string *
}
```

500 Failed to retrieve AccessMenu

```
▾accessMenuErrorDefault {
  error: string *
}
```

Try this operation

`PUT /accessmenu/{access_menu_id}`

Access Menu

Summary

Update an Access Menu

Parameters

Name	Located in	Description	Required	Schema
Body	body	Access Menu Update data	Yes	⇄
access_menu_id	path	Access Menu Id	Yes	⇄ integer
X-AUTH-TOKEN	header	User Auth Token	Yes	⇄ string
X-MODULE-TOKEN	header	Module Token	Yes	⇄ string

Body schema:

```
▾accessMenuCreateReply {
  all of:
    ▾ {
      entity_id:   integer *
      site_id:     integer *
      name:        string *
      description: string *
      active:      boolean *
      created_by:  string *
      updated_by:  string *
      site_key:    string
    }
}
```

Responses

Code	Description	Schema
200	The updated Access Menu	⇄
401	Unauthorized	⇄

200 schema:

```
▾accessMenuCreateReply {
  all of:
    ▾ {
      entity_id:   integer *
      site_id:     integer *
      name:        string *
      description: string *
      active:      boolean *
      created_by:  string *
      updated_by:  string *
      site_key:    string
    }
}
```

401 schema:

```
▾accessMenuErrorDefault {
  error: string *
}
```

Code	Description	Schema

404	Access Menu not Found	⇄ `▾accessMenuErrorDefault {`
		`error: string *`
		`}`

500	Failed to update Access Menu	⇄ `▾accessMenuErrorDefault {`
		`error: string *`
		`}`

Try this operation

`DELETE /accessmenu/{access_menu_id}`

Access Menu

Summary

Delete an Access Menu

Parameters

Name	Located in	Description	Required	Schema
access_menu_id	path	Access Menu Id	Yes	⇄ `integer`
X-MODULE-TOKEN	header	Module Token	Yes	⇄ `string`
X-AUTH-TOKEN	header	User Auth Token	Yes	⇄ `string`

Responses

Code	Description	Schema

200	OK	⇄ `▾replyDeleteAccessMenu {`
		`string: string`
		`}`

401	Unauthorized	⇄ `▾accessMenuErrorDefault {`
		`error: string *`
		`}`

404	Access Menu not Found	⇄ `▾accessMenuErrorDefault {`
		`error: string *`
		`}`

500	Failed to delete Access Menu	⇄ `▾accessMenuErrorDefault {`
		`error: string *`
		`}`

Try this operation

/accessmenu

`POST /accessmenu`

Access Menu

Summary

Create an Access Menu

Parameters

Name	Located in	Description	Required	Schema
access_menu	body	Access Menu data	Yes	⇄ accessMenuCreateReply
X-AUTH-TOKEN	header	User Auth Token	Yes	⇄ string
X-MODULE-TOKEN	header	Module Token	Yes	⇄ string
X-ENTITY-KEY	header	Entity Key	Yes	⇄ string

access_menu:

```
▾accessMenuCreateReply {
  all of:
    ▾ {
        entity_id:   integer *
        site_id:     integer *
        name:        string *
        description: string *
        active:      boolean *
        created_by:  string *
        updated_by:  string *
        site_key:    string
      }
}
```

Responses

Code	Description	Schema
201	the newly created Access Menu	⇄ accessMenuCreateReply

201:

```
▾accessMenuCreateReply {
  all of:
    ▾ {
        entity_id:   integer *
        site_id:     integer *
        name:        string *
        description: string *
        active:      boolean *
        created_by:  string *
        updated_by:  string *
        site_key:    string
      }
}
```

Code **Description** **Schema**

401 Unauthorized ⇄

```
▾accessMenuErrorDefault {
   error: string *
}
```

404 Access Menu not Found ⇄

```
▾accessMenuErrorDefault {
   error: string *
}
```

500 Failed to store new Access enu ⇄

```
▾accessMenuErrorDefault {
   error: string *
}
```

Try this operation

/authmethod/{auth_method_key}

GET /authmethod/{auth_method_key}

Authentication Method

Summary

Show an Authentication Method

Parameters

Name	Located in	Description	Required	Schema
auth_method_key	path	Authentication Method Id	Yes	⇄ integer
X-AUTH-TOKEN	header	User Auth Token	Yes	⇄ string
X-MODULE-TOKEN	header	Module Token	Yes	⇄ string

Responses

Code **Description** **Schema**

200 Show the Access Menu data ⇄

```
▾authMethod {
  all of:
    ▾ {
        auth_method_key: string *
        name:            string *
        description:     string *
        code:            string *
    }
}
```

404 Authentication Method not Found ⇄

```
▾authMethodErrorDefault {
   error: string *
}
```

500 Failed to retrieve Authentication Method ⇄

```
▾authMethodErrorDefault {
   error: string *
}
```

Try this operation

`PUT /authmethod/{auth_method_key}`

Authentication Method

Summary

Update an Authentication Method

Parameters

Name	Located in	Description	Required	Schema

Body	body	Authentication Method Update Data	Yes	⇄ ▾ `authMethod {`
				`all of:`
				`▸ { }`
				`}`

auth_method_key	path	Authentication Method Key	Yes	⇄ `integer`
X-AUTH-TOKEN	header	User Auth Token	Yes	⇄ `string`
X-MODULE-TOKEN	header	Module Token	Yes	⇄ `string`

Responses

Code **Description** **Schema**

200 The updated Authentication Method ⇄

```
▾authMethod {
  all of:
    ▾ {
        auth_method_key: string *
        name:            string *
        description:     string *
        code:            string *
      }
}
```

401 Unauthorized ⇄

```
▾authMethodErrorDefault {
   error: string *
}
```

404 Authentication Method not Found ⇄

```
▾authMethodErrorDefault {
   error: string *
}
```

500 Failed to update Authentication Method ⇄

```
▾authMethodErrorDefault {
   error: string *
}
```

Try this operation

`DELETE /authmethod/{auth_method_key}`

Authentication Method

Summary

Delete an Authentication Method

Parameters

Name	Located in	Description	Required	Schema
auth_method_key	path	Authentication Method Key	Yes	⇄ `integer`
X-MODULE-TOKEN	header	Module Token	Yes	⇄ `string`
X-AUTH-TOKEN	header	User Auth Token	Yes	⇄ `string`

Responses

Code	Description	Schema

200	OK	⇄ ▾**replyDeleteAuthMethod {**
		`string: string`
		`}`

401	Unauthorized	⇄ ▾**authMethodErrorDefault {**
		`error: string *`
		`}`

404	AccessMenu not Found	⇄ ▾**authMethodErrorDefault {**
		`error: string *`
		`}`

500	Failed to delete AccessMenu	⇄ ▾**authMethodErrorDefault {**
		`error: string *`
		`}`

Try this operation

`/authmethod`

`POST /authmethod`

Authentication Method

Summary

Create an Authentication Method

Parameters

Name	Located in	Description	Required	Schema
auth_method	body	Authentication Method data	Yes	⇄
X-AUTH-TOKEN	header	User Auth Token	Yes	⇄ `string`
X-MODULE-TOKEN	header	Module Token	Yes	⇄ `string`

auth_method

```
▾authMethod {
   all of:
     ▾ {
         auth_method_key: string
                          *
         name:            string
                          *
         description:     string
                          *
         code:            string
                          *
       }
 }
```

Responses

Code	Description	Schema
201	the newly created Authentication Method	⇄
401	Unauthorized	⇄
404	Authentication Method Method not Found	⇄
500	Failed to store new Authentication Method	⇄

201

```
▾authMethod {
   all of:
     ▾ {
         auth_method_key: string *
         name:            string *
         description:     string *
         code:            string *
       }
 }
```

401

```
▾authMethodErrorDefault {
   error: string *
 }
```

404

```
▾authMethodErrorDefault {
   error: string *
 }
```

500

```
▾authMethodErrorDefault {
   error: string *
 }
```

Try this operation

`/category/{category_key}`

`GET /category/{category_key}`

Category

Summary

Show a Category

Parameters

Name	Located in	Description	Required	Schema
category_key	path	Category Key	Yes	⇄ `string`
X-MODULE-TOKEN	header	Module Token	Yes	⇄ `string`

Responses

Code	Description	Schema
200	Show the Category data	⇄

```
▾categoryReply {
  all of:
    ▾ {
        category_key: string
        entity_id:    integer *
        name:         string *
        description:  string *
        created_by:   string *
        created_at:   string
        updated_at:   string
        deleted_at:   string
    }
}
```

Code	Description	Schema
401	Unauthorized	⇄

```
▾categoryErrorDefault {
   error: string *
}
```

Code	Description	Schema
404	Category not Found	⇄

```
▾categoryErrorDefault {
   error: string *
}
```

Code	Description	Schema
500	Failed to retrieve Category	⇄

```
▾categoryErrorDefault {
   error: string *
}
```

Try this operation

PUT /category/{category_key}

Category

Summary

Update a Category

Parameters

Name	Located in	Description	Required	Schema
Body	body	Category Update Data	Yes	⇄

```
▾category {
   all of:
     ▾ {
        entity_id:   integer
                     *
        name:        string
                     *
        description: string
                     *
        created_by:  string
                     *
     }
}
```

Name	Located in	Description	Required	Schema
category_key	path	Category Key	Yes	⇄ string
X-AUTH-TOKEN	header	User Auth Token	Yes	⇄ string
X-MODULE-TOKEN	header	Module Token	Yes	⇄ string

Responses

Code	Description	Schema
200	The updated Category	⇄

```
▾categoryReply {
   all of:
     ▾ {
        category_key: string
        entity_id:    integer *
        name:         string *
        description:  string *
        created_by:   string *
        created_at:   string
        updated_at:   string
        deleted_at:   string
     }
}
```

Code	Description	Schema
401	Unauthorized	⇄

```
▾categoryErrorDefault {
   error: string *
}
```

Code	Description	Schema
404	Category not Found	⇄

```
▾categoryErrorDefault {
   error: string *
}
```

Code	Description	Schema
500	Failed to update Category	⇄

```
▾categoryErrorDefault {
   error: string *
}
```

Try this operation

DELETE /category/{category_key}

Category

Summary

Delete a Category

Parameters

Name	Located in	Description	Required	Schema
category_key	path	Category Key	Yes	⇄ string
X-MODULE-TOKEN	header	Module Token	Yes	⇄ string
X-AUTH-TOKEN	header	User Auth Token	Yes	⇄ string

Responses

Code	Description	Schema

200	OK	⇄ ▾**replyDeleteCategory** {
		string: string
		}

401	Unauthorized	⇄ ▾**categoryErrorDefault** {
		error: string *
		}

404	AccessMenu not Found	⇄ ▾**categoryErrorDefault** {
		error: string *
		}

500	Failed to delete AccessMenu	⇄ ▾**categoryErrorDefault** {
		error: string *
		}

Try this operation

/category

`POST /category`

Category

Summary

Create a Category

Parameters

Name	Located in	Description	Required	Schema
category	body	Category Data	Yes	⇄
X-AUTH-TOKEN	header	User Auth Token	Yes	⇄ string
X-ENTITY-KEY	header	Entity Key	Yes	⇄ string
X-MODULE-TOKEN	header	Module Token	Yes	⇄ string

category:

```
▾category {
  all of:
    ▾ {
       entity_id:    integer
                     *
       name:         string
                     *
       description:  string
                     *
       created_by:   string
                     *
    }
}
```

Responses

Code	Description	Schema
201	the newly created Category	⇄
401	Unauthorized	⇄
404	Category not Found	⇄

201:

```
▾categoryReply {
  all of:
    ▾ {
       category_key: string
       entity_id:    integer *
       name:         string *
       description:  string *
       created_by:   string *
       created_at:   string
       updated_at:   string
       deleted_at:   string
    }
}
```

401:

```
▾categoryErrorDefault {
  error: string *
}
```

404:

```
▾categoryErrorDefault {
  error: string *
}
```

Code	Description	Schema

500	Failed to store new Category	⇄ ▾ **`categoryErrorDefault {`**
		`error: string *`
		`}`

Try this operation

/cbParameterTemplate

POST /cbParameterTemplate

PAD Parameter Template

Summary

Create a PAD Parameter Template

Parameters

Name	Located in	Description	Required	Schema

padParameterTemplate	body	PAD Parameter Template Data	Yes	⇄ ▾ **padParameterTemplateCreate**
				{
				all of:
				▸ **{ }**
				}

X-AUTH-TOKEN	header	User Auth Token	Yes	⇄ `string`
X-ENTITY-KEY	header	Entity Key	Yes	⇄ `string`
X-MODULE-TOKEN	header	Module Token	Yes	⇄ `string`

Responses

Code	Description	Schema
201	the newly created PAD Parameter Template	⇄ ▾ **padParameterTemplateReply {** **all of:** ▾ **{** id: integer parameter_template_key: string entity_id: integer created_at: string updated_at: string deleted_at: string } }

401	Unauthorized	⇄ ▾ **padParameterTemplateErrorDefault {**
		error: string *
		}

404	Entity not found	⇄ ▾ **padParameterTemplateErrorDefault {**
		error: string *
		}

500	Failed to store new Parameter template	⇄ ▾ **padParameterTemplateErrorDefault {**
		error: string *
		}

Try this operation

/cbParameterTemplate/{parameter_template_key}

`DELETE /cbParameterTemplate/{parameter_template_key}`

PAD Parameter Template

Summary

Delete a PAD Parameter Template

Parameters

Name	Located in	Description	Required	Schema
parameter_template_key	path	PAD Parameter Template Key	Yes	⇄ `string`
X-MODULE-TOKEN	header	Module Token	Yes	⇄ `string`
X-AUTH-TOKEN	header	User Auth Token	Yes	⇄ `string`

Responses

Code	Description	Schema
200	OK	⇄
401	Unauthorized	⇄
404	Parameter template not Found	⇄
500	Failed to delete Parameter template	⇄

200 OK

```
▾replyDeletePadParameterTemplate {
  string: string
}
```

401 Unauthorized

```
▾padParameterTemplateErrorDefault {
  error: string *
}
```

404 Parameter template not Found

```
▾padParameterTemplateErrorDefault {
  error: string *
}
```

500 Failed to delete Parameter template

```
▾padParameterTemplateErrorDefault {
  error: string *
}
```

Try this operation

`/coConstruction/{co_construction_key}`

GET /coConstruction/{co_construction_key}

Co-Construction

Summary

Show a Co-Construction

Parameters

Name	Located in	Description	Required	Schema
co_construction_key	path	Co-Construction Key	Yes	⇄ string
X-MODULE-TOKEN	header	Module Token	Yes	⇄ string

Responses

Code	Description	Schema
200	Show the Co-Construction data	⇄
401	Unauthorized	⇄
404	Co-Construction not Found	⇄
500	Failed to retrieve Co-Construction	⇄

200

```
▾discussionReply {
   all of:
      ▾ {
         pad_key:    integer *
         site_id:    integer *
         start_date: string *
         end_date:   string *
         created_by: string *
         updated_by: string *
         created_at: string
         updated_at: string
         deleted_at: string
      }
}
```

401

```
▾discussionErrorDefault {
   error: string *
}
```

404

```
▾discussionErrorDefault {
   error: string *
}
```

500

```
▾discussionErrorDefault {
   error: string *
}
```

Try this operation

PUT /coConstruction/{co_construction_key}

Co-Construction

Summary

Update a Co-Construction

Parameters

Name	Located in	Description	Required	Schema
Body	body	Co-Construction Update Data	Yes	⇄
co_construction_key	path	Co-Construction Key	Yes	⇄ string
X-AUTH-TOKEN	header	User Auth Token	Yes	⇄ string
X-MODULE-TOKEN	header	Module Token	Yes	⇄ string

Body

```
▾coConstructionUpdate
{
  all of:
    ▾ {
        cb_key: string
                  *
      }
}
```

Responses

Code	Description	Schema
200	The updated Co-Construction	⇄
401	Unauthorized	⇄
404	Co-Construction not Found	⇄
500	Failed to update Co-Construction	⇄

200

```
▾coConstructionReply {
  all of:
    ▾ {
        id:                   integer
        co_construction_key:  string
        cb_key:               string
        entity_id:            integer
        created_at:           string
        updated_at:           string
        deleted_at:           string
      }
}
```

401

```
▾coConstructionErrorDefault {
   error: string *
}
```

404

```
▾coConstructionErrorDefault {
   error: string *
}
```

500

```
▾coConstructionErrorDefault {
   error: string *
}
```

Try this operation

`DELETE /coConstruction/{co_construction_key}`

Co-Construction

Summary

Delete a Co-Construction

Parameters

Name	Located in	Description	Required	Schema
co_construction_key	path	Co-Construction Key	Yes	⇄ `string`
X-MODULE-TOKEN	header	Module Token	Yes	⇄ `string`
X-AUTH-TOKEN	header	User Auth Token	Yes	⇄ `string`

Responses

Code	Description	Schema

200	OK	⇄ ▾**replyDeleteCoConstruction** {
		string: string
		}

401	Unauthorized	⇄ ▾**coConstructionErrorDefault** {
		error: string *
		}

404	Co-Construction not Found	⇄ ▾**coConstructionErrorDefault** {
		error: string *
		}

500	Failed to delete Co-Construction	⇄ ▾**coConstructionErrorDefault** {
		error: string *
		}

Try this operation

`/coConstruction`

POST /coConstruction

Co-Construction

Summary

Create a Co-Construction

Parameters

Name	Located in	Description	Required	Schema
coConstruction	body	PCo-Construction Data	Yes	⇄
X-AUTH-TOKEN	header	User Auth Token	Yes	⇄ string
X-ENTITY-KEY	header	Entity Key	Yes	⇄ string
X-MODULE-TOKEN	header	Module Token	Yes	⇄ string

coConstruction

```
▾coConstructionCreate
{
  all of:
    ▾ {
        cb_key: string *
      }
}
```

Responses

Code	Description	Schema
201	the newly created Co-Construction	⇄
401	Unauthorized	⇄
404	Entity not found	⇄
500	Failed to store new Co-Construction	⇄

201

```
▾coConstructionReply {
  all of:
    ▾ {
        id:                   integer
        co_construction_key: string
        cb_key:               string
        entity_id:            integer
        created_at:           string
        updated_at:           string
        deleted_at:           string
      }
}
```

401

```
▾coConstructionErrorDefault {
  error: string *
}
```

404

```
▾coConstructionErrorDefault {
  error: string *
}
```

500

```
▾coConstructionErrorDefault {
  error: string *
}
```

Try this operation

/country/{country_id}

`GET /country/{country_id}`

Country

Summary

List a Country

Parameters

Name	Located in	Description	Required	Schema
country_id	path	Country Id	Yes	⇄ `string`
X-AUTH-TOKEN	header	User Auth Token	Yes	⇄ `string`
X-MODULE-TOKEN	header	Module Token	Yes	⇄ `string`

Responses

Code	Description	Schema
200	Show the Country data	⇄

```
▾replyCountry {
  data: ▾ {
          all of:
             ▾Country {
                all of:
                   ▾ {
                      id:         integer *
                      name:       string *
                      code:       string *
                      created_at: string
                      updated_at: string
                      deleted_at: string
                   }
             }
        }
}
```

Code	Description	Schema
404	Country not Found	⇄

```
▾countryErrorDefault {
  error: string *
}
```

Code	Description	Schema
500	Failed to retrieve Country	⇄

```
▾countryErrorDefault {
  error: string *
}
```

Try this operation

PUT /country/{country_id}

Country

Summary

Update a Country

Parameters

Name	Located in	Description	Required	Schema

Body	body	Country Update data	Yes	⇄ ▾updateCountry {
				name: string *
				code: string *
				}

country_id	path	Level Parameter Key	Yes	⇄ string
X-AUTH-TOKEN	header	User Auth Token	Yes	⇄ string
X-MODULE-TOKEN	header	Module Token	Yes	⇄ string

Responses

Code **Description** **Schema**

200 The updated Country ⇄

```
▾replyUpdatedCountry {
  country: ▾ {
             all of:
               ▾Country {
                 all of:
                   ▾ {
                     id:         integer *
                     name:       string *
                     code:       string *
                     created_at: string
                     updated_at: string
                     deleted_at: string
                   }
               }
           }
}
```

401 Unauthorized ⇄

```
▾countryErrorDefault {
  error: string *
}
```

404 Country not Found ⇄

```
▾countryErrorDefault {
  error: string *
}
```

500 Failed to update Country ⇄

```
▾countryErrorDefault {
  error: string *
}
```

Try this operation

`DELETE /country/{country_id}`

Country

Summary

Delete a Country

Parameters

Name	Located in	Description	Required	Schema
country_id	path	Country Id	Yes	⇄ `string`
X-MODULE-TOKEN	header	Module Token	Yes	⇄ `string`
X-AUTH-TOKEN	header	Authentication Token	Yes	⇄ `string`

Responses

Code	Description	Schema

200	OK	⇄ ▾**`replyDeleteCountry {`**
		`string: string`
		`}`

401	Unauthorized	⇄ ▾**`countryErrorDefault {`**
		`error: string *`
		`}`

404	Country not Found'	⇄ ▾**`countryErrorDefault {`**
		`error: string *`
		`}`

500	Failed to delete Country	⇄ ▾**`countryErrorDefault {`**
		`error: string *`
		`}`

Try this operation

`/country`

`POST /country`

Country

Summary

Creation of a Country

Parameters

Name	Located in	Description	Required	Schema
Body	body	Country data	Yes	⇄
X-AUTH-TOKEN	header	User Auth Token	Yes	⇄ `string`
X-MODULE-TOKEN	header	Module Token	Yes	⇄ `string`

Body

```
▾createCountry {
   name: string *
   code: string *
}
```

Responses

Code	Description	Schema
201	the newly created country	⇄
401	Unauthorized	⇄
500	Failed to store new Country	⇄

201

```
▾replyCreateCountry {
   country: ▾ {
               all of:
                 ▾Country {
                    all of:
                       ▾ {
                          id:         integer *
                          name:       string *
                          code:       string *
                          created_at: string
                          updated_at: string
                          deleted_at: string
                         }
                 }
             }
}
```

401

```
▾countryErrorDefault {
   error: string *
}
```

500

```
▾countryErrorDefault {
   error: string *
}
```

Try this operation

`/currency/{currency_id}`

GET /currency/{currency_id}

Currency

Summary

Show a Currency

Parameters

Name	Located in	Description	Required	Schema
currency_id	path	Currency Id	Yes	⇄ string
X-AUTH-TOKEN	header	User Auth Token	Yes	⇄ string
X-MODULE-TOKEN	header	Module Token	Yes	⇄ string

Responses

Code	Description	Schema
200	Show the Currency data	⇄
404	Currency not Found	⇄
500	Failed to retrieve Currency	⇄

200 Show the Currency data

```
▾replyCurrency {
  data: ▾ {
           all of:
               ▾Currency {
                 all of:
                     ▾ {
                        currency:       string *
                        symbol_left:    string *
                        symbol_right:   string *
                        code:           string *
                        decimal_place:  integer *
                        decimal_point:  string *
                        thousand_point: string *
                     }
               }
         }
}
```

404 Currency not Found

```
▾currencyErrorDefault {
  error: string *
}
```

500 Failed to retrieve Currency

```
▾currencyErrorDefault {
  error: string *
}
```

Try this operation

PUT /currency/{currency_id}

Currency

Summary

Update a Currency

Parameters

Name	Located in	Description	Required	Schema

Body	body	Currency Update data	Yes	⇄ ▾**updateCurrency {**
				currency: ▾ **{**
				all of:
				▸ **Currency { }**
				}
				}

currency_id	path	Currency Key	Yes	⇄ `string`
X-AUTH-TOKEN	header	User Auth Token	Yes	⇄ `string`
X-MODULE-TOKEN	header	Module Token	Yes	⇄ `string`

Responses

Code	Description	Schema
200	The updated Currency	⇄ ▾**replyUpdatedCurrency {** currency: ▾ **{** **all of:** ▾**Currency {** **all of:** ▾ **{** currency: string * symbol_left: string * symbol_right: string * code: string * decimal_place: integer * decimal_point: string * thousand_point: string * } } } }

401	Unauthorized	⇄ ▾**currencyErrorDefault {**
		error: string *
		}

404	Currency not Found	⇄ ▾**currencyErrorDefault {**
		error: string *
		}

Code	Description	Schema

500	Failed to update Currency	⇄ ▾**currencyErrorDefault {**
		error: string *
		}

Try this operation

`DELETE /currency/{currency_id}`

Currency

Summary

Delete a Currency

Parameters

Name	Located in	Description	Required	Schema
currency_id	path	Currency Id	Yes	⇄ `string`
X-MODULE-TOKEN	header	Module Token	Yes	⇄ `string`
X-AUTH-TOKEN	header	Authentication Token	Yes	⇄ `string`

Responses

Code	Description	Schema

200	OK	⇄ ▾**replyDeleteCurrency {**
		string: string
		}

401	Unauthorized	⇄ ▾**currencyErrorDefault {**
		error: string *
		}

404	Currency not Found'	⇄ ▾**currencyErrorDefault {**
		error: string *
		}

500	Failed to delete Currency	⇄ ▾**currencyErrorDefault {**
		error: string *
		}

Try this operation

/currency

POST /currency

Currency

Summary

Creation of a Currency

Parameters

Name	Located in	Description	Required	Schema
Body	body	Currency data	Yes	⇄
X-AUTH-TOKEN	header	User Auth Token	Yes	⇄ string
X-MODULE-TOKEN	header	Module Token	Yes	⇄ string

Body

```
▾createCurrency {
   currency: ▾ {
               all of:
                  ▸Currency
                  { }
             }
}
```

Responses

Code	Description	Schema
201	the newly created currency	⇄
401	Unauthorized	⇄
500	Failed to store new Currency	⇄

201

```
▾replyCreateCurrency {
   currency: ▾ {
               all of:
                  ▾Currency {
                     all of:
                        ▾ {
                           currency:       string
                                           *
                           symbol_left:    string
                                           *
                           symbol_right:   string
                                           *
                           code:           string
                                           *
                           decimal_place:  integer
                                           *
                           decimal_point:  string
                                           *
                           thousand_point: string
                                           *
                          }
                     }
             }
}
```

401

```
▾currencyErrorDefault {
   error: string *
}
```

500

```
▾currencyErrorDefault {
   error: string *
}
```

Try this operation

/discussion/{discussion_id}

GET /discussion/{discussion_id}

Discussion

Summary

Show a Discussion

Parameters

Name	Located in	Description	Required	Schema
discussion_id	path	Discussion Id	Yes	⇄ `integer`
X-MODULE-TOKEN	header	Module Token	Yes	⇄ `string`

Responses

Code	Description	Schema
200	Show the Discussion data	⇄
401	Unauthorized	⇄
404	Discussion not Found	⇄
500	Failed to retrieve Discussion	⇄

200

```
▾discussionReply {
  all of:
    ▾ {
       pad_key:    integer *
       site_id:    integer *
       start_date: string *
       end_date:   string *
       created_by: string *
       updated_by: string *
       created_at: string
       updated_at: string
       deleted_at: string
    }
}
```

401

```
▾discussionErrorDefault {
   error: string *
}
```

404

```
▾discussionErrorDefault {
   error: string *
}
```

500

```
▾discussionErrorDefault {
   error: string *
}
```

Try this operation

PUT /discussion/{discussion_id}

Discussion

Summary

Update a Discussion

Parameters

Name	Located in	Description	Required	Schema
Body	body	Discussion Update Data	Yes	⇄
discussion_id	path	Discussion Id	Yes	⇄ string
X-AUTH-TOKEN	header	User Auth Token	Yes	⇄ string
X-MODULE-TOKEN	header	Module Token	Yes	⇄ string

Body:

```
▾discussionUpdate {
  all of:
    ▾ {
        pad_key:    integer *
        start_date: string *
        end_date:   string *
        created_by: string *
        updated_by: string *
    }
}
```

Responses

Code	Description	Schema
200	The updated Discussion	⇄
401	Unauthorized	⇄
404	Discussion not Found	⇄

200

```
▾discussionReply {
  all of:
    ▾ {
        pad_key:    integer *
        site_id:    integer *
        start_date: string *
        end_date:   string *
        created_by: string *
        updated_by: string *
        created_at: string
        updated_at: string
        deleted_at: string
    }
}
```

401

```
▾discussionErrorDefault {
  error: string *
}
```

404

```
▾discussionErrorDefault {
  error: string *
}
```

Code	Description	Schema

500	Failed to update Discussion	⇄ ▾**discussionErrorDefault {**
		error: string *
		}

Try this operation

DELETE /discussion/{discussion_id}

Discussion

Summary

Delete a Discussion

Parameters

Name	Located in	Description	Required	Schema
discussion_id	path	PAD Key	Yes	⇄ string
X-MODULE-TOKEN	header	Module Token	Yes	⇄ string
X-AUTH-TOKEN	header	User Auth Token	Yes	⇄ string

Responses

Code	Description	Schema

200	OK	⇄ ▾**replyDeleteDiscussion {**
		string: string
		}

401	Unauthorized	⇄ ▾**discussionErrorDefault {**
		error: string *
		}

404	AccessMenu not Found	⇄ ▾**discussionErrorDefault {**
		error: string *
		}

500	Failed to delete AccessMenu	⇄ ▾**discussionErrorDefault {**
		error: string *
		}

Try this operation

/discussion

POST /discussion

Discussion

Summary

Create a Discussion

Parameters

Name	Located in	Description	Required	Schema
discussion	body	Discussion Data	Yes	⇄
X-AUTH-TOKEN	header	User Auth Token	Yes	⇄ string
X-SITE-KEY	header	Site Key	Yes	⇄ string
X-MODULE-TOKEN	header	Module Token	Yes	⇄ string

```
▾discussionCreate {
   all of:
      ▾ {
         pad_key:    integer
                     *
         start_date: string
                     *
         end_date:   string
                     *
         created_by: string
                     *
        }
}
```

Responses

Code	Description	Schema
201	the newly created Discussion	⇄
401	Unauthorized	⇄
404	Discussion not Found	⇄

```
▾discussionReply {
   all of:
      ▾ {
         pad_key:    integer *
         site_id:    integer *
         start_date: string *
         end_date:   string *
         created_by: string *
         updated_by: string *
         created_at: string
         updated_at: string
         deleted_at: string
        }
}
```

```
▾discussionErrorDefault {
   error: string *
}
```

```
▾discussionErrorDefault {
   error: string *
}
```

Code	**Description**
500	Failed to store new Discussion

Schema (500)

```
▾discussionErrorDefault {
   error: string *
}
```

Try this operation

/entity/{entity_key}

GET /entity/{entity_key}

Entity

Summary

Show an Entity

Parameters

Name	Located in	Description	Required	Schema
entity_key	path	Entity Key	Yes	⇄ `string`
X-AUTH-TOKEN	header	User Auth Token	Yes	⇄ `string`
X-MODULE-TOKEN	header	Module Token	Yes	⇄ `string`

Responses

Code	**Description**
200	Show the Entity data
404	Entity not Found
500	Failed to retrieve Entity

Schema (200)

```
▾replyEntity {
   data: ▾ {
            all of:
              ▾Entity {
                 all of:
                    ▾ {
                       entity_key:  string *
                       country_id:  integer *
                       timezone_id: integer *
                       currency_id: integer *
                       name:        string *
                       designation: string *
                       description: string *
                       url:         string *
                    }
              }
         }
}
```

Schema (404)

```
▾entityErrorDefault {
   error: string *
}
```

Schema (500)

```
▾entityErrorDefault {
   error: string *
}
```

Try this operation

PUT /entity/{entity_key}

Entity

Summary

Update a Entity

Parameters

Name	Located in	Description	Required	Schema
Body	body	Entity Update data	Yes	⇄
entity_key	path	Entity Key	Yes	⇄ string
X-AUTH-TOKEN	header	User Auth Token	Yes	⇄ string
X-MODULE-TOKEN	header	Module Token	Yes	⇄ string

Body:

```
▾updateEntity {
  entity:      ▾ {
                 all of:
                   ▸Entity
                  { }
               }
  layout_key: string
}
```

Responses

Code	Description	Schema
200	The updated Entity	⇄
401	Unauthorized	⇄
404	Entity not Found	⇄
500	Failed to update Entity	⇄

200:

```
▾replyUpdatedEntity {
  entity: ▾ {
            all of:
              ▾Entity {
                all of:
                  ▾ {
                    entity_key:  string *
                    country_id:  integer *
                    timezone_id: integer *
                    currency_id: integer *
                    name:        string *
                    designation: string *
                    description: string *
                    url:         string *
                  }
              }
          }
}
```

401:

```
▾entityErrorDefault {
  error: string *
}
```

404:

```
▾entityErrorDefault {
  error: string *
}
```

500:

```
▾entityErrorDefault {
  error: string *
}
```

Try this operation

DELETE /entity/{entity_key}

Entity

Summary

Delete a Entity

Parameters

Name	Located in	Description	Required	Schema
entity_key	path	Entity Key	Yes	⇄ `string`
X-MODULE-TOKEN	header	Module Token	Yes	⇄ `string`
X-AUTH-TOKEN	header	Authentication Token	Yes	⇄ `string`

Responses

Code	Description	Schema

200	OK	⇄ ▾**replyDeleteEntity {**
		`string: string`
		`}`

401	Unauthorized	⇄ ▾**entityErrorDefault {**
		`error: string *`
		`}`

404	Entity not Found'	⇄ ▾**entityErrorDefault {**
		`error: string *`
		`}`

500	Failed to delete Entity	⇄ ▾**entityErrorDefault {**
		`error: string *`
		`}`

Try this operation

/entity

`POST /entity`

Entity

Summary

Creation of a Entity

Parameters

Name	Located in	Description	Required	Schema
Body	body	Entity data	Yes	⇄
X-AUTH-TOKEN	header	User Auth Token	Yes	⇄ `string`
X-MODULE-TOKEN	header	Module Token	Yes	⇄ `string`

Body:

```
▾createEntity {
   entity:      ▾ {
                   all of:
                      ▸Entity
                      { }
                 }
   layout_key: string
}
```

Responses

Code	Description	Schema
201	the newly created entity	⇄
401	Unauthorized	⇄
500	Failed to store new Entity	⇄

201:

```
▾replyCreateEntity {
   entity: ▾ {
              all of:
                 ▾Entity {
                    all of:
                       ▾ {
                          entity_key:  string *
                          country_id:  integer *
                          timezone_id: integer *
                          currency_id: integer *
                          name:        string *
                          designation: string *
                          description: string *
                          url:         string *
                       }
                 }
           }
}
```

401:

```
▾entityErrorDefault {
   error: string *
}
```

500:

```
▾entityErrorDefault {
   error: string *
}
```

Try this operation

/entityCb

POST /entityCb

Entity Pad

Summary

Create an Entity Pad

Parameters

Name	Located in	Description	Required	Schema
entityPad	body	Entity Pad Data	Yes	⇄
X-AUTH-TOKEN	header	User Auth Token	Yes	⇄ `string`
X-ENTITY-KEY	header	Entity Key	Yes	⇄ `string`
X-MODULE-TOKEN	header	Module Token	Yes	⇄ `string`

entityPad

```
▾entityPadCreate {
  all of:
    ▾ {
        cb_key: string *
        code:   string *
    }
}
```

Responses

Code	Description	Schema
201	the newly created Entity Pad	⇄
401	Unauthorized	⇄
404	Model not found	⇄
500	Failed to store new Entity Pad	⇄

201

```
▾entityPadReply {
  all of:
    ▾ {
      id:         integer
      cb_key:     string
      entity_id:  integer
      cb_type_id: integer
      created_at: string
      updated_at: string
      deleted_at: string
    }
}
```

401

```
▾entityPadErrorDefault {
  error: string *
}
```

404

```
▾entityPadErrorDefault {
  error: string *
}
```

500

```
▾entityPadErrorDefault {
  error: string *
}
```

Try this operation

/entityCb/{entity_pad_key}

`DELETE /entityCb/{entity_pad_key}`

Entity Pad

Summary

Delete a Entity Pad

Parameters

Name	Located in	Description	Required	Schema
entity_pad_key	path	Entity Pad Key	Yes	⇄ `string`
X-MODULE-TOKEN	header	Module Token	Yes	⇄ `string`
X-AUTH-TOKEN	header	User Auth Token	Yes	⇄ `string`

Responses

Code	Description	Schema

200	OK	⇄ **`▾entityPadDeleteReply {`**
		`string: string`
		`}`

401	Unauthorized	⇄ **`▾entityPadErrorDefault {`**
		`error: string *`
		`}`

404	Entity Pad not Found	⇄ **`▾entityPadErrorDefault {`**
		`error: string *`
		`}`

500	Failed to delete Entity Pad	⇄ **`▾entityPadErrorDefault {`**
		`error: string *`
		`}`

Try this operation

`/entityGroup/{entity_group_key}`

GET /entityGroup/{entity_group_key}

Entity Group

Summary

Show an Entity Group

Parameters

Name	Located in	Description	Required	Schema
entity_group_key	path	Entity Group Key	Yes	⇄ string
X-AUTH-TOKEN	header	User Auth Token	Yes	⇄ string
X-MODULE-TOKEN	header	Module Token	Yes	⇄ string

Responses

Code	Description	Schema
200	Show the Entity Group data	⇄

```
▾entityGroupReply {
  all of:
    ▾ {
        id:               integer
        entity_id:        integer
        entity_group_key: string
        group_type_id:    integer
        parent_group_id:  integer
        designation:      string
        name:             string
        position:         integer
        created_at:       string
        updated_at:       string
        deleted_at:       string
      }
}
```

Code	Description	Schema
401	Unauthorized	⇄

```
▾entityGroupErrorDefault {
  error: string *
}
```

Code	Description	Schema
404	Group Type not Found	⇄

```
▾entityGroupErrorDefault {
  error: string *
}
```

Code	Description	Schema
500	Failed to retrieve Group Type	⇄

```
▾entityGroupErrorDefault {
  error: string *
}
```

Try this operation

`PUT /entityGroup/{entity_group_key}`

Entity Group

Summary

Update an Entity Group

Parameters

Name	Located in	Description	Required	Schema
Body	body	Entity Group Update Data	Yes	⇄

Body:

```
▾entityGroupUpdate {
  all of:
    ▾ {
        group_type_key:   string
                          *
        entity_group_key: string
        designation:      string
                          *
        name:             string
                          *
        position:         integer
      }
}
```

Name	Located in	Description	Required	Schema
entity_group_key	path	Entity Group Key	Yes	⇄ `string`
X-AUTH-TOKEN	header	User Auth Token	Yes	⇄ `string`
X-ENTITY-KEY	header	Entity Key	Yes	⇄ `string`
X-MODULE-TOKEN	header	Module Token	Yes	⇄ `string`

Responses

Code	Description	Schema
200	The updated Entity Group	⇄

200:

```
▾entityGroupReply {
  all of:
    ▾ {
        id:               integer
        entity_id:        integer
        entity_group_key: string
        group_type_id:    integer
        parent_group_id:  integer
        designation:      string
        name:             string
        position:         integer
        created_at:       string
        updated_at:       string
        deleted_at:       string
      }
}
```

Code	Description	Schema

401	Unauthorized	⇄ ▾**entityGroupErrorDefault** {
		error: string *
		}

404	Co-Construction not Found	⇄ ▾**entityGroupErrorDefault** {
		error: string *
		}

500	Failed to update Co-Construction	⇄ ▾**entityGroupErrorDefault** {
		error: string *
		}

Try this operation

`DELETE /entityGroup/{entity_group_key}`

Entity Group

Summary

Delete a Entity Group

Parameters

Name	Located in	Description	Required	Schema
entity_group_key	path	Entity Group Key	Yes	⇄ string
X-MODULE-TOKEN	header	Module Token	Yes	⇄ string
X-ENTITY-KEY	header	Entity Key	Yes	⇄ string
X-AUTH-TOKEN	header	User Auth Token	Yes	⇄ string

Responses

Code	Description	Schema

200	OK	⇄ ▾**entityGroupDeleteReply** {
		string: string
		}

401	Unauthorized	⇄ ▾**entityGroupErrorDefault** {
		error: string *
		}

404	Entity Group not Found | Entity not Found	⇄ ▾**entityGroupErrorDefault** {
		error: string *
		}

500	Failed to delete Entity Group	⇄ ▾**entityGroupErrorDefault** {
		error: string *
		}

Try this operation

/entityGroup

`POST /entityGroup`

Entity Group

Summary

Create an Entity Group

Parameters

Name	Located in	Description	Required	Schema

entityGroup	body	Entity Group Data	Yes	⇄ ▾**entityGroupCreate {**
				all of:
				▾ **{**
				`group_type_key: string *`
				`entity_group_key: string`
				`designation: string *`
				`name: string *`
				}
				}

X-AUTH-TOKEN	header	User Auth Token	Yes	⇄ `string`
X-ENTITY-KEY	header	Entity Key	Yes	⇄ `string`
X-MODULE-TOKEN	header	Module Token	Yes	⇄ `string`

Responses

Code	Description	Schema

201	the newly created Entity Group	⇄ ▾**entityGroupReply {**
		all of:
		▾ **{**
		`id: integer`
		`entity_id: integer`
		`entity_group_key: string`
		`group_type_id: integer`
		`parent_group_id: integer`
		`designation: string`
		`name: string`
		`position: integer`
		`created_at: string`
		`updated_at: string`
		`deleted_at: string`
		}
		}

401	Unauthorized	⇄ ▾**entityGroupErrorDefault {**
		`error: string *`
		}

Code	Description	Schema

404	Model not found	⇄ `▾entityGroupErrorDefault {`
		`error: string *`
		`}`

500	Failed to store Entity Group	⇄ `▾entityGroupErrorDefault {`
		`error: string *`
		`}`

Try this operation

/entityModule

`POST /entityModule`

Entity Module

Summary

Create an Entity Module

Parameters

Name	Located in	Description	Required	Schema
entityModule	body	Entity Module Data	Yes	⇄
X-AUTH-TOKEN	header	User Auth Token	Yes	⇄ `string`
X-MODULE-TOKEN	header	Module Token	Yes	⇄ `string`

entityModule

```
▾entityModuleCreate {
  all of:
    ▾ {
        entity_key: string
                    *
        module_key: string
                    *
        modules:     ▸[]
    }
}
```

Responses

Code	Description	Schema
201	the newly created Entity Module	⇄
401	Unauthorized	⇄
404	Model not found	⇄
500	Failed to store Entity Module	⇄

201

```
▾entityModuleReply {
  all of:
    ▾ {
        id:                integer
        entity_module_key: string
        entity_id:         integer
        module_id:         integer
        created_at:        string
        updated_at:        string
        deleted_at:        string
    }
}
```

401

```
▾entityModuleErrorDefault {
   error: string *
}
```

404

```
▾entityModuleErrorDefault {
   error: string *
}
```

500

```
▾entityModuleErrorDefault {
   error: string *
}
```

Try this operation

`/forum/{forum_id}`

GET /forum/{forum_id}

Forum

Summary

Show an Entity Group

Parameters

Name	Located in	Description	Required	Schema
forum_id	path	Forum Id	Yes	⇄ string
X-MODULE-TOKEN	header	Module Token	Yes	⇄ string

Responses

Code	Description	Schema
200	Show the Forum data	⇄
401	Unauthorized	⇄
404	Forum not Found	⇄
500	Failed to retrieve Forum	⇄

200

```
▾forumReply {
  all of:
    ▾ {
       id:         integer
       cb_key:     string
       entity_id:  integer
       pb_id:      integer
       start_date: string
       end_date:   string
       created_by: integer
       updated_by: integer
       created_at: string
       updated_at: string
       deleted_at: string
     }
}
```

401

```
▾forumErrorDefault {
   error: string
}
```

404

```
▾forumErrorDefault {
   error: string
}
```

500

```
▾forumErrorDefault {
   error: string
}
```

Try this operation

`PUT /forum/{forum_id}`

Forum

Summary

Update a Forum

Parameters

Name	Located in	Description	Required	Schema
Body	body	Forum Update Data	Yes	⇄
forum_id	path	Forum Id	Yes	⇄ `integer`
X-AUTH-TOKEN	header	User Auth Token	Yes	⇄ `string`
X-ENTITY-KEY	header	Entity Key	Yes	⇄ `string`
X-MODULE-TOKEN	header	Module Token	Yes	⇄ `string`

Body:

```
▾forumCreate {
  all of:
    ▾ {
        cb_key:     string
                    *
        start_date: string
                    *
        end_date:   string
                    *
    }
}
```

Responses

Code	Description	Schema
200	The updated Forum	⇄
401	Unauthorized	⇄
404	Forum not Found	⇄

200:

```
▾forumReply {
  all of:
    ▾ {
        id:         integer
        cb_key:     string
        entity_id:  integer
        pb_id:      integer
        start_date: string
        end_date:   string
        created_by: integer
        updated_by: integer
        created_at: string
        updated_at: string
        deleted_at: string
    }
}
```

401:

```
▾forumErrorDefault {
  error: string
}
```

404:

```
▾forumErrorDefault {
  error: string
}
```

Code	Description	Schema

500	Failed to update Forum	⇄ ▾**forumErrorDefault** {
		error: string
		}

Try this operation

DELETE /forum/{forum_id}

Forum

Summary

Delete a Forum

Parameters

Name	Located in	Description	Required	Schema
forum_id	path	Forum Id	Yes	⇄ `integer`
X-MODULE-TOKEN	header	Module Token	Yes	⇄ `string`
X-ENTITY-KEY	header	Entity Key	Yes	⇄ `string`
X-AUTH-TOKEN	header	User Auth Token	Yes	⇄ `string`

Responses

Code	Description	Schema

200	OK	⇄ ▾**forumDeleteReply** {
		string: string
		}

401	Unauthorized	⇄ ▾**forumErrorDefault** {
		error: string
		}

404	Forum not Found	⇄ ▾**forumErrorDefault** {
		error: string
		}

500	Failed to delete Forum	⇄ ▾**forumErrorDefault** {
		error: string
		}

Try this operation

/forum

POST /forum

Forum

Summary

Create a Forum

Parameters

Name	Located in	Description	Required	Schema
forum	body	Forum Data	Yes	⇄
X-AUTH-TOKEN	header	User Auth Token	Yes	⇄ string
X-ENTITY-KEY	header	Entity Key	Yes	⇄ string
X-MODULE-TOKEN	header	Module Token	Yes	⇄ string

forum

```
▾forumCreate {
  all of:
    ▾ {
        cb_key:     string
                    *
        start_date: string
                    *
        end_date:   string
                    *
      }
}
```

Responses

Code	Description	Schema
201	the newly created Forum	⇄
401	Unauthorized	⇄
404	Model not found	⇄

201

```
▾forumReply {
  all of:
    ▾ {
        id:         integer
        cb_key:     string
        entity_id:  integer
        pb_id:      integer
        start_date: string
        end_date:   string
        created_by: integer
        updated_by: integer
        created_at: string
        updated_at: string
        deleted_at: string
      }
}
```

401

```
▾forumErrorDefault {
  error: string
}
```

404

```
▾forumErrorDefault {
  error: string
}
```

Code	Description	Schema
500	Failed to store Forum	⇄

500:

```
▾forumErrorDefault {
   error: string
}
```

Try this operation

/geoarea/{geographic_area_key}

GET /geoarea/{geographic_area_key}

Geographic Area

Summary

Show a Geographic Area

Parameters

Name	Located in	Description	Required	Schema
geographic_area_key	path	Geographic Area Key	Yes	⇄ `string`
X-MODULE-TOKEN	header	Module Token	Yes	⇄ `string`

Responses

Code	Description	Schema
200	Show the Geographic Area data	⇄
404	Geographic Area not Found	⇄
500	Failed to retrieve Geographic Area	⇄

200:

```
▾geographicArea {
   all of:
      ▾ {
           geo_key:    string *
           entity_id:  integer *
           name:       string *
           created_by: string *
        }
}
```

404:

```
▾geographicAreaErrorDefault {
   error: string *
}
```

500:

```
▾geographicAreaErrorDefault {
   error: string *
}
```

Try this operation

PUT /geoarea/{geographic_area_key}

Geographic Area

Summary

Update a Geographic Area

Parameters

Name	Located in	Description	Required	Schema
Body	body	Geographic Area Update data	Yes	⇄

```
▾geographicArea {
   all of:
     ▾ {
         geo_key:     string *
         entity_id:   integer *
         name:        string *
         created_by:  string *
       }
 }
```

geographic_area_key	path	Geographic Area Key	Yes	⇄ string
X-AUTH-TOKEN	header	User Auth Token	Yes	⇄ string
X-MODULE-TOKEN	header	Module Token	Yes	⇄ string

Responses

Code	Description	Schema
200	The updated Geographic Area	⇄

```
▾geographicArea {
   all of:
     ▾ {
         geo_key:    string *
         entity_id:  integer *
         name:       string *
         created_by: string *
       }
 }
```

401	Unauthorized	⇄

```
▾geographicAreaErrorDefault {
   error: string *
 }
```

404	GeographicArea not Found	⇄

```
▾geographicAreaErrorDefault {
   error: string *
 }
```

500	Failed to update GeographicArea	⇄

```
▾geographicAreaErrorDefault {
   error: string *
 }
```

Try this operation

`DELETE /geoarea/{geographic_area_key}`

Geographic Area

Summary

Delete a Geographic Area

Parameters

Name	Located in	Description	Required	Schema
geographic_area_key	path	Geographic Area Key	Yes	⇄ `string`
X-MODULE-TOKEN	header	Module Token	Yes	⇄ `string`
X-AUTH-TOKEN	header	Authentication Token	Yes	⇄ `string`

Responses

Code	Description	Schema

200	OK	⇄ ▾**`replyDeleteGeographicArea {`**
		`string: string`
		`}`

401	Unauthorized	⇄ ▾**`geographicAreaErrorDefault {`**
		`error: string *`
		`}`

404	GeographicArea not Found	⇄ ▾**`geographicAreaErrorDefault {`**
		`error: string *`
		`}`

500	Failed to delete GeographicArea	⇄ ▾**`geographicAreaErrorDefault {`**
		`error: string *`
		`}`

Try this operation

/geoarea

POST /geoarea

Geographic Area

Summary

Creation of a Geographic Area

Parameters

Name	Located in	Description	Required	Schema
Body	body	GeographicArea data	Yes	⇄
X-AUTH-TOKEN	header	User Auth Token	Yes	⇄ string
X-MODULE-TOKEN	header	Module Token	Yes	⇄ string
X-ENTITY-KEY	header	Entity Key	Yes	⇄ string

Body:

```
▾geographicArea {
   all of:
     ▾ {
         geo_key:     string
                      *
         entity_id:   integer
                      *
         name:        string
                      *
         created_by:  string
                      *
       }
}
```

Responses

Code	Description	Schema
201	the newly created geographicarea	⇄
401	Unauthorized	⇄
500	Failed to store new GeographicArea	⇄

201:

```
▾geographicArea {
   all of:
     ▾ {
         geo_key:     string *
         entity_id:   integer *
         name:        string *
         created_by:  string *
       }
}
```

401:

```
▾geographicAreaErrorDefault {
   error: string *
}
```

500:

```
▾geographicAreaErrorDefault {
   error: string *
}
```

Try this operation

/groupType/{group_type_key}

`GET /groupType/{group_type_key}`

Group Type

Summary

Show a Proposal

Parameters

Name	Located in	Description	Required	Schema
group_type_key	path	Group Type Key	Yes	⇄ `string`
X-AUTH-TOKEN	header	User Auth Token	Yes	⇄ `string`
X-MODULE-TOKEN	header	Module Token	Yes	⇄ `string`

Responses

Code	Description	Schema
200	Show the Group Type data	⇄
401	Unauthorized	⇄
404	Group Type not Found	⇄
500	Failed to retrieve Group Type	⇄

200

```
▾groupTypeReply {
  all of:
    ▾ {
        id:            integer
        groupType_key: string
        code:          string
        created_at:    string
        updated_at:    string
        deleted_at:    string
      }
}
```

401

```
▾groupTypeErrorDefault {
   error: string
}
```

404

```
▾groupTypeErrorDefault {
   error: string
}
```

500

```
▾groupTypeErrorDefault {
   error: string
}
```

Try this operation

PUT /groupType/{group_type_key}

Group Type

Summary

Update a Group Type

Parameters

Name	Located in	Description	Required	Schema
Body	body	Group Type Update Data	Yes	⇄ groupTypeCreate
group_type_key	path	Group Type Key	Yes	⇄ string
X-AUTH-TOKEN	header	User Auth Token	Yes	⇄ string
X-MODULE-TOKEN	header	Module Token	Yes	⇄ string

Body

```
▾groupTypeCreate {
  all of:
    ▾ {
        code: string
              *
      }
}
```

Responses

Code	Description	Schema
200	The updated Group Type	⇄ groupTypeReply
401	Unauthorized	⇄ groupTypeErrorDefault
404	Group Type not Found	⇄ groupTypeErrorDefault
500	Failed to update Group Type	⇄ groupTypeErrorDefault

200

```
▾groupTypeReply {
  all of:
    ▾ {
        id:             integer
        groupType_key:  string
        code:           string
        created_at:     string
        updated_at:     string
        deleted_at:     string
      }
}
```

401

```
▾groupTypeErrorDefault {
  error: string
}
```

404

```
▾groupTypeErrorDefault {
  error: string
}
```

500

```
▾groupTypeErrorDefault {
  error: string
}
```

Try this operation

`DELETE /groupType/{group_type_key}`

Group Type

Summary

Delete a Group Type

Parameters

Name	Located in	Description	Required	Schema
group_type_key	path	Group Type Key	Yes	⇄ `string`
X-MODULE-TOKEN	header	Module Token	Yes	⇄ `string`
X-AUTH-TOKEN	header	User Auth Token	Yes	⇄ `string`

Responses

Code	Description	Schema

200	OK	⇄ ▾**`groupTypeDeleteReply {`**
		`string: string`
		`}`

401	Unauthorized	⇄ ▾**`groupTypeErrorDefault {`**
		`error: string`
		`}`

404	Group Type not Found	⇄ ▾**`groupTypeErrorDefault {`**
		`error: string`
		`}`

500	Failed to delete Group Type	⇄ ▾**`groupTypeErrorDefault {`**
		`error: string`
		`}`

Try this operation

`/groupType`

`POST /groupType`

Group Type

Summary

Create a Proposal

Parameters

Name	Located in	Description	Required	Schema
GroupType	body	Group Type Data	Yes	⇄
X-AUTH-TOKEN	header	User Auth Token	Yes	⇄ string
X-MODULE-TOKEN	header	Module Token	Yes	⇄ string

GroupType

```
▾groupTypeCreate {
  all of:
     ▾ {
        code: string *
     }
}
```

Responses

Code	Description	Schema
201	the newly created Group Type	⇄
401	Unauthorized	⇄
404	Group Type not found	⇄
500	Failed to store Group Type	⇄

201

```
▾groupTypeReply {
  all of:
     ▾ {
        id:             integer
        groupType_key:  string
        code:           string
        created_at:     string
        updated_at:     string
        deleted_at:     string
     }
}
```

401

```
▾groupTypeErrorDefault {
   error: string
}
```

404

```
▾groupTypeErrorDefault {
   error: string
}
```

500

```
▾groupTypeErrorDefault {
   error: string
}
```

Try this operation

/homePageConfiguration/{home_page_configuration_key}

`GET /homePageConfiguration/{home_page_configuration_key}`

Home Page Configuration

Summary

Show a Home Page Configuration

Parameters

Name	Located in	Description	Required	Schema
home_page_configuration_key	path	Home Page Configuration Key	Yes	⇄ string
X-AUTH-TOKEN	header	User Auth Token	Yes	⇄ string
X-MODULE-TOKEN	header	Module Token	Yes	⇄ string

Responses

Code **Description** **Schema**

200 Show the Home Page Configuration data ⇄

```
▾homePageConfigurationReply {
  all of:
    ▾ {
        id:                          integer
        home_page_configuration_key: string
        home_page_type_id:           integer
        site_id:                     integer
        group_name:                  string
        group_key:                   string
        value:                       string
        created_at:                  string
        updated_at:                  string
        deleted_at:                  string
      }
}
```

401 Unauthorized ⇄

```
▾homePageConfigurationErrorDefault {
  error: string
}
```

404 Home Page Configuration not Found ⇄

```
▾homePageConfigurationErrorDefault {
  error: string
}
```

500 Failed to retrieve Home Page Configuration ⇄

```
▾homePageConfigurationErrorDefault {
  error: string
}
```

Try this operation

PUT /homePageConfiguration/{home_page_configuration_key}

Home Page Configuration

Summary

Update a Home Page Configuration

Parameters

Name	Located in	Description	Required	Schema
Body	body	Home Page Configuration Update Data	Yes	⇄
home_page_configuration_key	path	Home Page Configuration Key	Yes	⇄ string
X-AUTH-TOKEN	header	User Auth Token	Yes	⇄ string
X-MODULE-TOKEN	header	Module Token	Yes	⇄ string

Body

```
▾homePageConfigura
  all of:
    ▾ {
        value:

        home_page_t

        translation
      }
}
```

Responses

Code	Description	Schema
200	The updated Home Page Configuration	⇄
401	Unauthorized	⇄
404	Home Page Configuration not Found	⇄

200

```
▾homePageConfigurationReply {
  all of:
    ▾ {
        id:                          integer
        home_page_configuration_key: string
        home_page_type_id:           integer
        site_id:                     integer
        group_name:                  string
        group_key:                   string
        value:                       string
        created_at:                  string
        updated_at:                  string
        deleted_at:                  string
      }
}
```

401

```
▾homePageConfigurationErrorDefault {
  error: string
}
```

404

```
▾homePageConfigurationErrorDefault {
  error: string
}
```

Code	Description	Schema

500	Failed to update Home Page Configuration	⇄ ▾**homePageConfigurationErrorDefault** {
		`error: string`
		}

Try this operation

DELETE /homePageConfiguration/{home_page_configuration_key}

Home Page Configuration

Summary

Delete a Home Page Configuration

Parameters

Name	Located in	Description	Required	Schema
home_page_configuration_key	path	Home Page Configuration Key	Yes	⇄ `string`
X-MODULE-TOKEN	header	Module Token	Yes	⇄ `string`
X-AUTH-TOKEN	header	User Auth Token	Yes	⇄ `string`

Responses

Code	Description	Schema

200	OK	⇄ ▾**homePageConfigurationDeleteReply** {
		`string: string`
		}

401	Unauthorized	⇄ ▾**homePageConfigurationErrorDefault** {
		`error: string`
		}

404	Home Page Configuration not Found	⇄ ▾**homePageConfigurationErrorDefault** {
		`error: string`
		}

500	Failed to delete Home Page Configuration	⇄ ▾**homePageConfigurationErrorDefault** {
		`error: string`
		}

Try this operation

/homePageType/{home_page_type_key}

`GET /homePageType/{home_page_type_key}`

Home Page Type

Summary

Show a Home Page Type

Parameters

Name	Located in	Description	Required	Schema
home_page_type_key	path	Home Page Type Key	Yes	⇄ `string`
X-MODULE-TOKEN	header	Module Token	Yes	⇄ `string`

Responses

Code **Description** **Schema**

200 Show the Home Page Type data ⇄

```
▾homePageTypeShowReply {
  all of:
    ▾ {
      id:                 integer
      home_page_type_key: string
      entity_id:          integer
      name:               string
      code:               string
      parent_id:          integer
      type_code:          string
      parent:             ▾ {
                            all of:
                              ▸
                              homePageTypeReply
                              { }
                          }
      childs:             ▾[
                            ▸homePageTypeReply
                            { }
                          ]
      created_at:         string
      updated_at:         string
      deleted_at:         string
    }
}
```

401 Unauthorized ⇄

```
▾homePageTypeErrorDefault {
  error: string
}
```

404 Home Page Type not Found ⇄

```
▾homePageTypeErrorDefault {
  error: string
}
```

500 Failed to retrieve Home Page Type ⇄

```
▾homePageTypeErrorDefault {
  error: string
}
```

Try this operation

`PUT /homePageType/{home_page_type_key}`

Home Page Type

Summary

Update a Home Page Type

Parameters

Name	Located in	Description	Required	Schema

Body	body	Home Page Type Update Data	Yes	⇄ ▾homePageTypeCreate {
				all of:
				▾ {
				name: string *
				code: string *
				type_code: string *
				parent_key: string
				}
				}

home_page_type_key	path	Home Page Type Key	Yes	⇄ `string`
X-AUTH-TOKEN	header	User Auth Token	Yes	⇄ `string`
X-MODULE-TOKEN	header	Module Token	Yes	⇄ `string`

Responses

Code	Description	Schema

200	The updated Home Page Type	⇄ ▾homePageTypeReply {
		all of:
		▾ {
		id: integer
		home_page_type_key: string
		entity_id: integer
		name: string
		code: string
		parent_id: integer
		type_code: string
		created_at: string
		updated_at: string
		deleted_at: string
		}
		}

401	Unauthorized	⇄ ▾homePageTypeErrorDefault {
		error: string
		}

Code	Description	Schema

404	Home Page Type not Found	⇄ ▾ **`homePageTypeErrorDefault {`**
		`error: string`
		`}`

500	Failed to update Home Page Type	⇄ ▾ **`homePageTypeErrorDefault {`**
		`error: string`
		`}`

Try this operation

`DELETE /homePageType/{home_page_type_key}`

Home Page Type

Summary

Delete a Home Page Type

Parameters

Name	Located in	Description	Required	Schema
home_page_type_key	path	Home Page Type Key	Yes	⇄ `string`
X-MODULE-TOKEN	header	Module Token	Yes	⇄ `string`
X-AUTH-TOKEN	header	User Auth Token	Yes	⇄ `string`

Responses

Code	Description	Schema

200	OK	⇄ ▾ **`homePageTypeDeleteReply {`**
		`string: string`
		`}`

401	Unauthorized	⇄ ▾ **`homePageTypeErrorDefault {`**
		`error: string`
		`}`

404	Home Page Type not Found	⇄ ▾ **`homePageTypeErrorDefault {`**
		`error: string`
		`}`

500	Failed to delete Home Page Type	⇄ ▾ **`homePageTypeErrorDefault {`**
		`error: string`
		`}`

Try this operation

/homePageType

POST /homePageType

Home Page Type

Summary

Create a Home Page Type

Parameters

Name	Located in	Description	Required	Schema
HomePageType	body	Home Page Type Data	Yes	⇄

```
▾homePageTypeCreate {
  all of:
    ▾ {
        name:       string *
        code:       string *
        type_code:  string *
        parent_key: string
      }
}
```

Name	Located in	Description	Required	Schema
X-ENTITY-KEY	header	Entity Key	Yes	⇄ string
X-AUTH-TOKEN	header	User Auth Token	Yes	⇄ string
X-MODULE-TOKEN	header	Module Token	Yes	⇄ string

Responses

Code	Description	Schema
201	the newly created Home Page Type	⇄

```
▾homePageTypeReply {
  all of:
    ▾ {
        id:                 integer
        home_page_type_key: string
        entity_id:          integer
        name:               string
        code:               string
        parent_id:          integer
        type_code:          string
        created_at:         string
        updated_at:         string
        deleted_at:         string
      }
}
```

Code	Description	Schema
401	Unauthorized	⇄

```
▾homePageTypeErrorDefault {
  error: string
}
```

Code	Description	Schema
404	Home Page Type not found	⇄

```
▾homePageTypeErrorDefault {
  error: string
}
```

Code	Description	Schema
500	Failed to store Home Page Type	⇄

```
▾homePageTypeErrorDefault {
  error: string
}
```

Try this operation

/idea/{idea_id}

GET /idea/{idea_id}

Idea

Summary

Show an Entity Group

Parameters

Name	Located in	Description	Required	Schema
idea_id	path	Idea Id	Yes	⇄ integer
X-MODULE-TOKEN	header	Module Token	Yes	⇄ string

Responses

Code	Description		Schema
200	Show the Idea data	⇄	▾ **ideaReply {** **all of:** ▾ **{** id: integer cb_key: string entity_id: integer start_date: string end_date: string created_at: string updated_at: string deleted_at: string } }

401	Unauthorized	⇄	▾ **ideaErrorDefault {**
			error: string
			}

404	Idea not Found	⇄	▾ **ideaErrorDefault {**
			error: string
			}

500	Failed to retrieve Idea	⇄	▾ **ideaErrorDefault {**
			error: string
			}

Try this operation

PUT /idea/{idea_id}

Idea

Summary

Update an Idea

Parameters

Name	Located in	Description	Required	Schema
Body	body	Idea Update Data	Yes	⇄
idea_id	path	Idea Id	Yes	⇄ `integer`
X-AUTH-TOKEN	header	User Auth Token	Yes	⇄ `string`
X-ENTITY-KEY	header	Entity Key	Yes	⇄ `string`
X-MODULE-TOKEN	header	Module Token	Yes	⇄ `string`

Body

```
▾ideaCreate {
  all of:
    ▾ {
       cb_key: string
                *
      }
}
```

Responses

Code	Description	Schema
200	The updated Idea	⇄
401	Unauthorized	⇄
404	Idea not Found	⇄
500	Failed to update Idea	⇄

200

```
▾ideaReply {
  all of:
    ▾ {
       id:         integer
       cb_key:     string
       entity_id:  integer
       start_date: string
       end_date:   string
       created_at: string
       updated_at: string
       deleted_at: string
      }
}
```

401

```
▾ideaErrorDefault {
  error: string
}
```

404

```
▾ideaErrorDefault {
  error: string
}
```

500

```
▾ideaErrorDefault {
  error: string
}
```

Try this operation

`DELETE /idea/{idea_id}`

Idea

Summary

Delete a Idea

Parameters

Name	Located in	Description	Required	Schema
idea_id	path	Idea Id	Yes	⇄ `integer`
X-MODULE-TOKEN	header	Module Token	Yes	⇄ `string`
X-AUTH-TOKEN	header	User Auth Token	Yes	⇄ `string`

Responses

Code	Description	Schema

200	OK	⇄ ▾**ideaDeleteReply {**
		`string: string`
		`}`

401	Unauthorized	⇄ ▾**ideaErrorDefault {**
		`error: string`
		`}`

404	Idea not Found	⇄ ▾**ideaErrorDefault {**
		`error: string`
		`}`

500	Failed to delete Idea	⇄ ▾**ideaErrorDefault {**
		`error: string`
		`}`

Try this operation

/idea

`POST /idea`

Idea

Summary

Create an Idea

Parameters

<table>
<tr><th>Name</th><th>Located in</th><th>Description</th><th>Required</th><th>Schema</th></tr>
<tr><td>idea</td><td>body</td><td>Idea Data</td><td>Yes</td><td>⇄<pre>▾ideaCreate {
  all of:
    ▾ {
        cb_key: string *
    }
}</pre></td></tr>
<tr><td>X-AUTH-TOKEN</td><td>header</td><td>User Auth Token</td><td>Yes</td><td>⇄ string</td></tr>
<tr><td>X-ENTITY-KEY</td><td>header</td><td>Entity Key</td><td>Yes</td><td>⇄ string</td></tr>
<tr><td>X-MODULE-TOKEN</td><td>header</td><td>Module Token</td><td>Yes</td><td>⇄ string</td></tr>
</table>

Responses

<table>
<tr><th>Code</th><th>Description</th><th>Schema</th></tr>
<tr><td><b>201</b></td><td>the newly created Idea</td><td>⇄<pre>▾ideaReply {
  all of:
    ▾ {
        id: integer
        cb_key: string
        entity_id: integer
        start_date: string
        end_date: string
        created_at: string
        updated_at: string
        deleted_at: string
    }
}</pre></td></tr>
<tr><td><b>401</b></td><td>Unauthorized</td><td>⇄<pre>▾ideaErrorDefault {
   error: string
}</pre></td></tr>
<tr><td><b>404</b></td><td>Entity not found</td><td>⇄<pre>▾ideaErrorDefault {
   error: string
}</pre></td></tr>
<tr><td><b>500</b></td><td>Failed to store Idea</td><td>⇄<pre>▾ideaErrorDefault {
   error: string
}</pre></td></tr>
</table>

Try this operation

`/kiosktype/{kiosk_type_id}`

GET /kiosktype/{kiosk_type_id}

Kiosk Type

Summary

Show a Kiosk Type

Parameters

Name	Located in	Description	Required	Schema
kiosk_type_id	path	Kiosk Type Id	Yes	⇄ `integer`
X-MODULE-TOKEN	header	Module Token	Yes	⇄ `string`

Responses

Code	Description	Schema
200	Show the Kiosk Type data	⇄
401	Unauthorized	⇄
404	Kiosk Type not Found	⇄
500	Failed to retrieve Kiosk Type	⇄

200

```
▾kioskTypeReply {
  all of:
    ▾ {
        id:         integer
        title:      string
        created_at: string
        updated_at: string
        deleted_at: string
      }
}
```

401

```
▾kioskTypeErrorDefault {
  error: string
}
```

404

```
▾kioskTypeErrorDefault {
  error: string
}
```

500

```
▾kioskTypeErrorDefault {
  error: string
}
```

Try this operation

/kiosk/{kiosk_key}

`GET /kiosk/{kiosk_key}`

Kiosk

Summary

Show a Kiosk

Parameters

Name	Located in	Description	Required	Schema
kiosk_key	path	Kiosk Key	Yes	⇄ `string`
X-MODULE-TOKEN	header	Module Token	Yes	⇄ `string`

Responses

Code	Description	Schema
200	Show the Kiosk data	⇄
401	Unauthorized	⇄
404	Kiosk not Found	⇄
500	Failed to retrieve Kiosk	⇄

200

```
▾kioskReply {
  all of:
    ▾ {
      id:            integer
      kiosk_key:     string
      entity_id:     integer
      entity_cb_id:  string
      event_key:     string
      kiosk_type_id: integer
      title:         string
      created_at:    string
      updated_at:    string
      entity_cb:     ▾[
                       ▾entityPadReply {
                         all of:
                           ▾ {
                             id:          integer
                             cb_key:      string
                             entity_id:   integer
                             cb_type_id:  integer
                             created_at:  string
                             updated_at:  string
                             deleted_at:  string
                           }
                       }
                     ]
    }
}
```

401

```
▾kioskErrorDefault {
  error: string
}
```

404

```
▾kioskErrorDefault {
  error: string
}
```

500

```
▾kioskErrorDefault {
  error: string
}
```

Try this operation

`PUT /kiosk/{kiosk_key}`

Kiosk

Summary

Update an Kiosk

Parameters

Name	Located in	Description	Required	Schema
Body	body	Kiosk Update Data	Yes	⇄

```
▾kioskCreate {
  all of:
    ▾ {
      cb_key:        string
      title:         string
                     *
      kiosk_type_id: string
                     *
      event_key:     string
                     *
    }
}
```

Name	Located in	Description	Required	Schema
kiosk_key	path	Kiosk Key	Yes	⇄ `string`
X-AUTH-TOKEN	header	User Auth Token	Yes	⇄ `string`
X-ENTITY-KEY	header	Entity Key	Yes	⇄ `string`
X-MODULE-TOKEN	header	Module Token	Yes	⇄ `string`

Responses

Code	Description	Schema

200 The updated Kiosk ⇄

```
▾kioskReply {
  all of:
     ▾ {
       id:            integer
       kiosk_key:     string
       entity_id:     integer
       entity_cb_id:  string
       event_key:     string
       kiosk_type_id: integer
       title:         string
       created_at:    string
       updated_at:    string
       entity_cb:     ▾[
                         ▾entityPadReply {
                           all of:
                              ▾ {
                                id:         integer
                                cb_key:     string
                                entity_id:  integer
                                cb_type_id: integer
                                created_at: string
                                updated_at: string
                                deleted_at: string
                              }
                         }
                      ]
     }
}
```

401 Unauthorized ⇄

```
▾kioskErrorDefault {
  error: string
}
```

404 Kiosk not Found ⇄

```
▾kioskErrorDefault {
  error: string
}
```

500 Failed to update Kiosk ⇄

```
▾kioskErrorDefault {
  error: string
}
```

Try this operation

`DELETE /kiosk/{kiosk_key}`

Kiosk

Summary

Delete a Kiosk

Parameters

Name	Located in	Description	Required	Schema
kiosk_key	path	Kiosk Key	Yes	⇄ `string`
X-MODULE-TOKEN	header	Module Token	Yes	⇄ `string`
X-AUTH-TOKEN	header	User Auth Token	Yes	⇄ `string`

Responses

Code	Description	Schema

200	OK	⇄ ▾**`kioskDeleteReply {`**
		`string: string`
		`}`

401	Unauthorized	⇄ ▾**`kioskErrorDefault {`**
		`error: string`
		`}`

404	Kiosk not Found	⇄ ▾**`kioskErrorDefault {`**
		`error: string`
		`}`

500	Failed to delete Kiosk	⇄ ▾**`kioskErrorDefault {`**
		`error: string`
		`}`

Try this operation

`/kiosk`

`POST /kiosk`

Kiosk

Summary

Create an Kiosk

Parameters

Name	Located in	Description	Required	Schema
kiosk	body	Kiosk Data	Yes	⇄
X-AUTH-TOKEN	header	User Auth Token	Yes	⇄ string
X-ENTITY-KEY	header	Entity Key	Yes	⇄ string
X-MODULE-TOKEN	header	Module Token	Yes	⇄ string

kiosk:

```
▾kioskCreate {
  all of:
    ▾ {
      cb_key:        string
      title:         string
                     *
      kiosk_type_id: string
                     *
      event_key:     string
                     *
    }
}
```

Responses

Code	Description	Schema

201 the newly created Kiosk ⇄

```
▾kioskReply {
  all of:
    ▾ {
      id:           integer
      kiosk_key:    string
      entity_id:    integer
      entity_cb_id: string
      event_key:    string
      kiosk_type_id: integer
      title:        string
      created_at:   string
      updated_at:   string
      entity_cb:    ▾[
                      ▾entityPadReply {
                        all of:
                          ▾ {
                            id:         integer
                            cb_key:     string
                            entity_id:  integer
                            cb_type_id: integer
                            created_at: string
                            updated_at: string
                            deleted_at: string
                          }
                      }
                    ]
    }
}
```

401 Unauthorized ⇄

```
▾kioskErrorDefault {
  error: string
}
```

404 Entity not found ⇄

```
▾kioskErrorDefault {
  error: string
}
```

500 Failed to store Kiosk ⇄

```
▾kioskErrorDefault {
  error: string
}
```

Try this operation

/language/{language_id}

GET /language/{language_id}

Language

Summary

Show an Language

Parameters

Name	Located in	Description	Required	Schema
language_id	path	Language Key	Yes	⇄ string
X-AUTH-TOKEN	header	User Auth Token	Yes	⇄ string
X-MODULE-TOKEN	header	Module Token	Yes	⇄ string

Responses

Code	Description	Schema
200	Show the Language data	⇄
404	Language not Found	⇄
500	Failed to retrieve Language	⇄

200 Show the Language data

```
▾replyLanguage {
   data: ▾ {
            all of:
               ▾Language {
                  all of:
                     ▾ {
                        name: string *
                        code: integer *
                     }
               }
            }
}
```

404 Language not Found

```
▾languageErrorDefault {
   error: string *
}
```

500 Failed to retrieve Language

```
▾languageErrorDefault {
   error: string *
}
```

Try this operation

PUT /language/{language_id}

Language

Summary

Update a Language

Parameters

Name	Located in	Description	Required	Schema
Body	body	Language Update data	Yes	⇄
language_id	path	Language Id	Yes	⇄ string
X-AUTH-TOKEN	header	User Auth Token	Yes	⇄ string
X-MODULE-TOKEN	header	Module Token	Yes	⇄ string

Body:

```
▾updateLanguage {
   language: ▾ {
                all of:
                  ▾Language {
                    all of:
                       ▾ {
                          name: string *
                          code: integer *
                       }
                  }
              }
}
```

Responses

Code	Description	Schema
200	The updated Language	⇄
401	Unauthorized	⇄
404	Language not Found	⇄

200:

```
▾replyUpdatedLanguage {
   language: ▾ {
                all of:
                  ▾Language {
                    all of:
                       ▾ {
                          name: string *
                          code: integer *
                       }
                  }
              }
}
```

401:

```
▾languageErrorDefault {
   error: string *
}
```

404:

```
▾languageErrorDefault {
   error: string *
}
```

Code	Description	Schema

500	Failed to update Language	⇄ ▾**languageErrorDefault** {
		error: string *
		}

Try this operation

DELETE /language/{language_id}

Language

Summary

Delete a Language

Parameters

Name	Located in	Description	Required	Schema
language_id	path	Language Id	Yes	⇄ string
X-MODULE-TOKEN	header	Module Token	Yes	⇄ string
X-AUTH-TOKEN	header	Authentication Token	Yes	⇄ string

Responses

Code	Description	Schema

200	OK	⇄ ▾**replyDeleteLanguage** {
		string: string
		}

401	Unauthorized	⇄ ▾**languageErrorDefault** {
		error: string *
		}

404	Language not Found'	⇄ ▾**languageErrorDefault** {
		error: string *
		}

500	Failed to delete Language	⇄ ▾**languageErrorDefault** {
		error: string *
		}

Try this operation

/language

`POST /language`

Language

Summary

Creation of a Language

Parameters

Name	Located in	Description	Required	Schema
Body	body	Language data	Yes	⇄
X-AUTH-TOKEN	header	User Auth Token	Yes	⇄ `string`
X-MODULE-TOKEN	header	Module Token	Yes	⇄ `string`

Body:

```
▾createLanguage {
   language: ▾ {
                all of:
                    ▾Language {
                        all of:
                            ▾ {
                                name: string
                                      *
                                code: integer
                                      *
                            }
                    }
                }
}
```

Responses

Code	Description	Schema
201	the newly created language	⇄
401	Unauthorized	⇄
500	Failed to store new Language	⇄

201:

```
▾replyCreateLanguage {
   language: ▾ {
                all of:
                    ▾Language {
                        all of:
                            ▾ {
                                name: string *
                                code: integer *
                            }
                    }
                }
}
```

401:

```
▾languageErrorDefault {
   error: string *
}
```

500:

```
▾languageErrorDefault {
   error: string *
}
```

Try this operation

`/layout/{layout_key}`

`GET /layout/{layout_key}`

Layout

Summary

Show a Layout

Parameters

Name	Located in	Description	Required	Schema
layout_key	path	Layout key	Yes	⇄ `string`
X-MODULE-TOKEN	header	Module Token	Yes	⇄ `string`

Responses

Code	Description	Schema
200	Show the Layout Type data	⇄
401	Unauthorized	⇄
404	Layout not Found	⇄
500	Failed to retrieve Layout	⇄

200

```
▾layoutReply {
  all of:
     ▾ {
         id:          integer
         layout_key: string
         name:        string
         reference:   string
         created_at: string
         updated_at: string
         deleted_at: string
     }
}
```

401

```
▾layoutErrorDefault {
   error: string
}
```

404

```
▾layoutErrorDefault {
   error: string
}
```

500

```
▾layoutErrorDefault {
   error: string
}
```

Try this operation

`PUT /layout/{layout_key}`

Layout

Summary

Update a Layout

Parameters

Name	Located in	Description	Required	Schema
Body	body	Layout Data	Yes	⇄
layout_key	path	Layout Key	Yes	⇄ `string`
X-AUTH-TOKEN	header	User Auth Token	Yes	⇄ `string`
X-MODULE-TOKEN	header	Module Token	Yes	⇄ `string`

```
▾layoutCreate {
   all of:
     ▾ {
         name:      string
                    *
         reference: string
                    *
       }
}
```

Responses

Code	Description	Schema
200	The updated Layout	⇄

401	Unauthorized	⇄ ▾`layoutErrorDefault {`
		`error: string`
		`}`

404	Layout not Found	⇄ ▾`layoutErrorDefault {`
		`error: string`
		`}`

500	Failed to update Layout	⇄ ▾`layoutErrorDefault {`
		`error: string`
		`}`

```
▾layoutReply {
   all of:
     ▾ {
         id:         integer
         layout_key: string
         name:       string
         reference:  string
         created_at: string
         updated_at: string
         deleted_at: string
       }
}
```

Try this operation

`DELETE /layout/{layout_key}`

Layout

Summary

Delete a Layout

Parameters

Name	Located in	Description	Required	Schema
layout_key	path	Layout Key	Yes	⇄ `string`
X-MODULE-TOKEN	header	Module Token	Yes	⇄ `string`
X-AUTH-TOKEN	header	User Auth Token	Yes	⇄ `string`

Responses

Code	Description	Schema

200	OK	⇄ ▾**layoutDeleteReply {**
		`string: string`
		}

401	Unauthorized	⇄ ▾**layoutErrorDefault {**
		`error: string`
		}

404	Layout not Found	⇄ ▾**layoutErrorDefault {**
		`error: string`
		}

500	Failed to delete Layout	⇄ ▾**layoutErrorDefault {**
		`error: string`
		}

Try this operation

`/layout`

POST /layout

Layout

Summary

Create a Layout

Parameters

Name	Located in	Description	Required	Schema
Layout	body	Layout Data	Yes	⇄
X-AUTH-TOKEN	header	User Auth Token	Yes	⇄ string
X-MODULE-TOKEN	header	Module Token	Yes	⇄ string

```
▾layoutCreate {
   all of:
       ▾ {
           name:      string
                      *
           reference: string
                      *
         }
 }
```

Responses

Code	Description	Schema
201	the newly created Layout	⇄
401	Unauthorized	⇄
404	Layout not found	⇄
500	Failed to store Layout	⇄

```
▾layoutReply {
   all of:
       ▾ {
           id:         integer
           layout_key: string
           name:       string
           reference:  string
           created_at: string
           updated_at: string
           deleted_at: string
         }
 }
```

```
▾layoutErrorDefault {
   error: string
 }
```

```
▾layoutErrorDefault {
   error: string
 }
```

```
▾layoutErrorDefault {
   error: string
 }
```

Try this operation

/level/list

`GET /level/list`

Level Parameter

Summary

List all levels of a site

Parameters

Name	Located in	Description	Required	Schema
X-MODULE-TOKEN	header	Module Token	Yes	⇄ string
X-SITE-KEY	header	Site Key	Yes	⇄ string

Responses

Code	Description	Schema
200	List of all Site Login Levels	⇄
500	Failed to retrieve the Levels	⇄

200

```
▾replyLevelList {
   data: ▾ {
            all of:
              ▾[
                 ▾Level Parameter {
                    all of:
                      ▾ {
                         id:                   integer
                                               *
                         level_parameter_key:  string
                                               *
                         mandatory:            boolean
                                               *
                         manual_verification:  boolean
                                               *
                         name:                 string
                                               *
                         position:             integer
                                               *
                         created_at:           string
                         updated_at:           string
                         deleted_at:           string
                      }
                 }
              ]
         }
}
```

500

```
▾levelErrorDefault {
   error: string *
}
```

Try this operation

`/level/{level_parameter_key}`

GET /level/{level_parameter_key}

Level Parameter

Summary

List a Level

Parameters

Name	Located in	Description	Required	Schema
level_parameter_key	path	Level Parameter Key	Yes	⇄ string
X-MODULE-TOKEN	header	Module Token	Yes	⇄ string
X-SITE-KEY	header	Site Key	Yes	⇄ string

Responses

Code	Description	Schema
200	Show the Login Level	⇄

```
▾replyShowLevel {
   data: ▾ {
            all of:
              ▾Level Parameter {
                 all of:
                   ▾ {
                       id:                    integer
                                              *
                       level_parameter_key:  string
                                              *
                       mandatory:             boolean
                                              *
                       manual_verification:  boolean
                                              *
                       name:                  string
                                              *
                       position:              integer
                                              *
                       created_at:            string
                       updated_at:            string
                       deleted_at:            string
                   }
              }
          }
}
```

Code	Description	Schema
500	Failed to retrieve the Levels	⇄

```
▾levelErrorDefault {
   error: string *
}
```

Try this operation

PUT /level/{level_parameter_key}

Level Parameter

Summary

Update a Level

Parameters

Name	Located in	Description	Required	Schema

Body	body	Login Level Update data	Yes	⇄ ▾**updateLevel** {
				`name:` `string *`
				`mandatory:` `boolean *`
				`manual_verification:` `boolean *`
				}

level_parameter_key	path	Level Parameter Key	Yes	⇄ `string`
X-AUTH-TOKEN	header	User Auth Token	Yes	⇄ `string`
X-MODULE-TOKEN	header	Module Token	Yes	⇄ `string`

Responses

Code	Description	Schema

200	The updated level	⇄ ▾**replyUpdatedLevel** {
		`level:` ▾ {
		all of:
		▸**Level Parameter { }**
		}
		}

401	Unauthorized	⇄ ▾**levelErrorDefault** {
		`error: string *`
		}

404	Level Parameter not Found	⇄ ▾**levelErrorDefault** {
		`error: string *`
		}

500	Failed to update Level Parameter	⇄ ▾**levelErrorDefault** {
		`error: string *`
		}

Try this operation

`DELETE /level/{level_parameter_key}`

Level Parameter

Summary

Delete a Level

Parameters

Name	Located in	Description	Required	Schema
level_parameter_key	path	Level Parameter Key	Yes	⇄ `string`
X-MODULE-TOKEN	header	Module Token	Yes	⇄ `string`
X-AUTH-TOKEN	header	Authentication Token	Yes	⇄ `string`

Responses

Code	Description		Schema

200	OK	⇄	▾**`replyDeleteLevel {`**
			`string: string`
			`}`

401	Unauthorized	⇄	▾**`levelErrorDefault {`**
			`error: string *`
			`}`

404	Level Parameter not Found'	⇄	▾**`levelErrorDefault {`**
			`error: string *`
			`}`

500	'Failed to delete Level Parameter	⇄	▾**`levelErrorDefault {`**
			`error: string *`
			`}`

Try this operation

`/level`

`POST /level`

Level Parameter

Summary

Creation of a Level

Parameters

Name	Located in	Description	Required	Schema
Body	body	Login Level data	Yes	⇄
X-AUTH-TOKEN	header	User Auth Token	Yes	⇄ string
X-ENTITY-KEY	header	Entity Key	Yes	⇄ string
X-MODULE-TOKEN	header	Module Token	Yes	⇄ string
X-SITE-KEY	header	Site Key	Yes	⇄ string

Body:

```
▾createLevel {
   name:                 string
                         *
   mandatory:            boolean
                         *
   manual_verification:  boolean
                         *
}
```

Responses

Code	Description	Schema
201	the newly created level	⇄

201:

```
▾replyCreateLevel {
   level: ▾ {
            all of:
              ▾Level Parameter {
                 all of:
                   ▾ {
                      id:                   integer
                                            *
                      level_parameter_key:  string
                                            *
                      mandatory:            boolean
                                            *
                      manual_verification:  boolean
                                            *
                      name:                 string
                                            *
                      position:             integer
                                            *
                      created_at:           string
                      updated_at:           string
                      deleted_at:           string
                   }
              }
          }
}
```

Code	Description	Schema

401	Unauthorized	⇄ ▾**levelErrorDefault {**
		error: string *
		}

404	Site not Found	⇄ ▾**levelErrorDefault {**
		error: string *
		}

500	Failed to store new Level	⇄ ▾**levelErrorDefault {**
		error: string *
		}

Try this operation

/moduleType/{module_type_key}

GET /moduleType/{module_type_key}

Module Type

Summary

Show a Module Type

Parameters

Name	Located in	Description	Required	Schema
module_type_key	path	Module Type Key	Yes	⇄ string
X-MODULE-TOKEN	header	Module Token	Yes	⇄ string

Responses

Code **Description** **Schema**

200 Show the Module Type data ⇄

```
▾moduleTypeShowReply {
   all of:
     ▾ {
        id:              integer
        module_type_key: string
        module_id:       integer
        name:            string
        code:            string
        module:          ▾ {
                           all of:
                             ▾moduleReply {
                                all of:
                                  ▾ {
                                     id:         integer
                                     module_key: string
                                     name:       string
                                     code:       string
                                     token:      string
                                     created_at: string
                                     updated_at: string
                                     deleted_at: string
                                  }
                             }
                         }
        created_at:      string
        updated_at:      string
        deleted_at:      string
     }
}
```

401 Unauthorized ⇄

```
▾moduleTypeErrorDefault {
   error: string
}
```

404 Module Type not Found ⇄

```
▾moduleTypeErrorDefault {
   error: string
}
```

500 Failed to retrieve Module Type ⇄

```
▾moduleTypeErrorDefault {
   error: string
}
```

Try this operation

PUT /moduleType/{module_type_key}

Module Type

Summary

Update a Module Type

Parameters

Name	Located in	Description	Required	Schema

Body	body	Module Type Update Data	Yes	⇄ ▾**moduleTypeUpdate {**
				all of:
				▾ **{**
				name: string *
				code: string *
				}
				}

module_type_key	path	Module Type Key	Yes	⇄ string
X-AUTH-TOKEN	header	User Auth Token	Yes	⇄ string
X-MODULE-TOKEN	header	Module Token	Yes	⇄ string

Responses

Code	Description	Schema
200	The updated Module Type	⇄ ▾**moduleTypeReply {** **all of:** ▾ **{** id: integer module_type_key: string module_id: integer name: string code: string created_at: string updated_at: string deleted_at: string } }

401	Unauthorized	⇄ ▾**moduleTypeErrorDefault {**
		error: string
		}

404	Module Type not Found	⇄ ▾**moduleTypeErrorDefault {**
		error: string
		}

500	Failed to update Module Type	⇄ ▾**moduleTypeErrorDefault {**
		error: string
		}

Try this operation

`DELETE /moduleType/{module_type_key}`

Module Type

Summary

Delete a module Type

Parameters

Name	Located in	Description	Required	Schema
module_type_key	path	module Type Key	Yes	⇄ `string`
X-MODULE-TOKEN	header	Module Token	Yes	⇄ `string`
X-AUTH-TOKEN	header	User Auth Token	Yes	⇄ `string`

Responses

Code	Description	Schema

200	OK	⇄ ▾**moduleTypeDeleteReply** {
		string: string
		}

401	Unauthorized	⇄ ▾**moduleTypeErrorDefault** {
		error: string
		}

404	Module Type not Found	⇄ ▾**moduleTypeErrorDefault** {
		error: string
		}

500	Failed to delete mMdule Type	⇄ ▾**moduleTypeErrorDefault** {
		error: string
		}

Try this operation

`/moduleType`

POST /moduleType

Module Type

Summary

Create a Module Type

Parameters

Name	Located in	Description	Required	Schema
ModuleType	body	Module Type Data	Yes	⇄
X-AUTH-TOKEN	header	User Auth Token	Yes	⇄ string
X-MODULE-TOKEN	header	Module Token	Yes	⇄ string

```
▾moduleTypeCreate {
  all of:
    ▾ {
        module_key:    string *
        module_types: ▾[
                        ▸
                        moduleType
                        { }
                      ]
      }
}
```

Responses

Code	Description	Schema
201	the newly createdModule Type	⇄
401	Unauthorized	⇄
404	Module Type not found	⇄
500	Failed to store Module Type	⇄

```
▾moduleTypeReply {
  all of:
    ▾ {
        id:              integer
        module_type_key: string
        module_id:       integer
        name:            string
        code:            string
        created_at:      string
        updated_at:      string
        deleted_at:      string
      }
}
```

```
▾moduleTypeErrorDefault {
   error: string
}
```

```
▾moduleTypeErrorDefault {
   error: string
}
```

```
▾moduleTypeErrorDefault {
   error: string
}
```

Try this operation

/module/{module_key}

GET /module/{module_key}

Module

Summary

Show a Module

Parameters

Name	Located in	Description	Required	Schema
module_key	path	Module Key	Yes	⇄ string
X-MODULE-TOKEN	header	Module Token	Yes	⇄ string

Responses

Code **Description** **Schema**

200 Show the Module data ⇄

```
▾moduleReply {
  all of:
    ▾ {
      id:         integer
      module_key: string
      name:       string
      code:       string
      token:      string
      created_at: string
      updated_at: string
      deleted_at: string
    }
}
```

401 Unauthorized ⇄

```
▾moduleErrorDefault {
  error: string
}
```

404 Module not Found ⇄

```
▾moduleErrorDefault {
  error: string
}
```

500 Failed to retrieve Module ⇄

```
▾moduleErrorDefault {
  error: string
}
```

Try this operation

PUT /module/{module_key}

Module

Summary

Update an Module

Parameters

<table>
<tr><th>Name</th><th>Located in</th><th>Description</th><th>Required</th><th>Schema</th></tr>
<tr><td>Body</td><td>body</td><td>Module Update Data</td><td>Yes</td><td>⇄<pre>
▾moduleCreate {
  all of:
    ▾ {
        name: string
              *
        code: string
              *
    }
}
</pre></td></tr>
<tr><td>module_key</td><td>path</td><td>Module Key</td><td>Yes</td><td>⇄ string</td></tr>
<tr><td>X-AUTH-TOKEN</td><td>header</td><td>User Auth Token</td><td>Yes</td><td>⇄ string</td></tr>
<tr><td>X-MODULE-TOKEN</td><td>header</td><td>Module Token</td><td>Yes</td><td>⇄ string</td></tr>
</table>

Responses

<table>
<tr><th>Code</th><th>Description</th><th>Schema</th></tr>
<tr><td><b>200</b></td><td>The updated Module</td><td>⇄<pre>
▾moduleReply {
  all of:
    ▾ {
        id: integer
        module_key: string
        name: string
        code: string
        token: string
        created_at: string
        updated_at: string
        deleted_at: string
    }
}
</pre></td></tr>
<tr><td><b>401</b></td><td>Unauthorized</td><td>⇄<pre>
▾moduleErrorDefault {
  error: string
}
</pre></td></tr>
<tr><td><b>404</b></td><td>Module not Found</td><td>⇄<pre>
▾moduleErrorDefault {
  error: string
}
</pre></td></tr>
<tr><td><b>500</b></td><td>Failed to update Module</td><td>⇄<pre>
▾moduleErrorDefault {
  error: string
}
</pre></td></tr>
</table>

Try this operation

`DELETE /module/{module_key}`

Module

Summary

Delete a Module

Parameters

Name	Located in	Description	Required	Schema
module_key	path	Module Key	Yes	⇄ `string`
X-MODULE-TOKEN	header	Module Token	Yes	⇄ `string`
X-AUTH-TOKEN	header	User Auth Token	Yes	⇄ `string`

Responses

Code	Description	Schema

200	OK	⇄ ▾**`moduleDeleteReply {`**
		`string: string`
		`}`

401	Unauthorized	⇄ ▾**`moduleErrorDefault {`**
		`error: string`
		`}`

404	Module not Found	⇄ ▾**`moduleErrorDefault {`**
		`error: string`
		`}`

500	Failed to delete Module	⇄ ▾**`moduleErrorDefault {`**
		`error: string`
		`}`

Try this operation

`/module`

`POST /module`

Module

Summary

Create a Module

Parameters

Name	Located in	Description	Required	Schema
module	body	Module Data	Yes	⇄
X-AUTH-TOKEN	header	User Auth Token	Yes	⇄ `string`
X-MODULE-TOKEN	header	Module Token	Yes	⇄ `string`

module

```
▾moduleCreate {
  all of:
    ▾ {
        name: string *
        code: string *
      }
  }
```

Responses

Code	Description	Schema
201	the newly created Module	⇄
401	Unauthorized	⇄
500	Failed to store Module	⇄

201

```
▾moduleReply {
  all of:
    ▾ {
        id:         integer
        module_key: string
        name:       string
        code:       string
        token:      string
        created_at: string
        updated_at: string
        deleted_at: string
      }
  }
```

401

```
▾moduleErrorDefault {
  error: string
}
```

500

```
▾moduleErrorDefault {
  error: string
}
```

Try this operation

`/page/{page_key}`

`GET /page/{page_key}`

Page

Summary

Show a Page

Parameters

Name	Located in	Description	Required	Schema
page_key	path	Page Key	Yes	⇄ `string`
X-MODULE-TOKEN	header	Module Token	Yes	⇄ `string`

Responses

Code	Description	Schema
200	Show the Page data	⇄
401	Unauthorized	⇄
404	Page not Found	⇄
500	Failed to retrieve Page	⇄

200

```
▾pageReply {
  all of:
    ▾ {
       id:         integer
       page_key:   string
       entity_id:  integer
       type:       string
       created_at: string
       updated_at: string
       deleted_at: string
     }
}
```

401

```
▾pageErrorDefault {
   error: string
}
```

404

```
▾pageErrorDefault {
   error: string
}
```

500

```
▾pageErrorDefault {
   error: string
}
```

Try this operation

PUT /page/{page_key}

Page

Summary

Update an Page

Parameters

<table>
<tr><th>Name</th><th>Located in</th><th>Description</th><th>Required</th><th>Schema</th></tr>
<tr><td>Body</td><td>body</td><td>Page Update Data</td><td>Yes</td><td>⇄<pre>
▾pageUpdate {
  all of:
    ▾ {
        entity_id: integer
                   *
        type: string
                   *
    }
}
</pre></td></tr>
<tr><td>page_key</td><td>path</td><td>Page Key</td><td>Yes</td><td>⇄ <code>string</code></td></tr>
<tr><td>X-AUTH-TOKEN</td><td>header</td><td>User Auth Token</td><td>Yes</td><td>⇄ <code>string</code></td></tr>
<tr><td>X-MODULE-TOKEN</td><td>header</td><td>Module Token</td><td>Yes</td><td>⇄ <code>string</code></td></tr>
</table>

Responses

<table>
<tr><th>Code</th><th>Description</th><th>Schema</th></tr>
<tr><td><b>200</b></td><td>The updated Page</td><td>⇄<pre>
▾pageReply {
  all of:
    ▾ {
        id: integer
        page_key: string
        entity_id: integer
        type: string
        created_at: string
        updated_at: string
        deleted_at: string
    }
}
</pre></td></tr>
<tr><td><b>401</b></td><td>Unauthorized</td><td>⇄<pre>
▾pageErrorDefault {
   error: string
}
</pre></td></tr>
<tr><td><b>404</b></td><td>Page not Found</td><td>⇄<pre>
▾pageErrorDefault {
   error: string
}
</pre></td></tr>
<tr><td><b>500</b></td><td>Failed to update Page</td><td>⇄<pre>
▾pageErrorDefault {
   error: string
}
</pre></td></tr>
</table>

Try this operation

`DELETE /page/{page_key}`

Page

Summary

Delete a Page

Parameters

Name	Located in	Description	Required	Schema
page_key	path	Page Key	Yes	⇄ `string`
X-MODULE-TOKEN	header	Module Token	Yes	⇄ `string`
X-AUTH-TOKEN	header	User Auth Token	Yes	⇄ `string`

Responses

Code	Description	Schema

200	OK	⇄ ▾**pageDeleteReply {**
		`string: string`
		}

401	Unauthorized	⇄ ▾**pageErrorDefault {**
		`error: string`
		}

404	Page not Found	⇄ ▾**pageErrorDefault {**
		`error: string`
		}

500	Failed to delete Page	⇄ ▾**pageErrorDefault {**
		`error: string`
		}

Try this operation

/page

POST /page

Page

Summary

Create a Page

Parameters

<table>
<tr><th>Name</th><th>Located in</th><th>Description</th><th>Required</th><th>Schema</th></tr>
<tr><td>page</td><td>body</td><td>Page Data</td><td>Yes</td><td>⇄ <pre>▾pageCreate {
  all of:
    ▾ {
        page_key: string *
        type: string *
      }
}</pre></td></tr>
<tr><td>X-ENTITY-KEY</td><td>header</td><td>Entity Key</td><td>Yes</td><td>⇄ <code>string</code></td></tr>
<tr><td>X-AUTH-TOKEN</td><td>header</td><td>User Auth Token</td><td>Yes</td><td>⇄ <code>string</code></td></tr>
<tr><td>X-MODULE-TOKEN</td><td>header</td><td>Module Token</td><td>Yes</td><td>⇄ <code>string</code></td></tr>
</table>

Responses

<table>
<tr><th>Code</th><th>Description</th><th>Schema</th></tr>
<tr><td><b>201</b></td><td>the newly created Page</td><td>⇄ <pre>▾pageReply {
  all of:
    ▾ {
        id: integer
        page_key: string
        entity_id: integer
        type: string
        created_at: string
        updated_at: string
        deleted_at: string
      }
}</pre></td></tr>
<tr><td><b>401</b></td><td>Unauthorized</td><td>⇄ <pre>▾pageErrorDefault {
  error: string
}</pre></td></tr>
<tr><td><b>404</b></td><td>Entity not found</td><td>⇄ <pre>▾pageErrorDefault {
  error: string
}</pre></td></tr>
<tr><td><b>500</b></td><td>Failed to store Page</td><td>⇄ <pre>▾pageErrorDefault {
  error: string
}</pre></td></tr>
</table>

Try this operation

/parameterTypes/{parameter_type_id}

GET /parameterTypes/{parameter_type_id}

Parameter Type

Summary

Show a Parameter Type

Parameters

Name	Located in	Description	Required	Schema
parameter_type_id	path	Parameter Type Id	Yes	⇄ `integer`
X-MODULE-TOKEN	header	Module Token	Yes	⇄ `string`

Responses

Code	Description	Schema
200	Show the Parameter Type data	⇄

```
▾parameterTypeReply {
  all of:
    ▾ {
      id:             integer
      name:           string
      code:           string
      user_parameter: boolean
      created_at:     string
      updated_at:     string
      deleted_at:     string
    }
}
```

Code	Description	Schema
401	Unauthorized	⇄

```
▾parameterTypeErrorDefault {
  error: string
}
```

Code	Description	Schema
404	Parameter Type not Found	⇄

```
▾parameterTypeErrorDefault {
  error: string
}
```

Code	Description	Schema
500	Failed to retrieve Parameter Type	⇄

```
▾parameterTypeErrorDefault {
  error: string
}
```

Try this operation

PUT /parameterTypes/{parameter_type_id}

Parameter Type

Summary

Update a Parameter Type

Parameters

Name	Located in	Description	Required	Schema

Body	body	Parameter Type Update Data	Yes	⇄ ▾**parameterTypeCreate** {
				all of:
				▾ {
				name: string *
				code: string *
				user_parameter: boolean
				}
				}

parameter_type_id	path	Parameter Type Id	Yes	⇄ `string`
X-AUTH-TOKEN	header	User Auth Token	Yes	⇄ `string`
X-MODULE-TOKEN	header	Module Token	Yes	⇄ `string`

Responses

Code	Description	Schema

200	The updated Parameter Type	⇄ ▾**parameterTypeReply** {
		all of:
		▾ {
		id: integer
		name: string
		code: string
		user_parameter: boolean
		created_at: string
		updated_at: string
		deleted_at: string
		}
		}

401	Unauthorized	⇄ ▾**parameterTypeErrorDefault** {
		error: string
		}

404	Parameter Type not Found	⇄ ▾**parameterTypeErrorDefault** {
		error: string
		}

500	Failed to update Parameter Type	⇄ ▾**parameterTypeErrorDefault** {
		error: string
		}

Try this operation

`DELETE /parameterTypes/{parameter_type_id}`

Parameter Type

Summary

Delete a parameter Type

Parameters

Name	Located in	Description	Required	Schema
parameter_type_id	path	parameter Type Id	Yes	⇄ `string`
X-MODULE-TOKEN	header	Module Token	Yes	⇄ `string`
X-AUTH-TOKEN	header	User Auth Token	Yes	⇄ `string`

Responses

Code	Description	Schema

200	OK	⇄ ▾**parameterTypeDeleteReply** {
		string: string
		}

401	Unauthorized	⇄ ▾**parameterTypeErrorDefault** {
		error: string
		}

404	Parameter Type not Found	⇄ ▾**parameterTypeErrorDefault** {
		error: string
		}

500	Failed to delete mMdule Type	⇄ ▾**parameterTypeErrorDefault** {
		error: string
		}

Try this operation

`/parameterTypes`

`POST /parameterTypes`

Parameter Type

Summary

Create a Parameter Type

Parameters

Name	Located in	Description	Required	Schema
ParameterType	body	Parameter Type Data	Yes	⇄

```
▾parameterTypeCreate {
  all of:
    ▾ {
        name:            string
                         *
        code:            string
                         *
        user_parameter:  boolean
      }
 }
```

Name	Located in	Description	Required	Schema
X-AUTH-TOKEN	header	User Auth Token	Yes	⇄ string
X-MODULE-TOKEN	header	Module Token	Yes	⇄ string

Responses

Code	Description	Schema
201	the newly createdParameter Type	⇄

```
▾parameterTypeReply {
  all of:
    ▾ {
        id:              integer
        name:            string
        code:            string
        user_parameter:  boolean
        created_at:      string
        updated_at:      string
        deleted_at:      string
      }
 }
```

Code	Description	Schema
401	Unauthorized	⇄

```
▾parameterTypeErrorDefault {
   error: string
 }
```

Code	Description	Schema
404	Parameter Type not found	⇄

```
▾parameterTypeErrorDefault {
   error: string
 }
```

Code	Description	Schema
500	Failed to store Parameter Type	⇄

```
▾parameterTypeErrorDefault {
   error: string
 }
```

Try this operation

`/parameterUserOption/{parameter_user_option_key}`

`GET /parameterUserOption/{parameter_user_option_key}`

Parameter User Option

Summary

Show a Parameter User Option

Parameters

Name	Located in	Description	Required	Schema
parameter_user_option_key	path	Parameter User Option key	Yes	⇄ `string`
X-MODULE-TOKEN	header	Module Token	Yes	⇄ `string`

Responses

Code **Description** **Schema**

200 Show the Parameter User Option data ⇄

```
▾parameterUserOptionShowReply {
   all of:
     ▾ {
        id:                        integer
        parameter_user_type_id:    integer
        parameter_user_option_key: string
        name:                      string
        created_at:                string
        updated_at:                string
        deleted_at:                string
     }
 }
```

401 Unauthorized ⇄

```
▾parameterUserOptionErrorDefault {
   error: string
 }
```

404 Parameter User Option not Found ⇄

```
▾parameterUserOptionErrorDefault {
   error: string
 }
```

500 Failed to retrieve Parameter User Option ⇄

```
▾parameterUserOptionErrorDefault {
   error: string
 }
```

Try this operation

PUT /parameterUserOption/{parameter_user_option_key}

Parameter User Option

Summary

Update a Parameter User Option

Parameters

Name	Located in	Description	Required	Schema
Body	body	Parameter User Option Update Data	Yes	⇄
parameter_user_option_key	path	Parameter User Option Key	Yes	⇄ string
X-AUTH-TOKEN	header	User Auth Token	Yes	⇄ string
X-MODULE-TOKEN	header	Module Token	Yes	⇄ string

Body

```
▾parameterUserOptionCre
  all of:
    ▾ {
        parameter_user_ty

        translations:
      }
}
```

Responses

Code	Description	Schema
200	The updated Parameter User Option	⇄
401	Unauthorized	⇄
404	Parameter User Option not Found	⇄
500	Failed to update Parameter User Option	⇄

200

```
▾parameterUserOptionReply {
  all of:
    ▾ {
        id:                        integer
        parameter_user_type_id:    integer
        parameter_user_option_key: string
        created_at:                string
        updated_at:                string
        deleted_at:                string
      }
}
```

401

```
▾parameterUserOptionErrorDefault {
  error: string
}
```

404

```
▾parameterUserOptionErrorDefault {
  error: string
}
```

500

```
▾parameterUserOptionErrorDefault {
  error: string
}
```

Try this operation

DELETE /parameterUserOption/{parameter_user_option_key}

Parameter User Option

Summary

Delete a parameter User Option

Parameters

Name	Located in	Description	Required	Schema
parameter_user_option_key	path	parameter User Option Key	Yes	⇄ `string`
X-MODULE-TOKEN	header	Module Token	Yes	⇄ `string`
X-AUTH-TOKEN	header	User Auth Token	Yes	⇄ `string`

Responses

Code	Description	Schema

200	OK	⇄ ▾`parameterUserOptionDeleteReply {`
		`string: string`
		`}`

401	Unauthorized	⇄ ▾`parameterUserOptionErrorDefault {`
		`error: string`
		`}`

404	Parameter User Option not Found	⇄ ▾`parameterUserOptionErrorDefault {`
		`error: string`
		`}`

500	Failed to delete mMdule Type	⇄ ▾`parameterUserOptionErrorDefault {`
		`error: string`
		`}`

Try this operation

/parameterUserOption

`POST /parameterUserOption`

Parameter User Option

Summary

Create a Parameter User Option

Parameters

Name	Located in	Description	Required	Schema
ParameterUserOption	body	Parameter User Option Data	Yes	⇄
X-AUTH-TOKEN	header	User Auth Token	Yes	⇄ string
X-MODULE-TOKEN	header	Module Token	Yes	⇄ string

ParameterUserOption:

```
▾parameterUserOptionCreate {
  all of:
    ▾ {
        parameter_user_type_ke

        translations:
      }
}
```

Responses

Code	Description	Schema
201	the newly createdParameter User Option	⇄
401	Unauthorized	⇄
404	Parameter User Option not found	⇄
500	Failed to store Parameter User Option	⇄

201:

```
▾parameterUserOptionReply {
  all of:
    ▾ {
        id:                        integer
        parameter_user_type_id:    integer
        parameter_user_option_key: string
        created_at:                string
        updated_at:                string
        deleted_at:                string
      }
}
```

401:

```
▾parameterUserOptionErrorDefault {
  error: string
}
```

404:

```
▾parameterUserOptionErrorDefault {
  error: string
}
```

500:

```
▾parameterUserOptionErrorDefault {
  error: string
}
```

Try this operation

`/parameterUserType/{parameter_user_type_key}`

GET /parameterUserType/{parameter_user_type_key}

Parameter User Type

Summary

Show a Parameter User Type

Parameters

Name	Located in	Description	Required	Schema
parameter_user_type_key	path	Parameter User Type key	Yes	⇄ integer
LANG-CODE	header	User Language	Yes	⇄ string
LANG-CODE-DEFAULT	header	Entity default Language	Yes	⇄ string
X-MODULE-TOKEN	header	Module Token	Yes	⇄ string

Responses

Code	Description	Schema
200	Show the Parameter User Type data	⇄

```
▾parameterUserTypeShowReply {
  all of:
    ▾ {
        id:                       integer
        parameter_user_type_id:   integer
        parameter_user_type_key:  string
        name:                     string
        parameter_user_options:   ▾[
                                     ▸
                                    parameterUserOptionShowRep
                                    { }
                                  ]
        created_at:               string
        updated_at:               string
        deleted_at:               string
      }
}
```

Code	Description	Schema
401	Unauthorized	⇄

```
▾parameterUserTypeErrorDefault {
   error: string
}
```

Code	Description	Schema
404	Parameter User Type not Found	⇄

```
▾parameterUserTypeErrorDefault {
   error: string
}
```

Code	Description	Schema
500	Failed to retrieve Parameter User Type	⇄

```
▾parameterUserTypeErrorDefault {
   error: string
}
```

Try this operation

```
/parameterUserTypes
```

`POST /parameterUserTypes`

Parameter User Type

Summary

Create a Parameter User Type

Parameters

Name	Located in	Description	Required	Schema
ParameterUserType	body	Parameter User Type Data	Yes	⇄
X-AUTH-TOKEN	header	User Auth Token	Yes	⇄ string
X-ENTITY-KEY	header	ENtity Key	Yes	⇄ string
X-MODULE-TOKEN	header	Module Token	Yes	⇄ string

```
▾parameterUserTypeCreate {
  all of:
    ▾ {
        parameter_type_code:

        mandatory:

        translations:
        parameter_user_options:
    }
}
```

Responses

Code	Description	Schema
201	the newly createdParameter User Type	⇄
401	Unauthorized	⇄
404	Parameter User Type not found	⇄

201:

```
▾parameterUserTypeReply {
  all of:
    ▾ {
        id:                       integer
        parameter_user_type_key:  string
        parameter_type_id:        integer
        entity_id:                integer
        mandatory:                boolean
        level_parameter_id:       integer
        created_at:               string
        updated_at:               string
        deleted_at:               string
    }
}
```

401:

```
▾parameterUserTypeErrorDefault {
  error: string
}
```

404:

```
▾parameterUserTypeErrorDefault {
  error: string
}
```

Code	Description	Schema

500	Failed to store Parameter User Type	⇄ ▾**parameterUserTypeErrorDefault {**
		`error: string`
		}

Try this operation

/parameterUserTypes/{parameter_user_type_key}

`PUT /parameterUserTypes/{parameter_user_type_key}`

Parameter User Type

Summary

Update a Parameter User Type

Parameters

Name	Located in	Description	Required	Schema
Body	body	Parameter User Type Update Data	Yes	⇄

```
▾parameterUserTypeCreate
  all of:
    ▾ {
      parameter_type_code

      mandatory:

      translations:
      parameter_user_opti
    }
}
```

Name	Located in	Description	Required	Schema
parameter_user_type_key	path	Parameter User Type Key	Yes	⇄ `string`
X-AUTH-TOKEN	header	User Auth Token	Yes	⇄ `string`
X-MODULE-TOKEN	header	Module Token	Yes	⇄ `string`

Responses

Code	Description	Schema
200	The updated Parameter User Type	⇄

```
▾parameterUserTypeReply {
  all of:
    ▾ {
      id:                      integer
      parameter_user_type_key: string
      parameter_type_id:       integer
      entity_id:               integer
      mandatory:               boolean
      level_parameter_id:      integer
      created_at:              string
      updated_at:              string
      deleted_at:              string
    }
}
```

Code	Description	Schema
401	Unauthorized	⇄

```
▾parameterUserTypeErrorDefault {
   error: string
}
```

Code	Description	Schema

404	Parameter User Type not Found	⇄ ▾**parameterUserTypeErrorDefault {**
		`error: string`
		}

500	Failed to update Parameter User Type	⇄ ▾**parameterUserTypeErrorDefault {**
		`error: string`
		}

Try this operation

DELETE /parameterUserTypes/{parameter_user_type_key}

Parameter User Type

Summary

Delete a parameter User Type

Parameters

Name	Located in	Description	Required	Schema
parameter_user_type_key	path	parameter User Type Key	Yes	⇄ `string`
X-MODULE-TOKEN	header	Module Token	Yes	⇄ `string`
X-AUTH-TOKEN	header	User Auth Token	Yes	⇄ `string`

Responses

Code	Description	Schema

200	OK	⇄ ▾**parameterUserTypeDeleteReply {**
		`string: string`
		}

401	Unauthorized	⇄ ▾**parameterUserTypeErrorDefault {**
		`error: string`
		}

404	Parameter User Type not Found	⇄ ▾**parameterUserTypeErrorDefault {**
		`error: string`
		}

500	Failed to delete mMdule Type	⇄ ▾**parameterUserTypeErrorDefault {**
		`error: string`
		}

Try this operation

/parameter

POST /parameter

Parameter

Summary

Create a Parameter

Parameters

Name	Located in	Description	Required	Schema
parameter	body	Parameter Data	Yes	⇄ parameterCreate
X-AUTH-TOKEN	header	User Auth Token	Yes	⇄ string
X-ENTITY-KEY	header	Entity Key	Yes	⇄ string
X-MODULE-TOKEN	header	Module Token	Yes	⇄ string

parameter:

```
▾parameterCreate {
  all of:
     ▾ {
        parameter_type_id: integer
                           *
        parameter:         string
                           *
        description:       string
                           *
        code:              string
                           *
     }
}
```

Responses

Code	Description	Schema
201	the newly created Parameter	⇄ parameterReply
401	Unauthorized	⇄ parameterErrorDefault
404	Entity not found	⇄ parameterErrorDefault

201:

```
▾parameterReply {
  all of:
     ▾ {
        id:                integer
        parameter_type_id: integer
        entity_id:         integer
        parameter:         string
        description:       string
        code:              string
        created_at:        string
        updated_at:        string
        deleted_at:        string
     }
}
```

401:

```
▾parameterErrorDefault {
  error: string
}
```

404:

```
▾parameterErrorDefault {
  error: string
}
```

Code	Description	Schema

500	Failed to store Parameter	⇄ ▾ **parameterErrorDefault {**
		`error: string`
		}

Try this operation

/parameter/{parameter_id}

PUT /parameter/{parameter_id}

Parameter

Summary

Update an Parameter

Parameters

Name	**Located in**	**Description**	**Required**	**Schema**
Body	body	Parameter Update Data	Yes	⇄
parameter_id	path	Parameter Id	Yes	⇄ `integer`
X-AUTH-TOKEN	header	User Auth Token	Yes	⇄ `string`
X-MODULE-TOKEN	header	Module Token	Yes	⇄ `string`

```
▾parameterCreate {
  all of:
    ▾ {
       parameter_type_id: integer
                          *
       parameter:         string
                          *
       description:       string
                          *
       code:              string
                          *
    }
}
```

Responses

Code	**Description**	**Schema**
200	The updated Parameter	⇄
401	Unauthorized	⇄

```
▾parameterReply {
  all of:
    ▾ {
       id:                integer
       parameter_type_id: integer
       entity_id:         integer
       parameter:         string
       description:       string
       code:              string
       created_at:        string
       updated_at:        string
       deleted_at:        string
    }
}
```

```
▾parameterErrorDefault {
  error: string
}
```

Code	Description	Schema

404	Parameter not Found	⇄ `▾parameterErrorDefault {`
		`error: string`
		`}`

500	Failed to update Parameter	⇄ `▾parameterErrorDefault {`
		`error: string`
		`}`

Try this operation

`DELETE /parameter/{parameter_id}`

Parameter

Summary

Delete a Parameter

Parameters

Name	Located in	Description	Required	Schema
parameter_id	path	Parameter Id	Yes	⇄ `integer`
X-MODULE-TOKEN	header	Module Token	Yes	⇄ `string`
X-AUTH-TOKEN	header	User Auth Token	Yes	⇄ `string`

Responses

Code	Description	Schema

200	OK	⇄ `▾parameterDeleteReply {`
		`string: string`
		`}`

401	Unauthorized	⇄ `▾parameterErrorDefault {`
		`error: string`
		`}`

404	Parameter not Found	⇄ `▾parameterErrorDefault {`
		`error: string`
		`}`

500	Failed to delete Parameter	⇄ `▾parameterErrorDefault {`
		`error: string`
		`}`

Try this operation

`/permissions/{permission_id}`

GET /permissions/{permission_id}

Permission

Summary

Show a Permission

Parameters

Name	Located in	Description	Required	Schema
permission_id	path	Permission Id	Yes	⇄ integer
X-MODULE-TOKEN	header	Module Token	Yes	⇄ string

Responses

Code	Description	Schema
200	Show the Permission data	⇄

```
▾permissionReply {
  all of:
    ▾ {
        id:         integer
        role_id:    integer
        code:       string
        module:     string
        api:        string
        create:     boolean
        view:       boolean
        update:     boolean
        delete:     boolean
        created_at: string
        updated_at: string
        deleted_at: string
      }
}
```

Code	Description	Schema
401	Unauthorized	⇄

```
▾permissionErrorDefault {
   error: string
}
```

Code	Description	Schema
404	Permission not Found	⇄

```
▾permissionErrorDefault {
   error: string
}
```

Code	Description	Schema
500	Failed to retrieve Permission	⇄

```
▾permissionErrorDefault {
   error: string
}
```

Try this operation

`PUT /permissions/{permission_id}`

Permission

Summary

Update an Permission

Parameters

Name	Located in	Description	Required	Schema
Body	body	Permission Update Data	Yes	⇄

```
▾permissionCreate {
  all of:
    ▾ {
      role_key: string
      code:     string
                *
      module:   string
                *
      api:      string
                *
      create:   boolean
                *
      view:     boolean
                *
      update:   boolean
                *
      delete:   boolean
                *
    }
}
```

Name	Located in	Description	Required	Schema
permission_id	path	Permission Id	Yes	⇄ `integer`
X-AUTH-TOKEN	header	User Auth Token	Yes	⇄ `string`
X-MODULE-TOKEN	header	Module Token	Yes	⇄ `string`

Responses

Code	Description	Schema
200	The updated Permission	⇄

```
▾permissionReply {
  all of:
    ▾ {
      id:         integer
      role_id:    integer
      code:       string
      module:     string
      api:        string
      create:     boolean
      view:       boolean
      update:     boolean
      delete:     boolean
      created_at: string
      updated_at: string
      deleted_at: string
    }
}
```

Code	Description	Schema

401	Unauthorized	⇄ ▾**permissionErrorDefault** {
		`error: string`
		}

404	Permission not Found	⇄ ▾**permissionErrorDefault** {
		`error: string`
		}

500	Failed to update Permission	⇄ ▾**permissionErrorDefault** {
		`error: string`
		}

Try this operation

`DELETE /permissions/{permission_id}`

Permission

Summary

Delete a Permission

Parameters

Name	Located in	Description	Required	Schema
permission_id	path	Permission Id	Yes	⇄ `integer`
X-MODULE-TOKEN	header	Module Token	Yes	⇄ `string`
X-AUTH-TOKEN	header	User Auth Token	Yes	⇄ `string`

Responses

Code	Description	Schema

200	OK	⇄ ▾**permissionDeleteReply** {
		`string: string`
		}

401	Unauthorized	⇄ ▾**permissionErrorDefault** {
		`error: string`
		}

404	Permission not Found	⇄ ▾**permissionErrorDefault** {
		`error: string`
		}

500	Failed to delete Permission	⇄ ▾**permissionErrorDefault** {
		`error: string`
		}

Try this operation

`/permissions`

`POST /permissions`

Permission

Summary

Create a Permission

Parameters

Name	Located in	Description	Required	Schema
permission	body	Permission Data	Yes	⇄
X-AUTH-TOKEN	header	User Auth Token	Yes	⇄ string
X-ENTITY-KEY	header	Entity Key	Yes	⇄ string
X-MODULE-TOKEN	header	Module Token	Yes	⇄ string

permission:

```
▾permissionCreate {
  all of:
    ▾ {
        role_key: string
        code:     string
                  *
        module:   string
                  *
        api:      string
                  *
        create:   boolean
                  *
        view:     boolean
                  *
        update:   boolean
                  *
        delete:   boolean
                  *
      }
}
```

Responses

Code	Description	Schema

201 the newly created Permission ⇄

```
▾permissionReply {
  all of:
    ▾ {
        id:         integer
        role_id:    integer
        code:       string
        module:     string
        api:        string
        create:     boolean
        view:       boolean
        update:     boolean
        delete:     boolean
        created_at: string
        updated_at: string
        deleted_at: string
      }
}
```

401 Unauthorized ⇄

```
▾permissionErrorDefault {
   error: string
}
```

404 Role not found ⇄

```
▾permissionErrorDefault {
   error: string
}
```

500 Failed to store Permission ⇄

```
▾permissionErrorDefault {
   error: string
}
```

Try this operation

`/proposal/{proposal_key}`

`GET /proposal/{proposal_key}`

Proposal

Summary

Show a Proposal

Parameters

Name	Located in	Description	Required	Schema
proposal_key	path	Proposal Key	Yes	⇄ `string`
X-MODULE-TOKEN	header	Module Token	Yes	⇄ `string`
X-ENTITY-KEY	header	Entity Token	Yes	⇄ `string`

Responses

Code	Description	Schema
200	Show the Proposal data	⇄

```
▾proposalReply {
  all of:
    ▾ {
       id:           integer
       proposal_key: string
       cb_key:       string
       entity_id:    integer
       created_at:   string
       updated_at:   string
       deleted_at:   string
    }
}
```

Code	Description	Schema
401	Unauthorized	⇄

```
▾proposalErrorDefault {
  error: string
}
```

Code	Description	Schema
404	Proposal not Found	⇄

```
▾proposalErrorDefault {
  error: string
}
```

Code	Description	Schema
500	Failed to retrieve Proposal	⇄

```
▾proposalErrorDefault {
  error: string
}
```

Try this operation

`DELETE /proposal/{proposal_key}`

Proposal

Summary

Delete a Proposal

Parameters

Name	Located in	Description	Required	Schema
proposal_key	path	Proposal Key	Yes	⇄ `integer`
X-MODULE-TOKEN	header	Module Token	Yes	⇄ `string`
X-AUTH-TOKEN	header	User Auth Token	Yes	⇄ `string`

Responses

Code	Description	Schema

200	OK	⇄ ▾**`proposalDeleteReply {`**
		`string: string`
		`}`

401	Unauthorized	⇄ ▾**`proposalErrorDefault {`**
		`error: string`
		`}`

404	Proposal not Found	⇄ ▾**`proposalErrorDefault {`**
		`error: string`
		`}`

500	Failed to delete Proposal	⇄ ▾**`proposalErrorDefault {`**
		`error: string`
		`}`

Try this operation

`/proposals`

`POST /proposals`

Proposal

Summary

Create a Proposal

Parameters

Name	Located in	Description	Required	Schema
proposal	body	Proposal Data	Yes	⇄
X-AUTH-TOKEN	header	User Auth Token	Yes	⇄ string
X-ENTITY-KEY	header	Entity Key	Yes	⇄ string
X-MODULE-TOKEN	header	Module Token	Yes	⇄ string

proposal

```
▾proposalCreate {
  all of:
    ▾ {
       role_key: string
       code:     string
                 *
       module:   string
                 *
       api:      string
                 *
       create:   boolean
                 *
       view:     boolean
                 *
       update:   boolean
                 *
       delete:   boolean
                 *
     }
}
```

Responses

Code	Description	Schema
201	the newly created Proposal	⇄
401	Unauthorized	⇄

201

```
▾proposalReply {
  all of:
    ▾ {
       id:           integer
       proposal_key: string
       cb_key:       string
       entity_id:    integer
       created_at:   string
       updated_at:   string
       deleted_at:   string
     }
}
```

401

```
▾proposalErrorDefault {
   error: string
}
```

Code	Description	Schema

404	Role not found	⇄ ▾**`proposalErrorDefault {`**
		`error: string`
		`}`

500	Failed to store Proposal	⇄ ▾**`proposalErrorDefault {`**
		`error: string`
		`}`

Try this operation

/proposals/{proposal_key}

`PUT /proposals/{proposal_key}`

Proposal

Summary

Update a Proposal

Parameters

Name	Located in	Description	Required	Schema
Body	body	Proposal Update Data	Yes	⇄ `▾proposalCreate { all of: ▾ { role_key: string, code: string *, module: string *, api: string *, create: boolean *, view: boolean *, update: boolean *, delete: boolean * } }`
proposal_key	path	Proposal Key	Yes	⇄ `integer`
X-AUTH-TOKEN	header	User Auth Token	Yes	⇄ `string`
X-MODULE-TOKEN	header	Module Token	Yes	⇄ `string`

Body schema:

```
▾proposalCreate {
  all of:
    ▾ {
        role_key: string
        code:     string
                  *
        module:   string
                  *
        api:      string
                  *
        create:   boolean
                  *
        view:     boolean
                  *
        update:   boolean
                  *
        delete:   boolean
                  *
      }
}
```

Responses

Code	Description	Schema
200	The updated Proposal	⇄ `▾proposalReply { all of: ▾ { id: integer, proposal_key: string, cb_key: string, entity_id: integer, created_at: string, updated_at: string, deleted_at: string } }`
401	Unauthorized	⇄ `▾proposalErrorDefault { error: string }`

200 schema:

```
▾proposalReply {
  all of:
    ▾ {
        id:           integer
        proposal_key: string
        cb_key:       string
        entity_id:    integer
        created_at:   string
        updated_at:   string
        deleted_at:   string
      }
}
```

401 schema:

```
▾proposalErrorDefault {
  error: string
}
```

Code **Description** **Schema**

404 Proposal not Found ⇄

```
▾proposalErrorDefault {
   error: string
}
```

500 Failed to update Proposal ⇄

```
▾proposalErrorDefault {
   error: string
}
```

Try this operation

/role/{role_key}

GET /role/{role_key}

Role

Summary

Show a Role

Parameters

Name	Located in	Description	Required	Schema
role_key	path	Role Key	Yes	⇄ string
X-MODULE-TOKEN	header	Module Token	Yes	⇄ string
X-ENTITY-KEY	header	Entity Key	Yes	⇄ string

Responses

Code **Description** **Schema**

200 Show the Role data ⇄

```
▾role {
  all of:
    ▾ {
        entity_id:   integer *
        role_key:    string *
        name:        string *
        code:        string *
        description: string *
    }
}
```

404 Role not Found ⇄

```
▾roleErrorDefault {
   error: string *
}
```

500 Failed to retrieve Role ⇄

```
▾roleErrorDefault {
   error: string *
}
```

Try this operation

`PUT /role/{role_key}`

Role

Summary

Update a Role

Parameters

Name	Located in	Description	Required	Schema
Body	body	Role Update data	Yes	⇄
role_key	path	Role Key	Yes	⇄ string
X-AUTH-TOKEN	header	User Auth Token	Yes	⇄ string
X-MODULE-TOKEN	header	Module Token	Yes	⇄ string

Body

```
▾role {
  all of:
    ▾ {
        entity_id:   integer *
        role_key:    string *
        name:        string *
        code:        string *
        description: string *
    }
}
```

Responses

Code	Description	Schema
200	The updated Role	⇄
401	Unauthorized	⇄
404	Role not Found	⇄
500	Failed to update Role	⇄

200

```
▾role {
  all of:
    ▾ {
        entity_id:   integer *
        role_key:    string *
        name:        string *
        code:        string *
        description: string *
    }
}
```

401

```
▾roleErrorDefault {
  error: string *
}
```

404

```
▾roleErrorDefault {
  error: string *
}
```

500

```
▾roleErrorDefault {
  error: string *
}
```

Try this operation

`DELETE /role/{role_key}`

Role

Summary

Delete a Role

Parameters

Name	Located in	Description	Required	Schema
role_key	path	Role Key	Yes	⇄ `string`
X-MODULE-TOKEN	header	Module Token	Yes	⇄ `string`
X-AUTH-TOKEN	header	Authentication Token	Yes	⇄ `string`

Responses

Code	Description	Schema

200	OK	⇄ ▾**replyDeleteRole {**
		`string: string`
		}

401	Unauthorized	⇄ ▾**roleErrorDefault {**
		`error: string *`
		}

404	Role not Found	⇄ ▾**roleErrorDefault {**
		`error: string *`
		}

500	Failed to delete Role	⇄ ▾**roleErrorDefault {**
		`error: string *`
		}

Try this operation

/role

POST /role

Role

Summary

Creation of a Role

Parameters

Name	Located in	Description	Required	Schema
Body	body	Role data	Yes	⇄ role
X-AUTH-TOKEN	header	User Auth Token	Yes	⇄ string
X-MODULE-TOKEN	header	Module Token	Yes	⇄ string
X-ENTITY-KEY	header	Entity Key	Yes	⇄ string

Body:

```
▾role {
   all of:
      ▾ {
         entity_id:   integer
                      *
         role_key:    string
                      *
         name:        string
                      *
         code:        string
                      *
         description: string
                      *
      }
}
```

Responses

Code	Description	Schema
201	the newly created role	⇄ role
401	Unauthorized	⇄ roleErrorDefault
500	Failed to store new Role	⇄ roleErrorDefault

201:

```
▾role {
   all of:
      ▾ {
         entity_id:   integer *
         role_key:    string *
         name:        string *
         code:        string *
         description: string *
      }
}
```

401:

```
▾roleErrorDefault {
   error: string *
}
```

500:

```
▾roleErrorDefault {
   error: string *
}
```

Try this operation

/SiteConf/{site_configuration_key}

GET /SiteConf/{site_configuration_key}

Site Configuration

Summary

Show a Site Configuration

Parameters

Name	Located in	Description	Required	Schema
site_configuration_key	path	Site Configuration Key	Yes	⇄ string
X-MODULE-TOKEN	header	Module Token	Yes	⇄ string

Responses

Code	Description	Schema
200	Show the Site Configuration data	⇄
401	Unauthorized	⇄
404	Site Configuration not Found	⇄
500	Failed to retrieve Site Configuration	⇄

200 Show the Site Configuration data

```
▾siteConfigurationReply {
   all of:
     ▾ {
        id:                integer
        site_conf_key:     string
        code:              string
        site_conf_group_id: integer
        created_at:        string
        updated_at:        string
        deleted_at:        string
      }
}
```

401 Unauthorized

```
▾siteConfigurationErrorDefault {
   error: string
}
```

404 Site Configuration not Found

```
▾siteConfigurationErrorDefault {
   error: string
}
```

500 Failed to retrieve Site Configuration

```
▾siteConfigurationErrorDefault {
   error: string
}
```

Try this operation

PUT /SiteConf/{site_configuration_key}

Site Configuration

Summary

Update a Site Configuration

Parameters

Name	Located in	Description	Required	Schema
Body	body	Site Configuration Data	Yes	⇄
site_configuration_key	path	Site Configuration Key	Yes	⇄ `string`
X-AUTH-TOKEN	header	User Auth Token	Yes	⇄ `string`
X-MODULE-TOKEN	header	Module Token	Yes	⇄ `string`

```
▾siteConfigurationCreate {
  all of:
     ▾ {
        request:  ▾[
                     ▸
                    siteConf
                    { }
                 ]
     }
}
```

Responses

Code	Description	Schema
200	The updated Site Configuration	⇄
401	Unauthorized	⇄
404	Site Configuration not Found	⇄
500	Failed to update Site Configuration	⇄

```
▾siteConfigurationReply {
  all of:
     ▾ {
        id:                 integer
        site_conf_key:      string
        code:               string
        site_conf_group_id: integer
        created_at:         string
        updated_at:         string
        deleted_at:         string
     }
}
```

```
▾siteConfigurationErrorDefault {
  error: string
}
```

```
▾siteConfigurationErrorDefault {
  error: string
}
```

```
▾siteConfigurationErrorDefault {
  error: string
}
```

Try this operation

`DELETE /SiteConf/{site_configuration_key}`

Site Configuration

Summary

Delete a Site Configuration

Parameters

Name	Located in	Description	Required	Schema
site_configuration_key	path	Site Configuration Key	Yes	⇄ `string`
X-MODULE-TOKEN	header	Module Token	Yes	⇄ `string`
X-AUTH-TOKEN	header	User Auth Token	Yes	⇄ `string`

Responses

Code	Description	Schema

200	OK	⇄ ▾**siteConfigurationDeleteReply {**
		`string: string`
		}

401	Unauthorized	⇄ ▾**siteConfigurationErrorDefault {**
		`error: string`
		}

404	Site Configuration not Found	⇄ ▾**siteConfigurationErrorDefault {**
		`error: string`
		}

500	Failed to delete Site Configuration	⇄ ▾**siteConfigurationErrorDefault {**
		`error: string`
		}

Try this operation

`/SiteConf`

`POST /SiteConf`

Site Configuration

Summary

Create a Site Configuration

Parameters

Name	Located in	Description	Required	Schema
Site Configuration	body	Site Configuration Data	Yes	⇄
X-AUTH-TOKEN	header	User Auth Token	Yes	⇄ `string`
X-MODULE-TOKEN	header	Module Token	Yes	⇄ `string`

Site Configuration:

```
▾siteConfigurationCreate {
  all of:
    ▾ {
        request: ▾[
                   ▸
                   siteConfiguratio
                   { }
                 ]
      }
}
```

Responses

Code	Description	Schema
201	the newly created Site Configuration	⇄
401	Unauthorized	⇄
404	Site Configuration not found	⇄
500	Failed to store Site Configuration	⇄

201:

```
▾siteConfigurationReply {
  all of:
    ▾ {
        id:                 integer
        site_conf_key:      string
        code:               string
        site_conf_group_id: integer
        created_at:         string
        updated_at:         string
        deleted_at:         string
      }
}
```

401:

```
▾siteConfigurationErrorDefault {
  error: string
}
```

404:

```
▾siteConfigurationErrorDefault {
  error: string
}
```

500:

```
▾siteConfigurationErrorDefault {
  error: string
}
```

Try this operation

`/siteConfGroup/{site_configuration_group_key}`

`GET /siteConfGroup/{site_configuration_group_key}`

Site Configuration Group

Summary

Show a Site Configuration Group

Parameters

Name	Located in	Description	Required	Schema
site_configuration_group_key	path	Site Configuration Group Key	Yes	⇄ `string`
X-MODULE-TOKEN	header	Module Token	Yes	⇄ `string`

Responses

Code	Description	Schema
200	Show the Site Configuration Group data	⇄

```
▾siteConfigurationGroupReply {
  all of:
    ▾ {
        id:                 integer
        site_conf_group_key: string
        code:               string
        created_at:         string
        updated_at:         string
        deleted_at:         string
    }
}
```

Code	Description	Schema
401	Unauthorized	⇄

```
▾
siteConfigurationGroupErrorDefault
{
   error: string
}
```

Code	Description	Schema
404	Site Configuration Group not Found	⇄

```
▾
siteConfigurationGroupErrorDefault
{
   error: string
}
```

Code	Description	Schema
500	Failed to retrieve Site Configuration Group	⇄

```
▾
siteConfigurationGroupErrorDefault
{
   error: string
}
```

Try this operation

PUT /siteConfGroup/{site_configuration_group_key}

Site Configuration Group

Summary

Update a Site Configuration Group

Parameters

Name	Located in	Description	Required	Schema
Body	body	Site Configuration Group Data	Yes	⇄
site_configuration_group_key	path	Site Configuration Group Key	Yes	⇄ string
X-AUTH-TOKEN	header	User Auth Token	Yes	⇄ string
X-MODULE-TOKEN	header	Module Token	Yes	⇄ string

Body:

```
▾siteConfigurationG
  all of:
     ▾ {
       request: ▾[
                  ▸
                 s
                 {
               ]
     }
}
```

Responses

Code	Description	Schema
200	The updated Site Configuration Group	⇄
401	Unauthorized	⇄
404	Site Configuration Group not Found	⇄

200:

```
▾siteConfigurationGroupReply {
  all of:
     ▾ {
       id:                 integer
       site_conf_group_key: string
       code:               string
       created_at:         string
       updated_at:         string
       deleted_at:         string
     }
}
```

401:

```
▾
siteConfigurationGroupErrorDefault
{
   error: string
}
```

404:

```
▾
siteConfigurationGroupErrorDefault
{
   error: string
}
```

Code	Description	Schema

500	Failed to update Site Configuration Group	⇄ ▾ `siteConfigurationGroupErrorDefault {`
		`error: string`
		`}`

Try this operation

`DELETE /siteConfGroup/{site_configuration_group_key}`

Site Configuration Group

Summary

Delete a Site Configuration Group

Parameters

Name	Located in	Description	Required	Schema
site_configuration_group_key	path	Site Configuration Group Key	Yes	⇄ `string`
X-MODULE-TOKEN	header	Module Token	Yes	⇄ `string`
X-AUTH-TOKEN	header	User Auth Token	Yes	⇄ `string`

Responses

Code	Description	Schema

200	OK	⇄ ▾ `siteConfigurationGroupDeleteReply {`
		`string: string`
		`}`

401	Unauthorized	⇄ ▾ `siteConfigurationGroupErrorDefault {`
		`error: string`
		`}`

404	Site Configuration not Found	⇄ ▾ `siteConfigurationGroupErrorDefault {`
		`error: string`
		`}`

500	Failed to delete Site Configuration	⇄ ▾ `siteConfigurationGroupErrorDefault {`
		`error: string`
		`}`

Try this operation

`/siteConfGroup`

`POST /siteConfGroup`

Site Configuration Group

Summary

Create a Site Configuration Group

Parameters

Name	Located in	Description	Required	Schema

Site Configuration Group	body	Site Configuration Group Data	Yes	⇄ ▾**siteConfigurationGroupCreate {**
				all of:
				▾ **{**
				request: ▾**[**
				▸ **siteConfiguratio**
				{ }
				]
				}
				}

X-AUTH-TOKEN	header	User Auth Token	Yes	⇄ `string`
X-MODULE-TOKEN	header	Module Token	Yes	⇄ `string`

Responses

Code	Description	Schema
201	the newly created Site Configuration Group	⇄ ▾**siteConfigurationGroupReply {** **all of:** ▾ **{** id: integer site_conf_group_key: string code: string created_at: string updated_at: string deleted_at: string } }
401	Unauthorized	⇄ ▾ **siteConfigurationGroupErrorDefault** **{** error: string }
404	Site Configuration Group not found	⇄ ▾ **siteConfigurationGroupErrorDefault** **{** error: string }
500	Failed to store Site Configuration Group	⇄ ▾ **siteConfigurationGroupErrorDefault** **{** error: string }

Try this operation

/site/{site_key}

`GET /site/{site_key}`

Site

Summary

Show a Site

Parameters

Name	Located in	Description	Required	Schema
site_key	path	Site Key	Yes	⇄ `string`
X-MODULE-TOKEN	header	Module Token	Yes	⇄ `string`

Responses

Code	Description	Schema

Code **Description** **Schema**

200 Show the Site data ⇄

```
▾siteCreateReply {
   all of:
     ▾ {
         cm_key:          string
         site_key:        string
         entity_id:       string
         name:            string
         description:     string
         link:            string
         partial_link:    boolean
         active:          boolean
         no_reply_email:  string
         start_date:      string
         end_date:        string
         use_terms:       ▾[
                            ▾use_terms {
                              all of:
                                ▾ {
                                    site_id:       integer
                                    language_code: string
                                                   *
                                    content:       string
                                                   *
                                  }
                            }
                          ]
         layout:          ▾[
                            ▾layoutReply {
                              all of:
                                ▾ {
                                    id:         integer
                                    layout_key: string
                                    name:       string
                                    reference:  string
                                    created_at: string
                                    updated_at: string
                                    deleted_at: string
                                  }
                            }
                          ]
       }
}
```

404 Site not Found ⇄

```
▾siteErrorDefault {
   error: string *
}
```

500 Failed to retrieve Site ⇄

```
▾siteErrorDefault {
   error: string *
}
```

Try this operation

`PUT /site/{site_key}`

Site

Summary

Update a Site

Parameters

Name	Located in	Description	Required	Schema
Body	body	Site Update data	Yes	⇄
site_key	path	Site Key	Yes	⇄ `string`
X-AUTH-TOKEN	header	User Auth Token	Yes	⇄ `string`
X-MODULE-TOKEN	header	Module Token	Yes	⇄ `string`

```
▾siteCreateReply {
  all of:
    ▾ {
        cm_key:          string
        site_key:        string
        entity_id:       string
        name:            string
        description:     string
        link:            string
        partial_link:    boolean
        active:          boolean
        no_reply_email:  string
        start_date:      string
        end_date:        string
        use_terms:       ▾[
                            ▸use_terms
                           { }
                          ]
        layout:          ▾[
                            ▸
                           layoutReply
                           { }
                          ]
      }
}
```

Responses

Code	Description	Schema

200 The updated Site ⇄

```
▾siteCreateReply {
  all of:
    ▾ {
       cm_key:         string
       site_key:       string
       entity_id:      string
       name:           string
       description:    string
       link:           string
       partial_link:   boolean
       active:         boolean
       no_reply_email: string
       start_date:     string
       end_date:       string
       use_terms:      ▾[
                         ▾use_terms {
                           all of:
                             ▾ {
                                site_id:       integer
                                language_code: string
                                               *
                                content:       string
                                               *
                             }
                         }
                        ]
       layout:         ▾[
                         ▾layoutReply {
                           all of:
                             ▾ {
                                id:         integer
                                layout_key: string
                                name:       string
                                reference:  string
                                created_at: string
                                updated_at: string
                                deleted_at: string
                             }
                         }
                        ]
    }
}
```

401 Unauthorized ⇄

```
▾siteErrorDefault {
   error: string *
}
```

404 Site not Found ⇄

```
▾siteErrorDefault {
   error: string *
}
```

500 Failed to update Site ⇄

```
▾siteErrorDefault {
   error: string *
}
```

Try this operation

DELETE /site/{site_key}

Site

Summary

Delete a Site

Parameters

Name	Located in	Description	Required	Schema
site_key	path	Site Key	Yes	⇄ `string`
X-MODULE-TOKEN	header	Module Token	Yes	⇄ `string`
X-AUTH-TOKEN	header	User Auth Token	Yes	⇄ `string`
X-ENTITY-KEY	header	Entity Key	Yes	⇄ `string`

Responses

Code	Description	Schema

200	OK	⇄ `▾replyDeleteSite {`
		`string: string`
		`}`

401	Unauthorized	⇄ `▾siteErrorDefault {`
		`error: string *`
		`}`

404	Site not Found	⇄ `▾siteErrorDefault {`
		`error: string *`
		`}`

500	Failed to delete Site	⇄ `▾siteErrorDefault {`
		`error: string *`
		`}`

Try this operation

/site

POST /site

Site

Summary

Create a Site

Parameters

Name	Located in	Description	Required	Schema
Site	body	Site data	Yes	⇄
X-AUTH-TOKEN	header	User Auth Token	Yes	⇄ string
X-MODULE-TOKEN	header	Module Token	Yes	⇄ string
X-ENTITY-KEY	header	Entity Key	Yes	⇄ string

```
▾siteCreateReply {
  all of:
    ▾ {
      cm_key:          string
      site_key:        string
      entity_id:       string
      name:            string
      description:     string
      link:            string
      partial_link:    boolean
      active:          boolean
      no_reply_email:  string
      start_date:      string
      end_date:        string
      use_terms:       ▾[
                         ▸use_terms
                         { }
                       ]
      layout:          ▾[
                         ▸
                         layoutReply
                         { }
                       ]
    }
}
```

Responses

Code	Description	Schema

201 the newly created Site ⇄

```
▾siteCreateReply {
  all of:
    ▾ {
      cm_key:         string
      site_key:       string
      entity_id:      string
      name:           string
      description:    string
      link:           string
      partial_link:   boolean
      active:         boolean
      no_reply_email: string
      start_date:     string
      end_date:       string
      use_terms:      ▾[
                        ▾use_terms {
                          all of:
                            ▾ {
                              site_id:       integer
                              language_code: string
                                             *
                              content:       string
                                             *
                            }
                        }
                      ]
      layout:         ▾[
                        ▾layoutReply {
                          all of:
                            ▾ {
                              id:         integer
                              layout_key: string
                              name:       string
                              reference:  string
                              created_at: string
                              updated_at: string
                              deleted_at: string
                            }
                        }
                      ]
    }
}
```

---

401 Unauthorized ⇄

```
▾siteErrorDefault {
  error: string *
}
```

---

404 Site not Found ⇄

```
▾siteErrorDefault {
  error: string *
}
```

---

500 Failed to store new Site ⇄

```
▾siteErrorDefault {
  error: string *
}
```

Try this operation

/statusLogin

POST /statusLogin

Status Login

Summary

Create a Status Login

Parameters

Name	Located in	Description	Required	Schema
StatusLogin	body	Status Login Data	Yes	⇄
X-AUTH-TOKEN	header	User Auth Token	Yes	⇄ string
X-MODULE-TOKEN	header	Module Token	Yes	⇄ string

StatusLogin

```
▾statusLoginCreate {
  all of:
      ▾ {
          code: string *
      }
}
```

Responses

Code	Description	Schema
201	the newly created Status Login	⇄
401	Unauthorized	⇄
404	Status Login not found	⇄
500	Failed to store Status Login	⇄

201

```
▾statusLoginReply {
  all of:
      ▾ {
          id:         integer
          code:       string
          created_at: string
          updated_at: string
          deleted_at: string
      }
}
```

401

```
▾statusLoginErrorDefault {
   error: string
}
```

404

```
▾statusLoginErrorDefault {
   error: string
}
```

500

```
▾statusLoginErrorDefault {
   error: string
}
```

Try this operation

/timezone/{timezone_id}

GET /timezone/{timezone_id}

Timezone

Summary

Show a Timezone

Parameters

Name	Located in	Description	Required	Schema
timezone_id	path	Timezone Id	Yes	⇄ string
X-AUTH-TOKEN	header	User Auth Token	Yes	⇄ string
X-MODULE-TOKEN	header	Module Token	Yes	⇄ string

Responses

Code	Description	Schema
200	Show the Timezone data	⇄
404	Timezone not Found	⇄
500	Failed to retrieve Timezone	⇄

200 Show the Timezone data

```
▾replyTimezone {
   data: ▾ {
              all of:
                 ▾Timezone {
                    all of:
                       ▾ {
                          country_code: string *
                          name:         string *
                       }
                   }
              }
}
```

404 Timezone not Found

```
▾timezoneErrorDefault {
   error: string *
}
```

500 Failed to retrieve Timezone

```
▾timezoneErrorDefault {
   error: string *
}
```

Try this operation

`PUT /timezone/{timezone_id}`

Timezone

Summary

Update a Timezone

Parameters

Name	Located in	Description	Required	Schema
Body	body	Timezone Update data	Yes	⇄
timezone_id	path	Timezone Id	Yes	⇄ string
X-AUTH-TOKEN	header	User Auth Token	Yes	⇄ string
X-MODULE-TOKEN	header	Module Token	Yes	⇄ string

```
▾updateTimezone {
  timezone: ▾ {
              all of:
                ▸
                Timezone
                { }
            }
}
```

Responses

Code	Description	Schema
200	The updated Timezone	⇄

```
▾replyUpdatedTimezone {
   timezone: ▾ {
                all of:
                  ▾Timezone {
                     all of:
                       ▾ {
                          country_code: string *
                          name:         string *
                       }
                  }
             }
}
```

Code	Description	Schema
401	Unauthorized	⇄

```
▾timezoneErrorDefault {
   error: string *
}
```

Code	Description	Schema
404	Timezone not Found	⇄

```
▾timezoneErrorDefault {
   error: string *
}
```

Code	Description	Schema
500	Failed to update Timezone	⇄

```
▾timezoneErrorDefault {
   error: string *
}
```

Try this operation

`DELETE /timezone/{timezone_id}`

Timezone

Summary

Delete a Timezone

Parameters

Name	Located in	Description	Required	Schema
timezone_id	path	Timezone Id	Yes	⇄ `string`
X-MODULE-TOKEN	header	Module Token	Yes	⇄ `string`
X-AUTH-TOKEN	header	Authentication Token	Yes	⇄ `string`

Responses

Code	Description	Schema

200	OK	⇄ ▾**`replyDeleteTimezone {`**
		`string: string`
		`}`

401	Unauthorized	⇄ ▾**`timezoneErrorDefault {`**
		`error: string *`
		`}`

404	Timezone not Found'	⇄ ▾**`timezoneErrorDefault {`**
		`error: string *`
		`}`

500	Failed to delete Timezone	⇄ ▾**`timezoneErrorDefault {`**
		`error: string *`
		`}`

Try this operation

`/timezone`

`POST /timezone`

Timezone

Summary

Creation of a Timezone

Parameters

Name	Located in	Description	Required	Schema

Body	body	Timezone data	Yes	⇄ ▾**createTimezone** {
				timezone: ▾ {
				all of:
				▸**Timezone** { }
				}
				}

X-AUTH-TOKEN	header	User Auth Token	Yes	⇄ `string`
X-MODULE-TOKEN	header	Module Token	Yes	⇄ `string`

Responses

Code	Description	Schema

201	the newly created timezone	⇄ ▾**replyCreateTimezone** {
		timezone: ▾ {
		all of:
		▾**Timezone** {
		all of:
		▾ {
		country_code: string *
		name: string *
		}
		}
		}
		}

401	Unauthorized	⇄ ▾**timezoneErrorDefault** {
		error: string *
		}

500	Failed to store new Timezone	⇄ ▾**timezoneErrorDefault** {
		error: string *
		}

Try this operation

`/user/{user_key}`

`GET /user/{user_key}`

User

Summary

Show a User

Parameters

Name	Located in	Description	Required	Schema
user_key	path	User Key	Yes	⇄ `string`
X-MODULE-TOKEN	header	Module Token	Yes	⇄ `string`
X-ENTITY-KEY	header	Entity Key	Yes	⇄ `string`

Responses

Code	Description	Schema
200	Show the User data	⇄

```
▾user {
   all of:
     ▾ {
        user_key:           string *
        admin:              boolean *
        geographic_area_id: string *
     }
}
```

Code	Description	Schema
404	User not Found	⇄

```
▾userErrorDefault {
   error: string *
}
```

Code	Description	Schema
500	Failed to retrieve User	⇄

```
▾userErrorDefault {
   error: string *
}
```

Try this operation

PUT /user/{user_key}

User

Summary

Update a User

Parameters

Name	Located in	Description	Required	Schema
Body	body	User Update data	Yes	⇄
user_key	path	User Key	Yes	⇄ string
X-AUTH-TOKEN	header	User Auth Token	Yes	⇄ string
X-MODULE-TOKEN	header	Module Token	Yes	⇄ string

Body:

```
▾user {
  all of:
    ▾ {
        user_key:           string
                            *
        admin:              boolean
                            *
        geographic_area_id: string
                            *
      }
}
```

Responses

Code	Description	Schema
200	The updated User	⇄
401	Unauthorized	⇄
404	User not Found	⇄
500	Failed to update User	⇄

200:

```
▾user {
  all of:
    ▾ {
        user_key:           string *
        admin:              boolean *
        geographic_area_id: string *
      }
}
```

401:

```
▾userErrorDefault {
  error: string *
}
```

404:

```
▾userErrorDefault {
  error: string *
}
```

500:

```
▾userErrorDefault {
  error: string *
}
```

Try this operation

`DELETE /user/{user_key}`

User

Summary

Delete a User

Parameters

Name	Located in	Description	Required	Schema
user_key	path	User Key	Yes	⇄ `string`
X-MODULE-TOKEN	header	Module Token	Yes	⇄ `string`
X-AUTH-TOKEN	header	Authentication Token	Yes	⇄ `string`
X-ENTITY-KEY	header	Entity Key	Yes	⇄ `string`

Responses

Code	Description	Schema

200	OK	⇄ ▾**`replyDeleteUser {`**
		`string: string`
		`}`

401	Unauthorized	⇄ ▾**`userErrorDefault {`**
		`error: string *`
		`}`

404	User not Found	⇄ ▾**`userErrorDefault {`**
		`error: string *`
		`}`

500	Failed to delete User	⇄ ▾**`userErrorDefault {`**
		`error: string *`
		`}`

Try this operation

`/user`

`POST /user`

User

Summary

Creation of a User

Parameters

Name	Located in	Description	Required	Schema
Body	body	User data	Yes	⇄
X-AUTH-TOKEN	header	User Auth Token	Yes	⇄ string
X-MODULE-TOKEN	header	Module Token	Yes	⇄ string
X-ENTITY-KEY	header	Entity Key	Yes	⇄ string

Body:

```
▾user {
  all of:
    ▾ {
      user_key:              string
                             *
      admin:                 boolean
                             *
      geographic_area_id:    string
                             *
    }
}
```

Responses

Code	Description	Schema
201	the newly created user	⇄
401	Unauthorized	⇄
404	Unauthorized	⇄
409	Entity not Found	⇄
500	Failed to store new User	⇄

201:

```
▾user {
  all of:
    ▾ {
      user_key:           string *
      admin:              boolean *
      geographic_area_id: string *
    }
}
```

401:

```
▾userErrorDefault {
  error: string *
}
```

404:

```
▾userErrorDefault {
  error: string *
}
```

409:

```
▾userErrorDefault {
  error: string *
}
```

500:

```
▾userErrorDefault {
  error: string *
}
```

Try this operation

Models

accessMenuErrorDefault

```
  ▾accessMenuErrorDefault {
⇄    error: string *
  }
```

accessMenuCreateReply

```
  ▾accessMenuCreateReply {
    all of:
      ▾ {
        entity_id:   integer *
        site_id:     integer *
        name:        string *
⇄       description: string *
        active:      boolean *
        created_by:  string *
        updated_by:  string *
        site_key:    string
      }
  }
```

accessMenuShowReply

```
   ▾accessMenuShowReply {
     all of:
        ▾ {
          entity_id:   integer *
          site_id:     integer *
          name:        string *
          description: string *
          active:      boolean *
          created_by:  string *
          updated_by:  string *
          site:        ▾[
                         ▾site {
                           all of:
                              ▾ {
                                cm_key:         string
⇄                               site_key:       string
                                entity_id:      string
                                name:           string
                                description:    string
                                link:           string
                                partial_link:   boolean
                                active:         boolean
                                no_reply_email: string
                                start_date:     string
                                end_date:       string
                              }
                         }
                       ]
        }
   }
```

replyDeleteAccessMenu

```
   ▾replyDeleteAccessMenu {
⇄    string: string
   }
```

authMethodErrorDefault

```
   ▾authMethodErrorDefault {
⇄    error: string *
   }
```

authMethod

```
  ▾authMethod {
    all of:
      ▾ {
        auth_method_key: string *
⇄       name:            string *
        description:     string *
        code:            string *
      }
  }
```

replyDeleteAuthMethod

```
  ▾replyDeleteAuthMethod {
⇄   string: string
  }
```

categoryErrorDefault

```
  ▾categoryErrorDefault {
⇄   error: string *
  }
```

category

```
  ▾category {
    all of:
      ▾ {
        entity_id:   integer *
⇄       name:        string *
        description: string *
        created_by:  string *
      }
  }
```

categoryReply

```
  ▾categoryReply {
    all of:
      ▾ {
        category_key: string
        entity_id:    integer *
        name:         string *
⇄       description:  string *
        created_by:   string *
        created_at:   string
        updated_at:   string
        deleted_at:   string
      }
  }
```

replyDeleteCategory

```
▾replyDeleteCategory {
   string: string
}
```

padParameterTemplateErrorDefault

```
▾padParameterTemplateErrorDefault {
   error: string *
}
```

padParameterTemplateCreate

```
▾padParameterTemplateCreate {
  all of:
    ▾ {
      parameter_template_key: string *
    }
}
```

padParameterTemplateReply

```
▾padParameterTemplateReply {
  all of:
    ▾ {
      id:                     integer
      parameter_template_key: string
      entity_id:              integer
      created_at:             string
      updated_at:             string
      deleted_at:             string
    }
}
```

replyDeletePadParameterTemplate

```
▾replyDeletePadParameterTemplate {
   string: string
}
```

replyDeleteCoConstruction

```
▾replyDeleteCoConstruction {
   string: string
}
```

coConstructionErrorDefault

```
▾coConstructionErrorDefault {
   error: string *
}
```

coConstructionCreate

```
    ▾coConstructionCreate {
      all of:
         ▾ {
⇄          cb_key: string *
         }
    }
```

coConstructionUpdate

```
    ▾coConstructionUpdate {
      all of:
         ▾ {
⇄          cb_key: string *
         }
    }
```

coConstructionReply

```
    ▾coConstructionReply {
      all of:
         ▾ {
           id:                   integer
           co_construction_key: string
           cb_key:               string
⇄          entity_id:            integer
           created_at:           string
           updated_at:           string
           deleted_at:           string
         }
    }
```

countryErrorDefault

```
    ▾countryErrorDefault {
⇄     error: string *
    }
```

Country

```
    ▾Country {
      all of:
         ▾ {
           id:         integer *
           name:       string *
⇄          code:       string *
           created_at: string
           updated_at: string
           deleted_at: string
         }
    }
```

replyCountry

```
▾replyCountry {
   data: ▾ {
          all of:
             ▾Country {
               all of:
                 ▾ {
                    id:         integer *
                    name:       string *
                    code:       string *
                    created_at: string
                    updated_at: string
                    deleted_at: string
                  }
              }
         }
}
```

createCountry

```
▾createCountry {
   name: string *
   code: string *
}
```

replyCreateCountry

```
▾replyCreateCountry {
   country: ▾ {
             all of:
                ▾Country {
                  all of:
                    ▾ {
                       id:         integer *
                       name:       string *
                       code:       string *
                       created_at: string
                       updated_at: string
                       deleted_at: string
                     }
                 }
            }
}
```

updateCountry

```
▾updateCountry {
   name: string *
   code: string *
}
```

replyUpdatedCountry

```
▾replyUpdatedCountry {
   country: ▾ {
               all of:
                  ▾Country {
                     all of:
                        ▾ {
                           id:         integer *
                           name:       string *
                           code:       string *
                           created_at: string
                           updated_at: string
                           deleted_at: string
                        }
                  }
            }
}
```

replyDeleteCountry

```
▾replyDeleteCountry {
   string: string
}
```

currencyErrorDefault

```
▾currencyErrorDefault {
   error: string *
}
```

Currency

```
▾Currency {
  all of:
     ▾ {
        currency:       string *
        symbol_left:    string *
        symbol_right:   string *
        code:           string *
        decimal_place:  integer *
        decimal_point:  string *
        thousand_point: string *
     }
}
```

replyCurrency

```
▾replyCurrency {
   data: ▾ {
           all of:
             ▾Currency {
               all of:
                 ▾ {
                   currency:       string *
                   symbol_left:    string *
⇄                  symbol_right:   string *
                   code:           string *
                   decimal_place:  integer *
                   decimal_point:  string *
                   thousand_point: string *
                 }
             }
         }
}
```

createCurrency

```
▾createCurrency {
   currency: ▾ {
               all of:
                 ▾Currency {
                   all of:
                     ▾ {
                       currency:       string *
                       symbol_left:    string *
⇄                      symbol_right:   string *
                       code:           string *
                       decimal_place:  integer *
                       decimal_point:  string *
                       thousand_point: string *
                     }
                 }
             }
}
```

replyCreateCurrency

```
▾replyCreateCurrency {
   currency: ▾ {
                all of:
                   ▾Currency {
                     all of:
                        ▾ {
                          currency:       string *
                          symbol_left:    string *
⇄                         symbol_right:   string *
                          code:           string *
                          decimal_place:  integer *
                          decimal_point:  string *
                          thousand_point: string *
                        }
                   }
              }
}
```

updateCurrency

```
▾updateCurrency {
   currency: ▾ {
                all of:
                   ▾Currency {
                     all of:
                        ▾ {
                          currency:       string *
                          symbol_left:    string *
⇄                         symbol_right:   string *
                          code:           string *
                          decimal_place:  integer *
                          decimal_point:  string *
                          thousand_point: string *
                        }
                   }
              }
}
```

replyUpdatedCurrency

```
    ▾replyUpdatedCurrency {
      currency: ▾ {
                  all of:
                    ▾Currency {
                      all of:
                        ▾ {
                          currency:        string *
                          symbol_left:     string *
 ⇄                        symbol_right:    string *
                          code:            string *
                          decimal_place:   integer *
                          decimal_point:   string *
                          thousand_point:  string *
                        }
                    }
                }
    }
```

replyDeleteCurrency

```
    ▾replyDeleteCurrency {
 ⇄    string: string
    }
```

discussionErrorDefault

```
    ▾discussionErrorDefault {
 ⇄    error: string *
    }
```

discussionCreate

```
    ▾discussionCreate {
      all of:
        ▾ {
          pad_key:    integer *
 ⇄        start_date: string *
          end_date:   string *
          created_by: string *
        }
    }
```

discussionUpdate

```
⇄  ▾discussionUpdate {
     all of:
       ▾ {
         pad_key:    integer *
         start_date: string *
         end_date:   string *
         created_by: string *
         updated_by: string *
       }
   }
```

discussionReply

```
⇄  ▾discussionReply {
     all of:
       ▾ {
         pad_key:    integer *
         site_id:    integer *
         start_date: string *
         end_date:   string *
         created_by: string *
         updated_by: string *
         created_at: string
         updated_at: string
         deleted_at: string
       }
   }
```

replyDeleteDiscussion

```
   ▾replyDeleteDiscussion {
⇄     string: string
   }
```

entityErrorDefault

```
   ▾entityErrorDefault {
⇄     error: string *
   }
```

Entity

```
  ▾Entity {
    all of:
      ▾ {
        entity_key:  string *
        country_id:  integer *
        timezone_id: integer *
⇄       currency_id: integer *
        name:        string *
        designation: string *
        description: string *
        url:         string *
      }
  }
```

replyEntity

```
  ▾replyEntity {
    data: ▾ {
              all of:
                ▾Entity {
                  all of:
                    ▾ {
                      entity_key:  string *
                      country_id:  integer *
                      timezone_id: integer *
⇄                     currency_id: integer *
                      name:        string *
                      designation: string *
                      description: string *
                      url:         string *
                    }
                }
            }
  }
```

createEntity

```
 ▾createEntity {
    entity:      ▾ {
                   all of:
                     ▾Entity {
                       all of:
                         ▾ {
                            entity_key:  string *
                            country_id:  integer *
                            timezone_id: integer *
⇄                           currency_id: integer *
                            name:        string *
                            designation: string *
                            description: string *
                            url:         string *
                          }
                      }
                  }
    layout_key: string
 }
```

replyCreateEntity

```
 ▾replyCreateEntity {
    entity: ▾ {
              all of:
                ▾Entity {
                  all of:
                    ▾ {
                       entity_key:  string *
                       country_id:  integer *
                       timezone_id: integer *
⇄                      currency_id: integer *
                       name:        string *
                       designation: string *
                       description: string *
                       url:         string *
                     }
                 }
             }
 }
```

updateEntity

```
   ▾updateEntity {
     entity:      ▾ {
                    all of:
                      ▾Entity {
                        all of:
                           ▾ {
                             entity_key:  string *
                             country_id:  integer *
                             timezone_id: integer *
⇄                            currency_id: integer *
                             name:        string *
                             designation: string *
                             description: string *
                             url:         string *
                           }
                      }
                 }
     layout_key: string
   }
```

replyUpdatedEntity

```
   ▾replyUpdatedEntity {
     entity: ▾ {
               all of:
                 ▾Entity {
                   all of:
                      ▾ {
                        entity_key:  string *
                        country_id:  integer *
                        timezone_id: integer *
⇄                       currency_id: integer *
                        name:        string *
                        designation: string *
                        description: string *
                        url:         string *
                      }
                 }
             }
   }
```

replyDeleteEntity

```
   ▾replyDeleteEntity {
⇄    string: string
   }
```

entityPadDeleteReply

```
    ▾entityPadDeleteReply {
 ⇄     string: string
    }
```

entityPadErrorDefault

```
    ▾entityPadErrorDefault {
 ⇄     error: string *
    }
```

entityPadCreate

```
    ▾entityPadCreate {
      all of:
        ▾ {
 ⇄        cb_key: string *
          code:   string *
        }
    }
```

entityPadReply

```
    ▾entityPadReply {
      all of:
        ▾ {
          id:         integer
          cb_key:     string
          entity_id:  integer
 ⇄        cb_type_id: integer
          created_at: string
          updated_at: string
          deleted_at: string
        }
    }
```

entityGroupErrorDefault

```
    ▾entityGroupErrorDefault {
 ⇄     error: string *
    }
```

entityGroupCreate

```
    ▾entityGroupCreate {
      all of:
        ▾ {
          group_type_key:   string *
 ⇄        entity_group_key: string
          designation:      string *
          name:             string *
        }
    }
```

entityGroupUpdate

```
▾entityGroupUpdate {
  all of:
    ▾ {
       group_type_key:   string *
       entity_group_key: string
⇄      designation:      string *
       name:             string *
       position:         integer
    }
}
```

entityGroupReply

```
▾entityGroupReply {
  all of:
    ▾ {
       id:               integer
       entity_id:        integer
       entity_group_key: string
       group_type_id:    integer
       parent_group_id:  integer
⇄      designation:      string
       name:             string
       position:         integer
       created_at:       string
       updated_at:       string
       deleted_at:       string
    }
}
```

entityGroupDeleteReply

```
▾entityGroupDeleteReply {
⇄  string: string
}
```

entityModuleErrorDefault

```
▾entityModuleErrorDefault {
⇄  error: string *
}
```

entityModuleCreate

```
  ▾entityModuleCreate {
    all of:
      ▾ {
        entity_key: string *
        module_key: string *
        modules:     ▾[
                       ▾ {
⇄                        module_key:       string
                         module_type_keys: ▾[
                                             ▸ { }
                                           ]
                       }
                     ]
      }
  }
```

entityModuleReply

```
  ▾entityModuleReply {
    all of:
      ▾ {
        id:                integer
        entity_module_key: string
        entity_id:         integer
⇄       module_id:         integer
        created_at:        string
        updated_at:        string
        deleted_at:        string
      }
  }
```

forumErrorDefault

```
  ▾forumErrorDefault {
⇄    error: string
  }
```

forumCreate

```
  ▾forumCreate {
    all of:
      ▾ {
        cb_key:     string *
⇄       start_date: string *
        end_date:   string *
      }
  }
```

forumReply

```
    ▾forumReply {
      all of:
        ▾ {
           id:         integer
           cb_key:     string
           entity_id:  integer
           pb_id:      integer
           start_date: string
 ⇄         end_date:   string
           created_by: integer
           updated_by: integer
           created_at: string
           updated_at: string
           deleted_at: string
         }
    }
```

forumDeleteReply

```
    ▾forumDeleteReply {
 ⇄    string: string
    }
```

geographicAreaErrorDefault

```
    ▾geographicAreaErrorDefault {
 ⇄    error: string *
    }
```

geographicArea

```
    ▾geographicArea {
      all of:
        ▾ {
           geo_key:    string *
 ⇄         entity_id:  integer *
           name:       string *
           created_by: string *
         }
    }
```

replyDeleteGeographicArea

```
    ▾replyDeleteGeographicArea {
 ⇄    string: string
    }
```

groupTypeErrorDefault

```
    ▾groupTypeErrorDefault {
 ⇄    error: string
    }
```

groupTypeCreate

```
⇄     ▾groupTypeCreate {
        all of:
          ▾ {
            code: string *
          }
      }
```

groupTypeReply

```
⇄     ▾groupTypeReply {
        all of:
          ▾ {
            id:            integer
            groupType_key: string
            code:          string
            created_at:    string
            updated_at:    string
            deleted_at:    string
          }
      }
```

groupTypeDeleteReply

```
⇄     ▾groupTypeDeleteReply {
        string: string
      }
```

homePageConfigurationErrorDefault

```
⇄     ▾homePageConfigurationErrorDefault {
        error: string
      }
```

translations

```
⇄     ▾translations {
        all of:
          ▾ {
            language_code: string
            value:         string
          }
      }
```

homePageConfigurationUpdate

```
▾homePageConfigurationUpdate {
  all of:
    ▾ {
      value:              string *
      home_page_type_key: string *
      translations:        ▾[
                             ▾translations {
                               all of:
                                 ▾ {
                                   language_code: string
                                   value:         string
                                 }
                             }
                           ]
    }
}
```

homePageConfigurationReply

```
▾homePageConfigurationReply {
  all of:
    ▾ {
      id:                          integer
      home_page_configuration_key: string
      home_page_type_id:           integer
      site_id:                     integer
      group_name:                  string
      group_key:                   string
      value:                       string
      created_at:                  string
      updated_at:                  string
      deleted_at:                  string
    }
}
```

homePageConfigurationDeleteReply

```
▾homePageConfigurationDeleteReply {
  string: string
}
```

homePageTypeErrorDefault

```
▾homePageTypeErrorDefault {
  error: string
}
```

homePageTypeCreate

```
▾homePageTypeCreate {
  all of:
    ▾ {
      name:       string *
⇄     code:       string *
      type_code:  string *
      parent_key: string
    }
}
```

homePageTypeShowReply

```
▾homePageTypeShowReply {
  all of:
    ▾ {
      id:                 integer
      home_page_type_key: string
      entity_id:          integer
      name:               string
      code:               string
      parent_id:          integer
      type_code:          string
      parent:             ▾ {
                            all of:
                              ▾homePageTypeReply {
                                all of:
                                  ▾ {
                                    id:                 integer
                                    home_page_type_key: string
                                    entity_id:          integer
                                    name:               string
                                    code:               string
                                    parent_id:          integer
                                    type_code:          string
                                    created_at:         string
                                    updated_at:         string
⇄                                   deleted_at:         string
                                  }
                              }
                          }
      childs:             ▾[
                            ▾homePageTypeReply {
                              all of:
                                ▾ {
                                  id:                 integer
                                  home_page_type_key: string
                                  entity_id:          integer
                                  name:               string
```

```
                                        code:               string
                                        parent_id:          integer
                                        type_code:          string
                                        created_at:         string
                                        updated_at:         string
                                        deleted_at:         string
                                     }
                                 }
                              ]
          created_at:         string
          updated_at:         string
          deleted_at:         string
        }
    }
```

homePageTypeReply

```
    ▾homePageTypeReply {
      all of:
        ▾ {
          id:                 integer
          home_page_type_key: string
          entity_id:          integer
          name:               string
 ⇄        code:               string
          parent_id:          integer
          type_code:          string
          created_at:         string
          updated_at:         string
          deleted_at:         string
        }
    }
```

homePageTypeDeleteReply

```
    ▾homePageTypeDeleteReply {
 ⇄    string: string
    }
```

ideaErrorDefault

```
    ▾ideaErrorDefault {
 ⇄    error: string
    }
```

ideaCreate

```
   ▾ideaCreate {
     all of:
        ▾ {
⇄          cb_key: string *
        }
   }
```

ideaReply

```
   ▾ideaReply {
     all of:
        ▾ {
           id:         integer
           cb_key:     string
           entity_id:  integer
⇄          start_date: string
           end_date:   string
           created_at: string
           updated_at: string
           deleted_at: string
        }
   }
```

ideaDeleteReply

```
   ▾ideaDeleteReply {
⇄    string: string
   }
```

kioskTypeErrorDefault

```
   ▾kioskTypeErrorDefault {
⇄    error: string
   }
```

kioskTypeReply

```
   ▾kioskTypeReply {
     all of:
        ▾ {
           id:         integer
           title:      string
⇄          created_at: string
           updated_at: string
           deleted_at: string
        }
   }
```

kioskErrorDefault

```
   ▾kioskErrorDefault {
⇄    error: string
   }
```

kioskCreate

```
▾kioskCreate {
  all of:
    ▾ {
      cb_key:        string
⇄     title:         string *
      kiosk_type_id: string *
      event_key:     string *
    }
}
```

kioskReply

```
▾kioskReply {
  all of:
    ▾ {
      id:            integer
      kiosk_key:     string
      entity_id:     integer
      entity_cb_id:  string
      event_key:     string
      kiosk_type_id: integer
      title:         string
      created_at:    string
      updated_at:    string
      entity_cb:     ▾[
⇄                      ▾entityPadReply {
                         all of:
                           ▾ {
                             id:         integer
                             cb_key:     string
                             entity_id:  integer
                             cb_type_id: integer
                             created_at: string
                             updated_at: string
                             deleted_at: string
                           }
                       }
                     ]
    }
}
```

kioskDeleteReply

```
▾kioskDeleteReply {
⇄  string: string
}
```

languageErrorDefault

```
    ▾ languageErrorDefault {
⇄     error: string *
    }
```

Language

```
    ▾ Language {
      all of:
        ▾ {
⇄         name: string *
          code: integer *
        }
    }
```

replyLanguage

```
    ▾ replyLanguage {
      data: ▾ {
              all of:
                ▾ Language {
                  all of:
                    ▾ {
⇄                     name: string *
                      code: integer *
                    }
                }
            }
    }
```

createLanguage

```
    ▾ createLanguage {
      language: ▾ {
                  all of:
                    ▾ Language {
                      all of:
                        ▾ {
⇄                         name: string *
                          code: integer *
                        }
                    }
                }
    }
```

replyCreateLanguage

```
   ▾replyCreateLanguage {
      language: ▾ {
                  all of:
                     ▾Language {
                        all of:
                           ▾ {
⇄                             name: string *
                              code: integer *
                           }
                     }
                }
   }
```

updateLanguage

```
   ▾updateLanguage {
      language: ▾ {
                  all of:
                     ▾Language {
                        all of:
                           ▾ {
⇄                             name: string *
                              code: integer *
                           }
                     }
                }
   }
```

replyUpdatedLanguage

```
   ▾replyUpdatedLanguage {
      language: ▾ {
                  all of:
                     ▾Language {
                        all of:
                           ▾ {
⇄                             name: string *
                              code: integer *
                           }
                     }
                }
   }
```

replyDeleteLanguage

```
   ▾replyDeleteLanguage {
⇄     string: string
   }
```

layoutErrorDefault

```
   ▾layoutErrorDefault {
⇄     error: string
   }
```

layoutCreate

```
    ▾layoutCreate {
      all of:
        ▾ {
⇄         name:      string *
          reference: string *
        }
    }
```

layoutReply

```
    ▾layoutReply {
      all of:
        ▾ {
          id:         integer
          layout_key: string
          name:       string
⇄         reference:  string
          created_at: string
          updated_at: string
          deleted_at: string
        }
    }
```

layoutDeleteReply

```
    ▾layoutDeleteReply {
⇄     string: string
    }
```

levelErrorDefault

```
    ▾levelErrorDefault {
⇄     error: string *
    }
```

Level Parameter

```
    ▾Level Parameter {
      all of:
        ▾ {
          id:                  integer *
          level_parameter_key: string *
          mandatory:           boolean *
          manual_verification: boolean *
⇄         name:                string *
          position:            integer *
          created_at:          string
          updated_at:          string
          deleted_at:          string
        }
    }
```

replyLevelList

```
▾replyLevelList {
   data: ▾ {
            all of:
              ▾[
                 ▾Level Parameter {
                    all of:
                       ▾ {
                          id:                   integer *
                          level_parameter_key: string *
                          mandatory:            boolean *
⇄                         manual_verification: boolean *
                          name:                 string *
                          position:             integer *
                          created_at:           string
                          updated_at:           string
                          deleted_at:           string
                       }
                 }
              ]
         }
}
```

replyShowLevel

```
▾replyShowLevel {
   data: ▾ {
            all of:
              ▾Level Parameter {
                 all of:
                    ▾ {
                       id:                   integer *
                       level_parameter_key: string *
                       mandatory:            boolean *
⇄                      manual_verification: boolean *
                       name:                 string *
                       position:             integer *
                       created_at:           string
                       updated_at:           string
                       deleted_at:           string
                    }
              }
         }
}
```

createLevel

```
    ▾createLevel {
      name:                 string *
⇄     mandatory:            boolean *
      manual_verification:  boolean *
    }
```

replyCreateLevel

```
    ▾replyCreateLevel {
      level: ▾ {
              all of:
                ▾Level Parameter {
                  all of:
                    ▾ {
                      id:                   integer *
                      level_parameter_key:  string *
                      mandatory:            boolean *
⇄                     manual_verification:  boolean *
                      name:                 string *
                      position:             integer *
                      created_at:           string
                      updated_at:           string
                      deleted_at:           string
                    }
                }
            }
    }
```

updateLevel

```
    ▾updateLevel {
      name:                 string *
⇄     mandatory:            boolean *
      manual_verification:  boolean *
    }
```

replyUpdatedLevel

```
    ▾replyUpdatedLevel {
      level: ▾ {
               all of:
                 ▾Level Parameter {
                   all of:
                     ▾ {
                       id:                   integer *
                       level_parameter_key: string *
                       mandatory:            boolean *
⇄                      manual_verification: boolean *
                       name:                 string *
                       position:             integer *
                       created_at:           string
                       updated_at:           string
                       deleted_at:           string
                     }
                 }
             }
    }
```

replyDeleteLevel

```
    ▾replyDeleteLevel {
⇄     string: string
    }
```

moduleTypeErrorDefault

```
    ▾moduleTypeErrorDefault {
⇄     error: string
    }
```

moduleType

```
    ▾moduleType {
      all of:
        ▾ {
⇄         name: string *
          code: string *
        }
    }
```

moduleTypeUpdate

```
    ▾moduleTypeUpdate {
      all of:
        ▾ {
⇄         name: string *
          code: string *
        }
    }
```

moduleTypeCreate

```
  ▾moduleTypeCreate {
    all of:
      ▾ {
        module_key:   string *
        module_types: ▾[
                        ▾moduleType {
                          all of:
⇄                           ▾ {
                              name: string *
                              code: string *
                            }
                        }
                      ]
      }
  }
```

moduleTypeShowReply

```
  ▾moduleTypeShowReply {
    all of:
      ▾ {
        id:              integer
        module_type_key: string
        module_id:       integer
        name:            string
        code:            string
        module:          ▾ {
                           all of:
                             ▾moduleReply {
                               all of:
                                 ▾ {
                                   id:         integer
⇄                                  module_key: string
                                   name:       string
                                   code:       string
                                   token:      string
                                   created_at: string
                                   updated_at: string
                                   deleted_at: string
                                 }
                             }
                         }
        created_at:      string
        updated_at:      string
        deleted_at:      string
      }
  }
```

moduleTypeReply

```
   ▾moduleTypeReply {
     all of:
       ▾ {
         id:              integer
         module_type_key: string
         module_id:       integer
⇄        name:            string
         code:            string
         created_at:      string
         updated_at:      string
         deleted_at:      string
       }
   }
```

moduleTypeDeleteReply

```
   ▾moduleTypeDeleteReply {
⇄    string: string
   }
```

moduleErrorDefault

```
   ▾moduleErrorDefault {
⇄    error: string
   }
```

moduleCreate

```
   ▾moduleCreate {
     all of:
       ▾ {
⇄        name: string *
         code: string *
       }
   }
```

moduleReply

```
   ▾moduleReply {
     all of:
       ▾ {
         id:         integer
         module_key: string
         name:       string
⇄        code:       string
         token:      string
         created_at: string
         updated_at: string
         deleted_at: string
       }
   }
```

moduleDeleteReply

```
    ▾moduleDeleteReply {
⇄     string: string
    }
```

pageErrorDefault

```
    ▾pageErrorDefault {
⇄     error: string
    }
```

pageCreate

```
    ▾pageCreate {
      all of:
        ▾ {
⇄           page_key: string *
            type:     string *
          }
    }
```

pageUpdate

```
    ▾pageUpdate {
      all of:
        ▾ {
⇄           entity_id: integer *
            type:      string *
          }
    }
```

pageReply

```
    ▾pageReply {
      all of:
        ▾ {
            id:         integer
            page_key:   string
            entity_id:  integer
⇄           type:       string
            created_at: string
            updated_at: string
            deleted_at: string
          }
    }
```

pageDeleteReply

```
    ▾pageDeleteReply {
⇄     string: string
    }
```

parameterTypeErrorDefault

```
 ▾parameterTypeErrorDefault {
⇄   error: string
  }
```

parameterTypeCreate

```
 ▾parameterTypeCreate {
   all of:
     ▾ {
         name:           string *
⇄        code:           string *
         user_parameter: boolean
       }
  }
```

parameterTypeReply

```
 ▾parameterTypeReply {
   all of:
     ▾ {
         id:             integer
         name:           string
         code:           string
⇄        user_parameter: boolean
         created_at:     string
         updated_at:     string
         deleted_at:     string
       }
  }
```

parameterTypeDeleteReply

```
 ▾parameterTypeDeleteReply {
⇄   string: string
  }
```

parameterUserOptionErrorDefault

```
 ▾parameterUserOptionErrorDefault {
⇄   error: string
  }
```

parameterUserTranslation

```
 ▾parameterUserTranslation {
    language_code: string
⇄   name:          string
  }
```

parameterUserOptionCreate

```
    ▾parameterUserOptionCreate {
      all of:
        ▾ {
          parameter_user_type_key: string *
          translations:           ▾[
⇄                                   ▾parameterUserTranslation {
                                      language_code: string
                                      name:          string
                                    }
                                  ]
        }
    }
```

parameterUserOptionShowReply

```
    ▾parameterUserOptionShowReply {
      all of:
        ▾ {
          id:                         integer
          parameter_user_type_id:     integer
          parameter_user_option_key: string
⇄         name:                       string
          created_at:                 string
          updated_at:                 string
          deleted_at:                 string
        }
    }
```

parameterUserOptionReply

```
    ▾parameterUserOptionReply {
      all of:
        ▾ {
          id:                         integer
          parameter_user_type_id:     integer
⇄         parameter_user_option_key: string
          created_at:                 string
          updated_at:                 string
          deleted_at:                 string
        }
    }
```

parameterUserOptionDeleteReply

```
    ▾parameterUserOptionDeleteReply {
⇄     string: string
    }
```

parameterUserTypeErrorDefault

```
▾parameterUserTypeErrorDefault {
   error: string
}
```

parameterUserTypeCreate

```
▾parameterUserTypeCreate {
  all of:
    ▾ {
      parameter_type_code:    string *
      mandatory:              boolean *
      translations:          ▾[
                               ▾parameterUserTranslation {
                                 language_code: string
                                 name:          string
                               }
                             ]
      parameter_user_options: ▾[
                               ▾parameterUserOptionCreate {
                                 all of:
                                   ▾ {
                                     parameter_user_type_key: string *
                                     translations:            ▸[]
                                   }
                               }
                             ]
    }
}
```

parameterUserTypeShowReply

```
  ▾parameterUserTypeShowReply {
    all of:
      ▾ {
        id:                      integer
        parameter_user_type_id:  integer
        parameter_user_type_key: string
        name:                    string
        parameter_user_options:  ▾[
                                   ▾parameterUserOptionShowReply {
                                     all of:
                                       ▾ {
                                         id:                        integer
                                         parameter_user_type_id:    integer
⇄                                        parameter_user_option_key: string
                                         name:                      string
                                         created_at:                string
                                         updated_at:                string
                                         deleted_at:                string
                                       }
                                   }
                                 ]
        created_at:              string
        updated_at:              string
        deleted_at:              string
      }
  }
```

parameterUserTypeReply

```
  ▾parameterUserTypeReply {
    all of:
      ▾ {
        id:                      integer
        parameter_user_type_key: string
        parameter_type_id:       integer
        entity_id:               integer
⇄       mandatory:               boolean
        level_parameter_id:      integer
        created_at:              string
        updated_at:              string
        deleted_at:              string
      }
  }
```

parameterUserTypeDeleteReply

```
  ▾parameterUserTypeDeleteReply {
⇄    string: string
  }
```

parameterErrorDefault

```
   ▾parameterErrorDefault {
⇄     error: string
   }
```

parameterCreate

```
   ▾parameterCreate {
     all of:
       ▾ {
         parameter_type_id: integer *
⇄        parameter:         string *
         description:       string *
         code:              string *
       }
   }
```

parameterReply

```
   ▾parameterReply {
     all of:
       ▾ {
         id:                integer
         parameter_type_id: integer
         entity_id:         integer
         parameter:         string
⇄        description:       string
         code:              string
         created_at:        string
         updated_at:        string
         deleted_at:        string
       }
   }
```

parameterDeleteReply

```
   ▾parameterDeleteReply {
⇄     string: string
   }
```

permissionErrorDefault

```
   ▾permissionErrorDefault {
⇄     error: string
   }
```

permissionCreate

```
  ▾permissionCreate {
    all of:
      ▾ {
        role_key: string
        code:     string *
        module:   string *
⇄       api:      string *
        create:   boolean *
        view:     boolean *
        update:   boolean *
        delete:   boolean *
      }
  }
```

permissionReply

```
  ▾permissionReply {
    all of:
      ▾ {
        id:         integer
        role_id:    integer
        code:       string
        module:     string
        api:        string
⇄       create:     boolean
        view:       boolean
        update:     boolean
        delete:     boolean
        created_at: string
        updated_at: string
        deleted_at: string
      }
  }
```

permissionDeleteReply

```
  ▾permissionDeleteReply {
⇄   string: string
  }
```

proposalErrorDefault

```
  ▾proposalErrorDefault {
⇄   error: string
  }
```

proposalCreate

```
▾proposalCreate {
  all of:
    ▾ {
      role_key: string
      code:     string *
      module:   string *
      api:      string *
      create:   boolean *
      view:     boolean *
      update:   boolean *
      delete:   boolean *
    }
}
```

proposalReply

```
▾proposalReply {
  all of:
    ▾ {
      id:           integer
      proposal_key: string
      cb_key:       string
      entity_id:    integer
      created_at:   string
      updated_at:   string
      deleted_at:   string
    }
}
```

proposalDeleteReply

```
▾proposalDeleteReply {
  string: string
}
```

roleErrorDefault

```
▾roleErrorDefault {
  error: string *
}
```

role

```
▾role {
  all of:
    ▾ {
      entity_id:   integer *
      role_key:    string *
      name:        string *
      code:        string *
      description: string *
    }
}
```

replyDeleteRole

```
▾replyDeleteRole {
  string: string
}
```

siteConfigurationErrorDefault

```
▾siteConfigurationErrorDefault {
  error: string
}
```

siteConfigurationTranslations

```
▾siteConfigurationTranslations {
  all of:
    ▾ {
      name:          string *
      language_code: string *
      description:   string
    }
}
```

siteConfigurationRequest

```
▾siteConfigurationRequest {
  all of:
    ▾ {
      code:          string *
      group:         integer *
      translations: ▾[
                      ▾siteConfigurationTranslations {
                        all of:
                          ▾ {
                            name:          string *
                            language_code: string *
                            description:   string
                          }
                      }
                    ]
    }
}
```

siteConfigurationCreate

```
   ▾siteConfigurationCreate {
     all of:
        ▾ {
          request: ▾[
                      ▾siteConfigurationRequest {
                        all of:
                           ▾ {
⇄                            code:         string *
                             group:        integer *
                             translations: ▸[]
                           }
                      }
                   ]
        }
   }
```

siteConfigurationReply

```
   ▾siteConfigurationReply {
     all of:
        ▾ {
          id:                 integer
          site_conf_key:      string
          code:               string
⇄         site_conf_group_id: integer
          created_at:         string
          updated_at:         string
          deleted_at:         string
        }
   }
```

siteConfigurationDeleteReply

```
   ▾siteConfigurationDeleteReply {
⇄     string: string
   }
```

siteConfigurationGroupErrorDefault

```
   ▾siteConfigurationGroupErrorDefault {
⇄     error: string
   }
```

siteConfigurationGroupTranslations

```
  ▾siteConfigurationGroupTranslations {
    all of:
      ▾ {
        name:          string *
⇄       language_code: string *
        description:   string
      }
  }
```

siteConfigurationGroupRequest

```
  ▾siteConfigurationGroupRequest {
    all of:
      ▾ {
        code:         string *
        group:        integer
        translations: ▾[
                        ▾siteConfigurationGroupTranslations {
                          all of:
⇄                           ▾ {
                              name:          string *
                              language_code: string *
                              description:   string
                            }
                        }
                      ]
      }
  }
```

siteConfigurationGroupCreate

```
  ▾siteConfigurationGroupCreate {
    all of:
      ▾ {
        request: ▾[
                   ▾siteConfigurationGroupRequest {
                     all of:
                       ▾ {
⇄                        code:         string *
                         group:        integer
                         translations: ▸[]
                       }
                   }
                 ]
      }
  }
```

siteConfigurationGroupReply

```
  ▾siteConfigurationGroupReply {
    all of:
      ▾ {
        id:                integer
        site_conf_group_key: string
⇄       code:              string
        created_at:        string
        updated_at:        string
        deleted_at:        string
      }
  }
```

siteConfigurationGroupDeleteReply

```
  ▾siteConfigurationGroupDeleteReply {
⇄   string: string
  }
```

siteErrorDefault

```
  ▾siteErrorDefault {
⇄   error: string *
  }
```

use_terms

```
  ▾use_terms {
    all of:
      ▾ {
        site_id:       integer
⇄       language_code: string *
        content:       string *
      }
  }
```

site

```
   ▾site {
    all of:
      ▾ {
        cm_key:         string
        site_key:       string
        entity_id:      string
        name:           string
        description:    string
⇄       link:           string
        partial_link:   boolean
        active:         boolean
        no_reply_email: string
        start_date:     string
        end_date:       string
      }
   }
```

siteCreateReply

```
  ▾siteCreateReply {
    all of:
      ▾ {
          cm_key:         string
          site_key:       string
          entity_id:      string
          name:           string
          description:    string
          link:           string
          partial_link:   boolean
          active:         boolean
          no_reply_email: string
          start_date:     string
          end_date:       string
          use_terms:      ▾[
                            ▾use_terms {
                              all of:
                                ▾ {
                                    site_id:       integer
⇄                                   language_code: string *
                                    content:       string *
                                  }
                            }
                          ]
          layout:         ▾[
                            ▾layoutReply {
                              all of:
                                ▾ {
                                    id:         integer
                                    layout_key: string
                                    name:       string
                                    reference:  string
                                    created_at: string
                                    updated_at: string
                                    deleted_at: string
                                  }
                            }
                          ]
      }
  }
```

replyDeleteSite

```
  ▾replyDeleteSite {
⇄    string: string
  }
```

statusLoginErrorDefault

```
▾statusLoginErrorDefault {
⇄   error: string
 }
```

statusLoginCreate

```
▾statusLoginCreate {
   all of:
     ▾ {
⇄      code: string *
     }
 }
```

statusLoginReply

```
▾statusLoginReply {
   all of:
     ▾ {
       id:         integer
       code:       string
⇄      created_at: string
       updated_at: string
       deleted_at: string
     }
 }
```

timezoneErrorDefault

```
▾timezoneErrorDefault {
⇄   error: string *
 }
```

Timezone

```
▾Timezone {
   all of:
     ▾ {
⇄      country_code: string *
       name:         string *
     }
 }
```

replyTimezone

```
▾replyTimezone {
   data: ▾ {
            all of:
               ▾Timezone {
⇄                all of:
                   ▾ {
                      country_code: string *
                      name:         string *
                   }
               }
         }
}
```

createTimezone

```
▾createTimezone {
   timezone: ▾ {
                all of:
                   ▾Timezone {
⇄                    all of:
                       ▾ {
                          country_code: string *
                          name:         string *
                       }
                   }
             }
}
```

replyCreateTimezone

```
▾replyCreateTimezone {
   timezone: ▾ {
                all of:
                   ▾Timezone {
⇄                    all of:
                       ▾ {
                          country_code: string *
                          name:         string *
                       }
                   }
             }
}
```

updateTimezone

```
  ▾updateTimezone {
    timezone: ▾ {
                all of:
                  ▾Timezone {
                    all of:
                      ▾ {
⇄                       country_code: string *
                        name:         string *
                      }
                  }
              }
  }
```

replyUpdatedTimezone

```
  ▾replyUpdatedTimezone {
    timezone: ▾ {
                all of:
                  ▾Timezone {
                    all of:
                      ▾ {
⇄                       country_code: string *
                        name:         string *
                      }
                  }
              }
  }
```

replyDeleteTimezone

```
  ▾replyDeleteTimezone {
⇄   string: string
  }
```

userErrorDefault

```
  ▾userErrorDefault {
⇄   error: string *
  }
```

user

```
  ▾user {
    all of:
      ▾ {
        user_key:           string *
⇄       admin:              boolean *
        geographic_area_id: string *
      }
  }
```

replyDeleteUser

```
   ▾replyDeleteUser {
⇄     string: string
   }
```

Questionnaire Module

Version 1.0.0

Contact information
pvalente@onesource.pt (mailto:pvalente@onesource.pt)

Paths

`/esParticipant/{key}`

`GET /esParticipant/{key}`

EsParticipant Method

Summary

Shows a EsParticipant Method

Parameters

Name	Located in	Description	Required	Schema
key	path	EsParticipant Method Key	Yes	⇄ `string`
X-MODULE-TOKEN	header	Module Token	Yes	⇄ `string`

Responses

Code	Description	Schema
200	Shows the Participant data	⇄
401	Unauthorized	⇄
404	Participant not Found	⇄
500	Failed to retrieve Participant	⇄

200

```
▾esParticipantsReply {
  all of:
    ▾ {
      id:                integer
      key:               string
      event_schedule_id: integer
      name:              string
      user_key:          string
      created_by:        string
      updated_by:        string
      created_at:        string
      updated_at:        string
    }
}
```

401

```
▾esParticipantsMethodErrorDefault {
  error: string *
}
```

404

```
▾esParticipantsMethodErrorDefault {
  error: string *
}
```

500

```
▾esParticipantsMethodErrorDefault {
  error: string *
}
```

Try this operation

PUT /esParticipant/{key}

EsParticipant Method

Summary

Update a EsParticipant Method

Parameters

Name	Located in	Description	Required	Schema
EsParticipant	body	EsParticipant Method data	Yes	⇄

```
▾esParticipantsUpdate {
  all of:
    ▾ {
        participant_id: integer *
        name:           string *
        periods:        ▾[
                           integer
                        ]
        questions:      ▾[
                           integer
                        ]
      }
}
```

key	path	Event Schedule Key	Yes	⇄ string
X-AUTH-TOKEN	header	User Auth Token	Yes	⇄ string
X-MODULE-TOKEN	header	Module Token	Yes	⇄ string

Responses

Code	Description	Schema
200	The Updated EsParticipant	⇄

```
▾esParticipantsReply {
  all of:
    ▾ {
        id:                 integer
        key:                string
        event_schedule_id:  integer
        name:               string
        user_key:           string
        created_by:         string
        updated_by:         string
        created_at:         string
        updated_at:         string
      }
}
```

401	Unauthorized	⇄

```
▾esParticipantsMethodErrorDefault {
  error: string *
}
```

404	Participant not Found	⇄

```
▾esParticipantsMethodErrorDefault {
  error: string *
}
```

Code	Description	Schema
500	This is already closed	⇄

```
▾esParticipantsMethodErrorDefault {
   error: string *
}
```

Try this operation

POST /esParticipant/{key}

EsParticipant Method

Summary

Create a EsParticipant Method

Parameters

Name	Located in	Description	Required	Schema
key	path	Event Schedule Key	Yes	⇄ string
EsParticipant	body	EsParticipant Method data	Yes	⇄

```
▾esParticipantsCreate {
  all of:
    ▾ {
       name:      string *
       periods:  ▾[
                   integer
                  ]
       questions: ▾[
                   integer
                  ]
    }
}
```

X-AUTH-TOKEN	header	User Auth Token	Yes	⇄ string
X-MODULE-TOKEN	header	Module Token	Yes	⇄ string

Responses

Code	Description	Schema
201	The newly created EsParticipant	⇄

```
▾esParticipantsReply {
  all of:
    ▾ {
       id:                integer
       key:               string
       event_schedule_id: integer
       name:              string
       user_key:          string
       created_by:        string
       updated_by:        string
       created_at:        string
       updated_at:        string
    }
}
```

401	Unauthorized	⇄

```
▾esParticipantsMethodErrorDefault
{
   error: string *
}
```

500	Error in storing attendance | Failed to store participation	⇄

```
▾esParticipantsMethodErrorDefault
{
   error: string *
}
```

Try this operation

`DELETE /esParticipant/{key}`

EsParticipant Method

Summary

Delete EsParticipant Method

Parameters

Name	Located in	Description	Required	Schema
key	path	EsParticipant Key	Yes	⇄ `string`
X-MODULE-TOKEN	header	Module Token	Yes	⇄ `string`
X-AUTH-TOKEN	header	User Auth Token	Yes	⇄ `string`

Responses

Code	Description		Schema

200	OK	⇄	▾**`esParticipantDeleteReply {`**
			`string: string`
			`}`

401	Unauthorized	⇄	▾**`esParticipantsMethodErrorDefault {`**
			`error: string *`
			`}`

404	Participant not Found	⇄	▾**`esParticipantsMethodErrorDefault {`**
			`error: string *`
			`}`

500	Failed to delete a Participant	⇄	▾**`esParticipantsMethodErrorDefault {`**
			`error: string *`
			`}`

Try this operation

`/eventSchedule/{event_schedule_key}`

GET /eventSchedule/{event_schedule_key}

Event Schedule Method

Summary

Shows a Event Schedule Method

Parameters

Name	Located in	Description	Required	Schema
event_schedule_key	path	Event Schedule Method Key	Yes	⇄ string
X-MODULE-TOKEN	header	Module Token	Yes	⇄ string

Responses

Code	Description	Schema
200	Shows the Event Schedule data	⇄
401	Unauthorized	⇄
404	Event Schedule not Found	⇄
500	Failed to retrieve the Event Schedule	⇄

200

```
▾eventSchedulesReply {
  all of:
    ▾ {
        id:             integer
        key:            string
        entity_id:      integer
        type_id:        integer
        title:          string
        description:    string
        local:          string
        closed:         integer
        public:         integer
        created_by:     string
        updated_by:     string
        es_period_id:   integer
        es_question_id: integer
        created_at:     string
        updated_at:     string
    }
}
```

401

```
▾eventSchedulesMethodErrorDefault {
   error: string *
}
```

404

```
▾eventSchedulesMethodErrorDefault {
   error: string *
}
```

500

```
▾eventSchedulesMethodErrorDefault {
   error: string *
}
```

Try this operation

PUT /eventSchedule/{event_schedule_key}

Event Schedule Method

Summary

Update a Event Schedule Method

Parameters

Name	Located in	Description	Required	Schema
Event Schedule	body	Event Schedule Method data	Yes	⇄
event_schedule_key	path	Event Schedule Key	Yes	⇄ string
X-AUTH-TOKEN	header	User Auth Token	Yes	⇄ string
X-MODULE-TOKEN	header	Module Token	Yes	⇄ string

```
▾eventSchedulesUpdate {
  all of:
    ▾ {
      entity_id:      integer
                      *
      title:          string
                      *
      type_id:        integer
                      *
      description:    string
                      *
      local:          string
                      *
      closed:         integer
                      *
      public:         integer
                      *
      periods:        ▸[]
      questions:      ▸[]
      es_period_id:   integer
                      *
      es_question_id: integer
                      *
    }
}
```

Responses

Code	Description	Schema

200 The Updated Event Schedule ⇄

```
▾eventSchedulesReply {
  all of:
    ▾ {
        id:             integer
        key:            string
        entity_id:      integer
        type_id:        integer
        title:          string
        description:    string
        local:          string
        closed:         integer
        public:         integer
        created_by:     string
        updated_by:     string
        es_period_id:   integer
        es_question_id: integer
        created_at:     string
        updated_at:     string
    }
}
```

401 Unauthorized ⇄

```
▾eventSchedulesMethodErrorDefault {
   error: string *
}
```

404 Event Schedule not Found ⇄

```
▾eventSchedulesMethodErrorDefault {
   error: string *
}
```

500 Failed to Update Event Schedule ⇄

```
▾eventSchedulesMethodErrorDefault {
   error: string *
}
```

Try this operation

`DELETE /eventSchedule/{event_schedule_key}`

Event Schedule Method

Summary

Delete Event Schedule Method

Parameters

Name	Located in	Description	Required	Schema
event_schedule_key	path	Event Schedule Key	Yes	⇄ `string`
X-MODULE-TOKEN	header	Module Token	Yes	⇄ `string`
X-AUTH-TOKEN	header	User Auth Token	Yes	⇄ `string`

Responses

Code	Description	Schema

200	OK	⇄ ▾**eventScheduleDeleteReply {**
		`string: string`
		}

401	Unauthorized	⇄ ▾**eventSchedulesMethodErrorDefault {**
		`error: string *`
		}

404	Event Schedule not Found	⇄ ▾**eventSchedulesMethodErrorDefault {**
		`error: string *`
		}

500	Failed to delete a Event Schedule	⇄ ▾**eventSchedulesMethodErrorDefault {**
		`error: string *`
		}

Try this operation

/eventSchedule

POST /eventSchedule

Event Schedule Method

Summary

Create a Event Schedule Method

Parameters

Name	Located in	Description	Required	Schema
Event Schedule	body	Event Schedule Method data	Yes	⇄
X-AUTH-TOKEN	header	User Auth Token	Yes	⇄ string
X-MODULE-TOKEN	header	Module Token	Yes	⇄ string

```
▾eventSchedulesCreate {
  all of:
    ▾ {
      entity_id:   integer
                   *
      title:       string
                   *
      type_id:     integer
                   *
      description: string
                   *
      local:       string
                   *
      closed:      integer
                   *
      public:      integer
                   *
      periods:     ▸[]
      questions:   ▸[]
    }
}
```

Responses

Code	Description	Schema

201 The newly created Event Schedule ⇄

```
▾eventSchedulesReply {
  all of:
    ▾ {
      id:             integer
      key:            string
      entity_id:      integer
      type_id:        integer
      title:          string
      description:    string
      local:          string
      closed:         integer
      public:         integer
      created_by:     string
      updated_by:     string
      es_period_id:   integer
      es_question_id: integer
      created_at:     string
      updated_at:     string
    }
}
```

401 Unauthorized ⇄

```
▾eventSchedulesMethodErrorDefault {
  error: string *
}
```

500 Failed to store new Event Schedule ⇄

```
▾eventSchedulesMethodErrorDefault {
  error: string *
}
```

Try this operation

/formConfiguration/{form_configuration_key}

GET /formConfiguration/{form_configuration_key}

Form Configuration Method

Summary

Shows a Form Configuration Method

Parameters

Name	Located in	Description	Required	Schema
form_configuration_key	path	Form Configuration Method Key	Yes	⇄ string
LANG-CODE	header	Module Token	No	⇄ string
LANG-CODE-DEFAULT	header	Module Token	Yes	⇄ string
X-MODULE-TOKEN	header	Module Token	Yes	⇄ string

Responses

Code	Description		Schema
200	Shows the Form Configuration data	⇄	▾**formConfigurationsReply {** **all of:** ▾ **{** id: integer form_configuration_key: string code: string updated_at: string } }

401	Unauthorized	⇄	▾**formConfigurationsMethodErrorDefault {**
			error: string *
			}

404	Form Configuration not Found	⇄	▾**formConfigurationsMethodErrorDefault {**
			error: string *
			}

500	Failed to retrieve the Form Configuration	⇄	▾**formConfigurationsMethodErrorDefault {**
			error: string *
			}

Try this operation

/formConfigurations

`POST /formConfigurations`

Form Configuration Method

Summary

Create a Form Configuration Method

Parameters

Name	Located in	Description	Required	Schema
Form Configuration	body	Form Configuration Method data	Yes	⇄
X-AUTH-TOKEN	header	User Auth Token	Yes	⇄ `string`
X-MODULE-TOKEN	header	Module Token	Yes	⇄ `string`

Form Configuration:

```
▾formConfigurationsCreate {
  all of:
    ▾ {
        code:           string
                        *
        translations:  ▸[]
    }
}
```

Responses

Code	Description	Schema
201	The newly created Form Configuration	⇄
401	Unauthorized	⇄
500	Failed to store new Form Configuration	⇄

201

```
▾formConfigurationsReply {
  all of:
    ▾ {
        id:                      integer
        form_configuration_key:  string
        code:                    string
        updated_at:              string
    }
}
```

401

```
▾formConfigurationsMethodErrorDefault
{
   error: string *
}
```

500

```
▾formConfigurationsMethodErrorDefault
{
   error: string *
}
```

Try this operation

`/formConfigurations/{form_configuration_key}`

`PUT /formConfigurations/{form_configuration_key}`

Form Configuration Method

Summary

Update a Form Configuration Method

Parameters

Name	Located in	Description	Required	Schema
Form Configuration	body	Form Configuration Method data	Yes	⇄ ▸ **`formConfigurationsCreate { }`**
form_configuration_key	path	Form Configuration Key	Yes	⇄ `string`
X-AUTH-TOKEN	header	User Auth Token	Yes	⇄ `string`
X-MODULE-TOKEN	header	Module Token	Yes	⇄ `string`

Responses

Code **Description** **Schema**

200 The Updated Form Configuration ⇄

```
▾formConfigurationsReply {
  all of:
    ▾ {
       id:                    integer
       form_configuration_key: string
       code:                  string
       updated_at:            string
     }
}
```

401 Unauthorized ⇄

```
▾formConfigurationsMethodErrorDefault
{
   error: string *
}
```

404 Form Configuration not Found ⇄

```
▾formConfigurationsMethodErrorDefault
{
   error: string *
}
```

500 Failed to Update Form Configuration ⇄

```
▾formConfigurationsMethodErrorDefault
{
   error: string *
}
```

Try this operation

`DELETE /formConfigurations/{form_configuration_key}`

Form Configuration Method

Summary

Delete Form Configuration Method

Parameters

Name	Located in	Description	Required	Schema
form_configuration_key	path	Form Configuration Key	Yes	⇄ `string`
X-MODULE-TOKEN	header	Module Token	Yes	⇄ `string`
X-AUTH-TOKEN	header	User Auth Token	Yes	⇄ `string`

Responses

Code	Description	Schema

200	OK	⇄ ▾ **`formConfigurationDeleteReply {`**
		`string: string`
		`}`

401	Unauthorized	⇄ ▾ **`formConfigurationsMethodErrorDefault {`**
		`error: string *`
		`}`

404	Form Configuration not Found	⇄ ▾ **`formConfigurationsMethodErrorDefault {`**
		`error: string *`
		`}`

500	Failed to delete a Form Configuration	⇄ ▾ **`formConfigurationsMethodErrorDefault {`**
		`error: string *`
		`}`

Try this operation

`/formReply/{form_reply_key}`

`GET /formReply/{form_reply_key}`

Form Reply Method

Summary

Shows a Form Reply Method

Parameters

Name	Located in	Description	Required	Schema
form_reply_key	path	Form Reply Method Key	Yes	⇄ `string`
X-MODULE-TOKEN	header	Module Token	Yes	⇄ `string`

Responses

Code Description Schema

200 Shows the Form Reply data ⇄

```
▾formRepliesReply {
  all of:
    ▾ {
       id:             integer
       form_reply_key: string
       form_id:        integer
       completed:      integer
       step:           integer
       created_by:     string
       created_at:     string
       updated_at:     string
     }
}
```

401 Unauthorized ⇄

```
▾formRepliesMethodErrorDefault {
   error: string *
}
```

404 Form Reply not Found ⇄

```
▾formRepliesMethodErrorDefault {
   error: string *
}
```

500 Failed to retrieve the Form Reply ⇄

```
▾formRepliesMethodErrorDefault {
   error: string *
}
```

Try this operation

`DELETE /formReply/{form_reply_key}`

Form Reply Method

Summary

Delete a Form Reply

Parameters

Name	Located in	Description	Required	Schema
form_reply_key	path	Form Reply Key	Yes	⇄ `integer`
X-MODULE-TOKEN	header	Module Token	Yes	⇄ `string`
X-AUTH-TOKEN	header	User Auth Token	Yes	⇄ `string`

Responses

Code	Description	Schema

200	OK	⇄ ▾**formReplyDeleteReply {**
		`string: string`
		}

401	Unauthorized	⇄ ▾**formRepliesMethodErrorDefault {**
		`error: string *`
		}

404	Form Reply not Found	⇄ ▾**formRepliesMethodErrorDefault {**
		`error: string *`
		}

500	Failed to delete Form Reply	⇄ ▾**formRepliesMethodErrorDefault {**
		`error: string *`
		}

Try this operation

`/formReply`

POST /formReply

Form Reply Method

Summary

Create a Form Reply Method

Parameters

Name	Located in	Description	Required	Schema
Form Reply	body	Form Reply Method data	Yes	⇄

```
▾formRepliesCreate {
  all of:
    ▾ {
        form_key:          string
                           *
        completed:         boolean
                           *
        step:              integer
        question_replies:  ▸[]
      }
}
```

Name	Located in	Description	Required	Schema
X-AUTH-TOKEN	header	User Auth Token	Yes	⇄ string
X-MODULE-TOKEN	header	Module Token	Yes	⇄ string

Responses

Code	Description	Schema
201	The newly created Form Reply	⇄

```
▾formRepliesReply {
  all of:
    ▾ {
        id:              integer
        form_reply_key:  string
        form_id:         integer
        completed:       integer
        step:            integer
        created_by:      string
        created_at:      string
        updated_at:      string
      }
}
```

Code	Description	Schema
401	Unauthorized	⇄

```
▾formRepliesMethodErrorDefault {
  error: string *
}
```

Code	Description	Schema
500	Failed to store new Form Reply	⇄

```
▾formRepliesMethodErrorDefault {
  error: string *
}
```

Try this operation

/formReplyAnswer/{form_reply_answer_key}

`GET /formReplyAnswer/{form_reply_answer_key}`

Form Reply Answer

Summary

Show a form reply answer

Parameters

Name	Located in	Description	Required	Schema
form_reply_answer_key	path	Form reply answer Key	Yes	⇄ string
X-MODULE-TOKEN	header	Module Token	Yes	⇄ string
X-AUTH-TOKEN	header	User Auth Token	Yes	⇄ string

Responses

Code Description Schema

200 Show the Form Reply Answer data ⇄

```
▾formReplyAnswerReply {
  all of:
     ▾ {
         id:                    integer
         form_reply_answer_key: string
         form_reply_id:         integer
         question_id:           integer
         question_option_id:    integer
         answer:                string
         created_at:            string
         updated_at:            string
         deleted_at:            string
     }
}
```

401 Unauthorized ⇄

```
▾formReplyAnswerErrorDefault {
   error: string
}
```

404 Form Reply Answer not Found ⇄

```
▾formReplyAnswerErrorDefault {
   error: string
}
```

500 Failed to retrieve Form Reply Answer ⇄

```
▾formReplyAnswerErrorDefault {
   error: string
}
```

Try this operation

`PUT /formReplyAnswer/{form_reply_answer_key}`

Form Reply Answer

Summary

Update a Form Reply Answer

Parameters

Name	Located in	Description	Required	Schema
Body	body	Form Reply Answer Data	Yes	⇄ ▸ `formReplyAnswerCreate { }`
form_reply_answer_key	path	Form Reply Answer Key	Yes	⇄ `string`
X-AUTH-TOKEN	header	User Auth Token	Yes	⇄ `string`
X-MODULE-TOKEN	header	Module Token	Yes	⇄ `string`

Responses

Code	Description	Schema
200	The updated Form Reply Answer	⇄

```
▾formReplyAnswerReply {
   all of:
     ▾ {
        id:                    integer
        form_reply_answer_key: string
        form_reply_id:         integer
        question_id:           integer
        question_option_id:    integer
        answer:                string
        created_at:            string
        updated_at:            string
        deleted_at:            string
     }
}
```

Code	Description	Schema
401	Unauthorized	⇄

```
▾formReplyAnswerErrorDefault {
   error: string
}
```

Code	Description	Schema
404	Form Reply Answer not Found	⇄

```
▾formReplyAnswerErrorDefault {
   error: string
}
```

Code	Description	Schema
500	Failed to update Form Reply Answer	⇄

```
▾formReplyAnswerErrorDefault {
   error: string
}
```

Try this operation

`DELETE /formReplyAnswer/{form_reply_answer_key}`

Form Reply Answer

Summary

Delete a Form reply answer

Parameters

Name	Located in	Description	Required	Schema
form_reply_answer_key	path	Form reply answer Key	Yes	⇄ `integer`
X-MODULE-TOKEN	header	Module Token	Yes	⇄ `string`
X-AUTH-TOKEN	header	User Auth Token	Yes	⇄ `string`

Responses

Code	Description	Schema

200	OK	⇄ ▾**formReplyAnswerDeleteReply** {
		`string: string`
		}

401	Unauthorized	⇄ ▾**formReplyAnswerErrorDefault** {
		`error: string`
		}

404	Form Reply Answer not Found	⇄ ▾**formReplyAnswerErrorDefault** {
		`error: string`
		}

500	Failed to delete Form Reply Answer	⇄ ▾**formReplyAnswerErrorDefault** {
		`error: string`
		}

Try this operation

/formReplyAnswer

`POST /formReplyAnswer`

Form Reply Answer

Summary

Create a Form Reply Answer

Parameters

Name	Located in	Description	Required	Schema
Body	body	Form Reply Answer Data	Yes	⇄ ▸ **formReplyAnswerCreate** { }
X-AUTH-TOKEN	header	User Auth Token	Yes	⇄ string
X-MODULE-TOKEN	header	Module Token	Yes	⇄ string

Responses

Code	Description		Schema
201	the newly created Form Reply Answer	⇄	

```
▾formReplyAnswerReply {
  all of:
     ▾ {
        id:                     integer
        form_reply_answer_key:  string
        form_reply_id:          integer
        question_id:            integer
        question_option_id:     integer
        answer:                 string
        created_at:             string
        updated_at:             string
        deleted_at:             string
     }
}
```

Code	Description		Schema
401	Unauthorized	⇄	

```
▾formReplyAnswerErrorDefault {
   error: string
}
```

Code	Description		Schema
404	Form Reply Answer not found	⇄	

```
▾formReplyAnswerErrorDefault {
   error: string
}
```

Code	Description		Schema
500	Failed to store Form Reply Answer	⇄	

```
▾formReplyAnswerErrorDefault {
   error: string
}
```

Try this operation

`/form/{form_key}`

`GET /form/{form_key}`

Form Method

Summary

Shows a Form Method

Parameters

Name	Located in	Description	Required	Schema
form_key	path	Form Method Key	Yes	⇄ `string`
X-MODULE-TOKEN	header	Module Token	Yes	⇄ `string`
X-ENTITY-KEY	header	Entity Key	Yes	⇄ `string`
LANG-CODE	header	Entity Key	No	⇄ `string`
LANG-CODE-DEFAULT	header	Entity Key	Yes	⇄ `string`

Responses

Code	Description	Schema
200	Shows the Form data	⇄
401	Unauthorized	⇄
404	Form not Found	⇄
500	Failed to retrieve the Form	⇄

200

```
▾formsReply {
  all of:
    ▾ {
        form_key:     string
        entity_key:   string
        title:        string
        description:  string
        public:       integer
        created_by:   string
        start_date:   string
        end_date:     string
        link:         string
        file_id:      string
        created_at:   string
        updated_at:   string
      }
}
```

401

```
▾formsMethodErrorDefault {
   error: string *
}
```

404

```
▾formsMethodErrorDefault {
   error: string *
}
```

500

```
▾formsMethodErrorDefault {
   error: string *
}
```

Try this operation

`PUT /form/{form_key}`

Form Method

Summary

Update a Form Method

Parameters

Name	Located in	Description	Required	Schema
Form	body	Form Method data	Yes	⇄
form_key	path	Form Key	Yes	⇄ string
X-ENTITY-KEY	header	Entity Key	Yes	⇄ string
X-AUTH-TOKEN	header	User Auth Token	Yes	⇄ string
X-MODULE-TOKEN	header	Module Token	Yes	⇄ string

```
▾formsCreate {
  all of:
    ▾ {
      title:          string
                      *
      description:    string
                      *
      public:         integer
                      *
      start_date:     string
                      *
      end_date:       string
                      *
      link:           string
                      *
      file_id:        string
                      *
      configurations: ▸[]
    }
}
```

Responses

Code	Description	Schema

200 The Updated Form ⇄

```
▾formsReply {
  all of:
    ▾ {
        form_key:    string
        entity_key:  string
        title:       string
        description: string
        public:      integer
        created_by:  string
        start_date:  string
        end_date:    string
        link:        string
        file_id:     string
        created_at:  string
        updated_at:  string
      }
}
```

401 Unauthorized ⇄

```
▾formsMethodErrorDefault {
   error: string *
}
```

404 Form not Found ⇄

```
▾formsMethodErrorDefault {
   error: string *
}
```

500 Failed to Update Form ⇄

```
▾formsMethodErrorDefault {
   error: string *
}
```

Try this operation

`DELETE /form/{form_key}`

Form Method

Summary

Delete Form Method

Parameters

Name	Located in	Description	Required	Schema
form_key	path	Form Key	Yes	⇄ `string`
X-ENTITY-KEY	header	Entity Key	Yes	⇄ `string`
X-MODULE-TOKEN	header	Module Token	Yes	⇄ `string`
X-AUTH-TOKEN	header	User Auth Token	Yes	⇄ `string`

Responses

Code	Description	Schema

200	OK	⇄ ▾**formDeleteReply {**
		`string: string`
		`}`

401	Unauthorized	⇄ ▾**formsMethodErrorDefault {**
		`error: string *`
		`}`

404	Form not Found	⇄ ▾**formsMethodErrorDefault {**
		`error: string *`
		`}`

500	Failed to delete a Form	⇄ ▾**formsMethodErrorDefault {**
		`error: string *`
		`}`

Try this operation

/form

POST /form

Form Method

Summary

Create a Form Method

Parameters

Name	Located in	Description	Required	Schema
Form	body	Form Method data	Yes	⇄
X-ENTITY-KEY	header	Entity Key	Yes	⇄ string
X-AUTH-TOKEN	header	User Auth Token	Yes	⇄ string
X-MODULE-TOKEN	header	Module Token	Yes	⇄ string

Form:

```
▾formsCreate {
  all of:
    ▾ {
      title:            string
                        *
      description:      string
                        *
      public:           integer
                        *
      start_date:       string
                        *
      end_date:         string
                        *
      link:             string
                        *
      file_id:          string
                        *
      configurations:  ▾[
                          ▸ {
                          }
                        ]
    }
}
```

Responses

Code	Description	Schema

201 The newly created Form ⇄

```
▾formsReply {
  all of:
    ▾ {
        form_key:    string
        entity_key:  string
        title:       string
        description: string
        public:      integer
        created_by:  string
        start_date:  string
        end_date:    string
        link:        string
        file_id:     string
        created_at:  string
        updated_at:  string
      }
}
```

401 Unauthorized ⇄

```
▾formsMethodErrorDefault {
  error: string *
}
```

500 Failed to store new Form ⇄

```
▾formsMethodErrorDefault {
  error: string *
}
```

Try this operation

/icon/{icon_key}

`GET /icon/{icon_key}`

Icon Method

Summary

Shows a Icon Method

Parameters

Name	Located in	Description	Required	Schema
icon_key	path	Icon Method Key	Yes	⇄ `string`
X-MODULE-TOKEN	header	Module Token	Yes	⇄ `string`

Responses

Code	Description	Schema
200	Shows the Icon data	⇄

```
▾iconsReply {
  all of:
    ▾ {
        id:         integer
        icon_key:   string
        name:       string
        file_id:    string
        file_code:  string
        created_at: string
        updated_at: string
      }
}
```

Code	Description	Schema
401	Unauthorized	⇄

```
▾iconsMethodErrorDefault {
  error: string *
}
```

Code	Description	Schema
404	Icon not Found	⇄

```
▾iconsMethodErrorDefault {
  error: string *
}
```

Code	Description	Schema
500	Failed to retrieve Icon	⇄

```
▾iconsMethodErrorDefault {
  error: string *
}
```

Try this operation

PUT /icon/{icon_key}

Icon Method

Summary

Update a Icon Method

Parameters

Name	Located in	Description	Required	Schema
Icon	body	Icon Method data	Yes	⇄

```
▾iconsCreate {
  all of:
    ▾ {
        name:      string
                   *
        file_id:   string
                   *
        file_code: string
                   *
      }
}
```

Name	Located in	Description	Required	Schema
icon_key	path	Icon Key	Yes	⇄ string
X-AUTH-TOKEN	header	User Auth Token	Yes	⇄ string
X-MODULE-TOKEN	header	Module Token	Yes	⇄ string

Responses

Code	Description	Schema
200	The Updated Icon	⇄

```
▾iconsReply {
  all of:
    ▾ {
        id:         integer
        icon_key:   string
        name:       string
        file_id:    string
        file_code:  string
        created_at: string
        updated_at: string
      }
}
```

Code	Description	Schema
401	Unauthorized	⇄

```
▾iconsMethodErrorDefault {
   error: string *
}
```

Code	Description	Schema
404	Icon not Found	⇄

```
▾iconsMethodErrorDefault {
   error: string *
}
```

Code	Description	Schema
500	Failed to Update Icon	⇄

```
▾iconsMethodErrorDefault {
   error: string *
}
```

Try this operation

`DELETE /icon/{icon_key}`

Icon Method

Summary

Delete Icon Method

Parameters

Name	Located in	Description	Required	Schema
icon_key	path	Icon Key	Yes	⇄ `string`
X-MODULE-TOKEN	header	Module Token	Yes	⇄ `string`
X-AUTH-TOKEN	header	User Auth Token	Yes	⇄ `string`

Responses

Code	Description	Schema

200	OK	⇄ ▾**iconDeleteReply {**
		`string: string`
		}

401	Unauthorized	⇄ ▾**iconsMethodErrorDefault {**
		`error: string *`
		}

404	Icon not Found	⇄ ▾**iconsMethodErrorDefault {**
		`error: string *`
		}

500	Failed to delete a Icon	⇄ ▾**iconsMethodErrorDefault {**
		`error: string *`
		}

Try this operation

/icon

POST /icon

Icon Method

Summary

Create a Icon Method

Parameters

Name	Located in	Description	Required	Schema
Icon	body	Icon Method data	Yes	⇄

```
▾iconsCreate {
  all of:
    ▾ {
      name:      string
                 *
      file_id:   string
                 *
      file_code: string
                 *
    }
}
```

X-AUTH-TOKEN	header	User Auth Token	Yes	⇄ string
X-MODULE-TOKEN	header	Module Token	Yes	⇄ string

Responses

Code	Description	Schema
201	The newly created Icon	⇄

```
▾iconsReply {
  all of:
    ▾ {
      id:         integer
      icon_key:   string
      name:       string
      file_id:    string
      file_code:  string
      created_at: string
      updated_at: string
    }
}
```

401	Unauthorized	⇄ ▾iconsMethodErrorDefault {
		error: string *
		}

500	Failed to store new Icon	⇄ ▾iconsMethodErrorDefault {
		error: string *
		}

Try this operation

/questionGroup/{question_group_key}

`GET /questionGroup/{question_group_key}`

Question Group Method

Summary

Shows a Question Group Method

Parameters

Name	Located in	Description	Required	Schema
question_group_key	path	Question Group Method Key	Yes	⇄ string
X-MODULE-TOKEN	header	Module Token	Yes	⇄ string

Responses

Code	Description	Schema
200	Shows the Question Group data	⇄
401	Unauthorized	⇄
404	Question Group not Found	⇄
500	Failed to retrieve Question Group	⇄

200

```
▾questionGroupsReply {
  all of:
    ▾ {
        question_group_key: string
        form_id:            integer
        title:              string
        description:        string
        position:           integer
        created_at:         string
        updated_at:         string
      }
}
```

401

```
▾questionGroupsMethodErrorDefault {
   error: string *
}
```

404

```
▾questionGroupsMethodErrorDefault {
   error: string *
}
```

500

```
▾questionGroupsMethodErrorDefault {
   error: string *
}
```

Try this operation

`PUT /questionGroup/{question_group_key}`

Question Group Method

Summary

Update a Question Group Method

Parameters

Name	Located in	Description	Required	Schema
Question Group	body	Question Group Method data	Yes	⇄ ▸ **`questionGroupsUpdate { }`**
question_group_key	path	Question Group Key	Yes	⇄ `string`
X-AUTH-TOKEN	header	User Auth Token	Yes	⇄ `string`
X-MODULE-TOKEN	header	Module Token	Yes	⇄ `string`

Responses

Code	Description	Schema
200	The Updated Question Group	⇄

```
▾questionGroupsReply {
  all of:
    ▾ {
        question_group_key: string
        form_id:            integer
        title:              string
        description:        string
        position:           integer
        created_at:         string
        updated_at:         string
    }
}
```

Code	Description	Schema
401	Unauthorized	⇄

```
▾questionGroupsMethodErrorDefault {
   error: string *
}
```

Code	Description	Schema
404	Question Group not Found	⇄

```
▾questionGroupsMethodErrorDefault {
   error: string *
}
```

Code	Description	Schema
500	Failed to Update Question Group	⇄

```
▾questionGroupsMethodErrorDefault {
   error: string *
}
```

Try this operation

`DELETE /questionGroup/{question_group_key}`

Question Group Method

Summary

Delete Question Group Method

Parameters

Name	Located in	Description	Required	Schema
question_group_key	path	Question Group Key	Yes	⇄ `string`
X-MODULE-TOKEN	header	Module Token	Yes	⇄ `string`
X-AUTH-TOKEN	header	User Auth Token	Yes	⇄ `string`

Responses

Code	Description	Schema

200	OK	⇄ ▾ **`questionGroupDeleteReply {`**
		`string: string`
		`}`

401	Unauthorized	⇄ ▾ **`questionGroupsMethodErrorDefault {`**
		`error: string *`
		`}`

404	Question Group not Found	⇄ ▾ **`questionGroupsMethodErrorDefault {`**
		`error: string *`
		`}`

500	Failed to delete a Question Group	⇄ ▾ **`questionGroupsMethodErrorDefault {`**
		`error: string *`
		`}`

Try this operation

`/questionGroup`

`POST /questionGroup`

Question Group Method

Summary

Create a Question Group Method

Parameters

Name	Located in	Description	Required	Schema
Question Group	body	Question Group Method data	Yes	⇄ ▸ **`questionGroupsCreate { }`**
X-AUTH-TOKEN	header	User Auth Token	Yes	⇄ `string`
X-MODULE-TOKEN	header	Module Token	Yes	⇄ `string`

Responses

Code	Description	Schema
201	The newly created Question Group	⇄
401	Unauthorized	⇄
500	Failed to store new Question Group	⇄

201

```
▾questionGroupsReply {
  all of:
    ▾ {
        question_group_key: string
        form_id:            integer
        title:              string
        description:        string
        position:           integer
        created_at:         string
        updated_at:         string
      }
}
```

401

```
▾questionGroupsMethodErrorDefault {
   error: string *
}
```

500

```
▾questionGroupsMethodErrorDefault {
   error: string *
}
```

Try this operation

`/questionOption/{question_option_key}`

`GET /questionOption/{question_option_key}`

Question Option Method

Summary

Shows a Question Option Method

Parameters

Name	Located in	Description	Required	Schema
question_option_key	path	Question Option Method Key	Yes	⇄ `string`
X-MODULE-TOKEN	header	Module Token	Yes	⇄ `string`

Responses

Code	Description	Schema
200	Shows the Question Option data	⇄

```
▾questionOptionsReply {
  all of:
    ▾ {
       id:                 integer
       question_option_key: string
       icon_id:            string
       question_id:        string
       label:              string
       file_id:            string
       file_code:          string
       position:           integer
       created_at:         string
       updated_at:         string
     }
}
```

Code	Description	Schema
401	Unauthorized	⇄

```
▾questionOptionsMethodErrorDefault {
   error: string *
}
```

Code	Description	Schema
404	Question Option not Found	⇄

```
▾questionOptionsMethodErrorDefault {
   error: string *
}
```

Code	Description	Schema
500	Failed to retrieve the Question Option	⇄

```
▾questionOptionsMethodErrorDefault {
   error: string *
}
```

Try this operation

PUT /questionOption/{question_option_key}

Question Option Method

Summary

Update a Question Option Method

Parameters

Name	Located in	Description	Required	Schema
Question Option	body	Question Option Method data	Yes	⇄
question_option_key	path	Question Option Key	Yes	⇄ string
X-AUTH-TOKEN	header	User Auth Token	Yes	⇄ string
X-MODULE-TOKEN	header	Module Token	Yes	⇄ string

```
▾questionOptionsUpdate {
  all of:
   ▾ {
     question_key: string
                   *
     icon_key:     string
                   *
     label:        string
                   *
     dependencies: ▸[]
   }
}
```

Responses

Code	Description	Schema
200	The Updated Question Option	⇄
401	Unauthorized	⇄

```
▾questionOptionsReply {
  all of:
   ▾ {
     id:                  integer
     question_option_key: string
     icon_id:             string
     question_id:         string
     label:               string
     file_id:             string
     file_code:           string
     position:            integer
     created_at:          string
     updated_at:          string
   }
}
```

```
▾questionOptionsMethodErrorDefault {
  error: string *
}
```

Code	Description	Schema

404	Question Option not Found	⇄ `▾questionOptionsMethodErrorDefault {`
		`  error: string *`
		`}`

500	Failed to Update Question Option	⇄ `▾questionOptionsMethodErrorDefault {`
		`  error: string *`
		`}`

Try this operation

`DELETE /questionOption/{question_option_key}`

Question Option Method

Summary

Delete Question Option Method

Parameters

Name	Located in	Description	Required	Schema
question_option_key	path	Question Option Key	Yes	⇄ `string`
X-MODULE-TOKEN	header	Module Token	Yes	⇄ `string`
X-AUTH-TOKEN	header	User Auth Token	Yes	⇄ `string`

Responses

Code	Description	Schema

200	OK	⇄ `▾questionOptionDeleteReply {`
		`  string: string`
		`}`

401	Unauthorized	⇄ `▾questionOptionsMethodErrorDefault {`
		`  error: string *`
		`}`

404	Question Option not Found	⇄ `▾questionOptionsMethodErrorDefault {`
		`  error: string *`
		`}`

500	Failed to delete a Question Option	⇄ `▾questionOptionsMethodErrorDefault {`
		`  error: string *`
		`}`

Try this operation

`/questionOption`

`POST /questionOption`

Question Option Method

Summary

Create a Question Option Method

Parameters

Name	Located in	Description	Required	Schema
Question Option	body	Question Option Method data	Yes	⇄
X-AUTH-TOKEN	header	User Auth Token	Yes	⇄ string
X-MODULE-TOKEN	header	Module Token	Yes	⇄ string

Question Option:

```
▾questionOptionsCreate {
  all of:
    ▾ {
        question_key: string
                      *
        icon_key:     string
                      *
        label:        string
                      *
      }
}
```

Responses

Code	Description	Schema
201	The newly created Question Option	⇄
401	Unauthorized	⇄
500	Failed to store new Question Option	⇄

201:

```
▾questionOptionsReply {
  all of:
    ▾ {
        id:                  integer
        question_option_key: string
        icon_id:             string
        question_id:         string
        label:               string
        file_id:             string
        file_code:           string
        position:            integer
        created_at:          string
        updated_at:          string
      }
}
```

401:

```
▾questionOptionsMethodErrorDefault {
   error: string *
}
```

500:

```
▾questionOptionsMethodErrorDefault {
   error: string *
}
```

Try this operation

`/questionType/{question_type_key}`

GET /questionType/{question_type_key}

Question Type Method

Summary

Shows a Question Type Method

Parameters

Name	Located in	Description	Required	Schema
question_type_key	path	Question Type Method Key	Yes	⇄ string
X-MODULE-TOKEN	header	Module Token	Yes	⇄ string

Responses

Code	Description	Schema

200 Shows the Question Type data ⇄

```
▾questionTypesReply {
  all of:
    ▾ {
      id:                 integer
      question_type_key:  string
      name:               string
      created_at:         string
      updated_at:         string
    }
}
```

401 Unauthorized ⇄

```
▾questionTypesMethodErrorDefault {
  error: string *
}
```

404 Question Type not Found ⇄

```
▾questionTypesMethodErrorDefault {
  error: string *
}
```

500 Failed to retrieve Question Type ⇄

```
▾questionTypesMethodErrorDefault {
  error: string *
}
```

Try this operation

PUT /questionType/{question_type_key}

Question Type Method

Summary

Update a Question Type Method

Parameters

Name	Located in	Description	Required	Schema
Question Type	body	Question Type Method data	Yes	⇄
question_type_key	path	Question Type Key	Yes	⇄ `string`
X-AUTH-TOKEN	header	User Auth Token	Yes	⇄ `string`
X-MODULE-TOKEN	header	Module Token	Yes	⇄ `string`

```
▾
questionTypesCreate
{
   all of:
     ▾ {
         name: string
                  *
       }
}
```

Responses

Code	Description	Schema
200	The Updated Question Type	⇄
401	Unauthorized	⇄
404	Question Type not Found	⇄
500	Failed to Update Question Type	⇄

200

```
▾questionTypesReply {
  all of:
    ▾ {
        id:                 integer
        question_type_key:  string
        name:               string
        created_at:         string
        updated_at:         string
      }
}
```

401

```
▾questionTypesMethodErrorDefault {
   error: string *
}
```

404

```
▾questionTypesMethodErrorDefault {
   error: string *
}
```

500

```
▾questionTypesMethodErrorDefault {
   error: string *
}
```

Try this operation

`DELETE /questionType/{question_type_key}`

Question Type Method

Summary

Delete Question Type Method

Parameters

Name	Located in	Description	Required	Schema
question_type_key	path	Question Type Key	Yes	⇄ `string`
X-MODULE-TOKEN	header	Module Token	Yes	⇄ `string`
X-AUTH-TOKEN	header	User Auth Token	Yes	⇄ `string`

Responses

Code	Description	Schema

200	OK	⇄ ▾`questionTypeDeleteReply {`
		`string: string`
		`}`

401	Unauthorized	⇄ ▾`questionTypesMethodErrorDefault {`
		`error: string *`
		`}`

404	Question Type not Found	⇄ ▾`questionTypesMethodErrorDefault {`
		`error: string *`
		`}`

500	Failed to delete a Question Type	⇄ ▾`questionTypesMethodErrorDefault {`
		`error: string *`
		`}`

Try this operation

/questionType

POST /questionType

Question Type Method

Summary

Create a Question Type Method

Parameters

Name	Located in	Description	Required	Schema
Question Type	body	Question Type Method data	Yes	⇄
X-AUTH-TOKEN	header	User Auth Token	Yes	⇄ string
X-MODULE-TOKEN	header	Module Token	Yes	⇄ string

Question Type:

```
▾
questionTypesCreate
{
   all of:
     ▾ {
         name: string
                 *
       }
}
```

Responses

Code	Description	Schema
201	The newly created Question Type	⇄
401	Unauthorized	⇄
500	Failed to store new Question Type	⇄

201:

```
▾questionTypesReply {
  all of:
     ▾ {
        id:                integer
        question_type_key: string
        name:              string
        created_at:        string
        updated_at:        string
       }
}
```

401:

```
▾questionTypesMethodErrorDefault {
   error: string *
}
```

500:

```
▾questionTypesMethodErrorDefault {
   error: string *
}
```

Try this operation

/question/{question_key}

`GET /question/{question_key}`

Question Method

Summary

Shows a Question Method

Parameters

Name	Located in	Description	Required	Schema
question_key	path	Question Method Key	Yes	⇄ `string`
X-MODULE-TOKEN	header	Module Token	Yes	⇄ `string`

Responses

Code	Description	Schema
200	Shows the Question data	⇄

```
▾questionsReply {
  all of:
    ▾ {
        id:                      integer
        question_key:            string
        question_group_id:       integer
        question_type_id:        integer
        question:                string
        description:             string
        mandatory:               integer
        position:                integer
        reuse_question_options:  integer
        created_at:              string
        updated_at:              string
    }
}
```

Code	Description	Schema
401	Unauthorized	⇄

```
▾questionsMethodErrorDefault {
   error: string *
}
```

Code	Description	Schema
404	Question not Found	⇄

```
▾questionsMethodErrorDefault {
   error: string *
}
```

Code	Description	Schema
500	Failed to retrieve the Question	⇄

```
▾questionsMethodErrorDefault {
   error: string *
}
```

Try this operation

`PUT /question/{question_key}`

Question Method

Summary

Update a Question Method

Parameters

Name	Located in	Description	Required	Schema
Question	body	Question Method data	Yes	⇄
question_key	path	Question Key	Yes	⇄ `string`
X-AUTH-TOKEN	header	User Auth Token	Yes	⇄ `string`
X-MODULE-TOKEN	header	Module Token	Yes	⇄ `string`

```
▾questionsCreate {
  all of:
     ▾ {
        question_group_key:      string
                                 *
        question_type_key:       string
                                 *
        question:                string
                                 *
        description:             string
                                 *
        mandatory:               integer
                                 *
        position:                integer
                                 *
        reuse_question_options:  integer
     }
}
```

Responses

Code	Description	Schema

200 The Updated Question ⇄

```
▾questionsReply {
  all of:
    ▾ {
      id:                      integer
      question_key:            string
      question_group_id:       integer
      question_type_id:        integer
      question:                string
      description:             string
      mandatory:               integer
      position:                integer
      reuse_question_options:  integer
      created_at:              string
      updated_at:              string
    }
}
```

401 Unauthorized ⇄

```
▾questionsMethodErrorDefault {
   error: string *
}
```

404 Question not Found ⇄

```
▾questionsMethodErrorDefault {
   error: string *
}
```

500 Failed to Update Question ⇄

```
▾questionsMethodErrorDefault {
   error: string *
}
```

Try this operation

`DELETE /question/{question_key}`

Question Method

Summary

Delete Question Method

Parameters

Name	Located in	Description	Required	Schema
question_key	path	Question Key	Yes	⇄ `string`
X-MODULE-TOKEN	header	Module Token	Yes	⇄ `string`
X-AUTH-TOKEN	header	User Auth Token	Yes	⇄ `string`

Responses

Code	Description	Schema

200	OK	⇄ ▾**`questionDeleteReply {`**
		`string: string`
		`}`

401	Unauthorized	⇄ ▾**`questionsMethodErrorDefault {`**
		`error: string *`
		`}`

404	Question not Found	⇄ ▾**`questionsMethodErrorDefault {`**
		`error: string *`
		`}`

500	Failed to delete a Question	⇄ ▾**`questionsMethodErrorDefault {`**
		`error: string *`
		`}`

Try this operation

`/question`

`POST /question`

Question Method

Summary

Create a Question Method

Parameters

Name	Located in	Description	Required	Schema
Question	body	Question Method data	Yes	⇄
X-AUTH-TOKEN	header	User Auth Token	Yes	⇄ `string`
X-MODULE-TOKEN	header	Module Token	Yes	⇄ `string`

Question:

```
▾questionsCreate {
  all of:
    ▾ {
      question_group_key:      string
                               *
      question_type_key:       string
                               *
      question:                string
                               *
      description:             string
                               *
      mandatory:               integer
                               *
      position:                integer
                               *
      reuse_question_options:  integer
    }
}
```

Responses

Code	Description	Schema
201	The newly created Question	⇄

201:

```
▾questionsReply {
  all of:
    ▾ {
      id:                      integer
      question_key:            string
      question_group_id:       integer
      question_type_id:        integer
      question:                string
      description:             string
      mandatory:               integer
      position:                integer
      reuse_question_options:  integer
      created_at:              string
      updated_at:              string
    }
}
```

Code	Description	Schema

401	Unauthorized	⇄ ▾**questionsMethodErrorDefault {**
		error: string *
		}

500	Failed to store new Question	⇄ ▾**questionsMethodErrorDefault {**
		error: string *
		}

Try this operation

Models

esParticipantsMethodErrorDefault

```
  ▾esParticipantsMethodErrorDefault {
⇄    error: string *
  }
```

esParticipantsReply

```
  ▾esParticipantsReply {
     all of:
       ▾ {
          id:                integer
          key:               string
          event_schedule_id: integer
          name:              string
⇄         user_key:          string
          created_by:        string
          updated_by:        string
          created_at:        string
          updated_at:        string
        }
  }
```

esParticipantsCreate

```
  ▾esParticipantsCreate {
     all of:
       ▾ {
          name:      string *
          periods:   ▾[
                        integer
⇄                    ]
          questions: ▾[
                        integer
                     ]
        }
  }
```

esParticipantsUpdate

```
▾esParticipantsUpdate {
  all of:
    ▾ {
      participant_id: integer *
      name:           string *
      periods:        ▾[
                        integer
                      ]
      questions:      ▾[
                        integer
                      ]
    }
}
```

esParticipantDeleteReply

```
▾esParticipantDeleteReply {
  string: string
}
```

eventSchedulesMethodErrorDefault

```
▾eventSchedulesMethodErrorDefault {
  error: string *
}
```

eventSchedulesReply

```
▾eventSchedulesReply {
  all of:
    ▾ {
      id:             integer
      key:            string
      entity_id:      integer
      type_id:        integer
      title:          string
      description:    string
      local:          string
      closed:         integer
      public:         integer
      created_by:     string
      updated_by:     string
      es_period_id:   integer
      es_question_id: integer
      created_at:     string
      updated_at:     string
    }
}
```

eventSchedulesCreate

```
▾eventSchedulesCreate {
  all of:
    ▾ {
      entity_id:   integer *
      title:       string *
      type_id:     integer *
      description: string *
      local:       string *
      closed:      integer *
      public:      integer *
      periods:     ▾[
                     ▾ {
⇄                      start_date: string
                       end_date:   string
                       start_time: string
                       end_time:   string
                     }
                   ]
      questions:   ▾[
                     ▾ {
                       question: string
                     }
                   ]
    }
}
```

eventSchedulesUpdate

```
▾eventSchedulesUpdate {
  all of:
    ▾ {
      entity_id:      integer *
      title:          string *
      type_id:        integer *
      description:    string *
      local:          string *
      closed:         integer *
      public:         integer *
      periods:        ▾[
                        ▾ {
                          start_date: string
                          end_date:   string
                          start_time: string
                          end_time:   string
                        }
                      ]
      questions:      ▾[
                        ▾ {
                          question: string
                        }
                      ]
      es_period_id:   integer *
      es_question_id: integer *
    }
}
```

eventScheduleDeleteReply

```
▾eventScheduleDeleteReply {
  string: string
}
```

formConfigurationsMethodErrorDefault

```
▾formConfigurationsMethodErrorDefault {
  error: string *
}
```

formConfigurationsReply

```
▾formConfigurationsReply {
  all of:
    ▾ {
      id:                     integer
      form_configuration_key: string
      code:                   string
      updated_at:             string
    }
}
```

formConfigurationTranslations

```
▾formConfigurationTranslations {
  all of:
    ▾ {
      language_code: string *
      name:          string *
      description:   string
    }
}
```

formConfigurationsCreate

```
▾formConfigurationsCreate {
  all of:
    ▾ {
      code:         string *
      translations: ▾[
                      ▾formConfigurationTranslations {
                        all of:
                          ▾ {
                            language_code: string *
                            name:          string *
                            description:   string
                          }
                      }
                    ]
    }
}
```

formConfigurationDeleteReply

```
▾formConfigurationDeleteReply {
  string: string
}
```

formRepliesMethodErrorDefault

```
▾formRepliesMethodErrorDefault {
  error: string *
}
```

formRepliesReply

```
  ▾formRepliesReply {
    all of:
      ▾ {
        id:             integer
        form_reply_key: string
        form_id:        integer
⇄       completed:      integer
        step:           integer
        created_by:     string
        created_at:     string
        updated_at:     string
      }
  }
```

formRepliesCreate

```
  ▾formRepliesCreate {
    all of:
      ▾ {
        form_key:         string *
        completed:        boolean *
        step:             integer
        question_replies: ▾[
⇄                           ▾ {
                              question_id: integer
                              string:      string
                            }
                          ]
      }
  }
```

formReplyDeleteReply

```
  ▾formReplyDeleteReply {
⇄   string: string
  }
```

formReplyAnswerErrorDefault

```
  ▾formReplyAnswerErrorDefault {
⇄   error: string
  }
```

formReplyAnswerCreate

```
  ▾formReplyAnswerCreate {
    all of:
⇄     ▾ {
        form_id: integer *
      }
  }
```

formReplyAnswerReply

```
    ▾formReplyAnswerReply {
      all of:
        ▾ {
          id:                     integer
          form_reply_answer_key: string
          form_reply_id:          integer
          question_id:            integer
⇄         question_option_id:    integer
          answer:                 string
          created_at:             string
          updated_at:             string
          deleted_at:             string
        }
    }
```

formReplyAnswerDeleteReply

```
    ▾formReplyAnswerDeleteReply {
⇄     string: string
    }
```

formsMethodErrorDefault

```
    ▾formsMethodErrorDefault {
⇄     error: string *
    }
```

formsReply

```
    ▾formsReply {
      all of:
        ▾ {
          form_key:    string
          entity_key:  string
          title:       string
          description: string
          public:      integer
⇄         created_by:  string
          start_date:  string
          end_date:    string
          link:        string
          file_id:     string
          created_at:  string
          updated_at:  string
        }
    }
```

formsCreate

```
   ▾formsCreate {
     all of:
        ▾ {
          title:          string *
          description:    string *
          public:         integer *
          start_date:     string *
          end_date:       string *
          link:           string *
⇄         file_id:        string *
          configurations: ▾[
                            ▾ {
                              form_configuration_key: string
                              value:                  string
                            }
                          ]
        }
   }
```

formDeleteReply

```
   ▾formDeleteReply {
⇄    string: string
   }
```

iconsMethodErrorDefault

```
   ▾iconsMethodErrorDefault {
⇄    error: string *
   }
```

iconsReply

```
   ▾iconsReply {
     all of:
        ▾ {
          id:         integer
          icon_key:   string
          name:       string
⇄         file_id:    string
          file_code:  string
          created_at: string
          updated_at: string
        }
   }
```

iconsCreate

```
  ▾iconsCreate {
    all of:
      ▾ {
        name:      string *
⇄       file_id:   string *
        file_code: string *
      }
  }
```

iconDeleteReply

```
  ▾iconDeleteReply {
⇄   string: string
  }
```

questionGroupsMethodErrorDefault

```
  ▾questionGroupsMethodErrorDefault {
⇄   error: string *
  }
```

questionGroupsReply

```
  ▾questionGroupsReply {
    all of:
      ▾ {
        question_group_key: string
        form_id:            integer
        title:              string
⇄       description:        string
        position:           integer
        created_at:         string
        updated_at:         string
      }
  }
```

questionGroupsCreate

```
  ▾questionGroupsCreate {
    all of:
      ▾ {
        form_key:    string *
⇄       title:       string *
        description: string *
      }
  }
```

questionGroupsUpdate

```
    ▾questionGroupsUpdate {
      all of:
        ▾ {
          form_key:    string *
⇄         title:       string *
          description: string *
          position:    integer
        }
    }
```

questionGroupDeleteReply

```
    ▾questionGroupDeleteReply {
⇄     string: string
    }
```

questionOptionsMethodErrorDefault

```
    ▾questionOptionsMethodErrorDefault {
⇄     error: string *
    }
```

questionOptionsReply

```
    ▾questionOptionsReply {
      all of:
        ▾ {
          id:                  integer
          question_option_key: string
          icon_id:             string
          question_id:         string
⇄         label:               string
          file_id:             string
          file_code:           string
          position:            integer
          created_at:          string
          updated_at:          string
        }
    }
```

questionOptionsCreate

```
    ▾questionOptionsCreate {
      all of:
        ▾ {
          question_key: string *
⇄         icon_key:     string *
          label:        string *
        }
    }
```

questionOptionsUpdate

```
▾questionOptionsUpdate {
  all of:
    ▾ {
      question_key: string *
      icon_key:     string *
      label:        string *
      dependencies: ▾[
                      string
                    ]
    }
}
```

questionOptionDeleteReply

```
▾questionOptionDeleteReply {
  string: string
}
```

questionTypesMethodErrorDefault

```
▾questionTypesMethodErrorDefault {
  error: string *
}
```

questionTypesReply

```
▾questionTypesReply {
  all of:
    ▾ {
      id:                integer
      question_type_key: string
      name:              string
      created_at:        string
      updated_at:        string
    }
}
```

questionTypesCreate

```
▾questionTypesCreate {
  all of:
    ▾ {
      name: string *
    }
}
```

questionTypeDeleteReply

```
▾questionTypeDeleteReply {
  string: string
}
```

questionsMethodErrorDefault

```
   ▾questionsMethodErrorDefault {
⇄     error: string *
   }
```

questionsReply

```
   ▾questionsReply {
     all of:
       ▾ {
         id:                     integer
         question_key:           string
         question_group_id:      integer
         question_type_id:       integer
         question:               string
⇄        description:            string
         mandatory:              integer
         position:               integer
         reuse_question_options: integer
         created_at:             string
         updated_at:             string
       }
   }
```

questionsCreate

```
   ▾questionsCreate {
     all of:
       ▾ {
         question_group_key:     string *
         question_type_key:      string *
         question:               string *
⇄        description:            string *
         mandatory:              integer *
         position:               integer *
         reuse_question_options: integer
       }
   }
```

questionDeleteReply

```
   ▾questionDeleteReply {
⇄     string: string
   }
```

Vote Module

Version 1.0.0

Contact information
pvalente@onesource.pt (mailto:pvalente@onesource.pt)

Paths

/configuration/{configuration_id}

GET /configuration/{configuration_id}

Configuration

Summary

Show a Configuration detail

Parameters

Name	Located in	Description	Required	Schema
configuration_id	path	Configuration id	Yes	⇄ string
X-MODULE-TOKEN	header	Module Token	Yes	⇄ string
X-ENTITY-KEY	header	Entity Token	Yes	⇄ string
LANG-CODE	header	Language code	Yes	⇄ string
LANG-CODE-DEFAULT	header	Default Language code	Yes	⇄ string

Responses

Code **Description** **Schema**

200 Show the Configuration data ⇄

```
▾configurationReply {
  all of:
    ▾ {
        id:             integer
        method_id:      integer
        code:           string
        parameter_type: string
        name:           string
        description:    string
        created_at:     string
        updated_at:     string
      }
}
```

401 Unauthorized ⇄

```
▾errorDefault {
  error: string
}
```

404 Configuration not Found ⇄

```
▾errorDefault {
  error: string
}
```

500 Failed to retrieve Configuration ⇄

```
▾errorDefault {
  error: string
}
```

Try this operation

PUT /configuration/{configuration_id}

Configuration

Summary

Update a Configuration

Parameters

Name	Located in	Description	Required	Schema
Body	body	Configuration Update Data	Yes	⇄

```
▾configurationCreate {
  all of:
    ▾ {
        method_id:      integer
                        *
        code:           string
        parameter_type: string
        translations:   ▸[]
    }
}
```

Name	Located in	Description	Required	Schema
configuration_id	path	Configuration id	Yes	⇄ integer
X-AUTH-TOKEN	header	User Auth Token	Yes	⇄ string
X-MODULE-TOKEN	header	Module Token	Yes	⇄ string
LANG-CODE	header	Language code	Yes	⇄ string
LANG-CODE-DEFAULT	header	Default Language code	Yes	⇄ string

Responses

Code	Description	Schema
200	The updated Configuration	⇄

```
▾configurationReply {
  all of:
    ▾ {
        id:             integer
        method_id:      integer
        code:           string
        parameter_type: string
        name:           string
        description:    string
        created_at:     string
        updated_at:     string
    }
}
```

Code	Description	Schema

401	Unauthorized	⇄ ▾**errorDefault {**
		`error: string`
		}

404	Configuration not Found	⇄ ▾**errorDefault {**
		`error: string`
		}

500	Failed to create Configuration	⇄ ▾**errorDefault {**
		`error: string`
		}

Try this operation

`DELETE /configuration/{configuration_id}`

Configuration

Summary

Delete a Configuration

Parameters

Name	Located in	Description	Required	Schema
configuration_id	path	Configuration Id	Yes	⇄ `integer`
X-MODULE-TOKEN	header	Module Token	Yes	⇄ `string`
X-AUTH-TOKEN	header	User Auth Token	Yes	⇄ `string`

Responses

Code	Description	Schema

200	OK	⇄ ▾**configurationDeleteReply {**
		`string: string`
		}

401	Unauthorized	⇄ ▾**errorDefault {**
		`error: string`
		}

404	Configuration not Found	⇄ ▾**errorDefault {**
		`error: string`
		}

500	Failed to delete Configuration	⇄ ▾**errorDefault {**
		`error: string`
		}

Try this operation

`/configuration`

`POST /configuration`

Configuration

Summary

Create a Configuration

Parameters

Name	Located in	Description	Required	Schema
Body	body	Configuration Created Data	Yes	⇄
X-AUTH-TOKEN	header	User Auth Token	Yes	⇄ `string`
X-MODULE-TOKEN	header	Module Token	Yes	⇄ `string`
LANG-CODE	header	Language code	Yes	⇄ `string`
LANG-CODE-DEFAULT	header	Default Language code	Yes	⇄ `string`

Body:

```
▾configurationCreate {
  all of:
    ▾ {
      method_id:      integer
                      *
      code:           string
      parameter_type: string
      translations:   ▸[]
    }
}
```

Responses

Code	Description	Schema
200	Created Configuration	⇄
401	Unauthorized	⇄
500	Failed to update Method	⇄

200:

```
▾configurationReply {
  all of:
    ▾ {
      id:             integer
      method_id:      integer
      code:           string
      parameter_type: string
      name:           string
      description:    string
      created_at:     string
      updated_at:     string
    }
}
```

401:

```
▾errorDefault {
  error: string
}
```

500:

```
▾errorDefault {
  error: string
}
```

Try this operation

/event/{event_key}

`GET /event/{event_key}`

Event

Summary

Show a Event detail

Parameters

Name	Located in	Description	Required	Schema
event_key	path	Event key	Yes	⇄ string
X-MODULE-TOKEN	header	Module Token	Yes	⇄ string
X-ENTITY-KEY	header	Entity Token	Yes	⇄ string
LANG-CODE	header	Language code	Yes	⇄ string
LANG-CODE-DEFAULT	header	Default Language code	Yes	⇄ string

Responses

Code	Description	Schema
200	Show the Event data	⇄

```
▾eventShowReply {
  all of:
    ▾ {
      key:             string
      name:            string
      method_id:       integer
      created_by:      string
      start_date:      string
      end_date:        string
      start_time:      string
      end_time:        string
      created_at:      string
      updated_at:      string
      method:          ▾ {
                         all of:
                           ▾eventMethodReply {
                             all of:
                               ▾ {
                                 id:              string
                                 code:            string
                                 method_group_id: integer
                                 name:            string
                                 description:     string
                                 created_at:      string
                                 updated_at:      string
                               }
                           }
                       }
      configurations:  ▸[]
    }
}
```

Code	Description	Schema

401	Unauthorized	⇄ ▾**errorDefault {**
		`error: string`
		}

404	Event not Found	⇄ ▾**errorDefault {**
		`error: string`
		}

500	Failed to retrieve Event	⇄ ▾**errorDefault {**
		`error: string`
		}

Try this operation

PUT /event/{event_key}

Event

Summary

Update a Event

Parameters

Name	Located in	Description	Required	Schema
event_key	path	Method Id	Yes	⇄ string
Body	body	Event Update Data	Yes	⇄
X-AUTH-TOKEN	header	User Auth Token	Yes	⇄ string
X-MODULE-TOKEN	header	Module Token	Yes	⇄ string
LANG-CODE	header	Language code	Yes	⇄ string
LANG-CODE-DEFAULT	header	Default Language code	Yes	⇄ string
TIMEZONE	header	Timezone	Yes	⇄ string

```
▾eventCreate {
   all of:
     ▾ {
         method_id:       integer
         start_date:      string
         end_date:        integer
         start_time:      string
         end_time:        string
         created_at:      string
         updated_at:      string
         configurations: ▸ { }
       }
  }
```

Responses

Code	Description	Schema

200 Update a Event ⇄

```
▾eventReply {
  all of:
    ▾ {
        key:        string
        name:       string
        method_id:  integer
        created_by: string
        start_date: string
        end_date:   string
        start_time: string
        end_time:   string
        created_at: string
        updated_at: string
    }
}
```

401 Unauthorized ⇄

```
▾errorDefault {
  error: string
}
```

500 Failed to update Event ⇄

```
▾errorDefault {
  error: string
}
```

Try this operation

/event

`POST /event`

Event

Summary

Create a Event

Parameters

Name	Located in	Description	Required	Schema
Body	body	Event Created Data	Yes	⇄

```
▾eventCreate {
   all of:
     ▾ {
         method_id:      integer
         start_date:     string
         end_date:       integer
         start_time:     string
         end_time:       string
         created_at:     string
         updated_at:     string
         configurations: ▸ { }
       }
 }
```

Name	Located in	Description	Required	Schema
X-AUTH-TOKEN	header	User Auth Token	Yes	⇄ `string`
X-MODULE-TOKEN	header	Module Token	Yes	⇄ `string`
LANG-CODE	header	Language code	Yes	⇄ `string`
LANG-CODE-DEFAULT	header	Default Language code	Yes	⇄ `string`
TIMEZONE	header	Timezone	Yes	⇄ `string`

Responses

Code	Description	Schema

200 Created Event ⇄

```
▾eventReply {
  all of:
    ▾ {
        key:        string
        name:       string
        method_id:  integer
        created_by: string
        start_date: string
        end_date:   string
        start_time: string
        end_time:   string
        created_at: string
        updated_at: string
      }
}
```

401 Unauthorized ⇄

```
▾errorDefault {
  error: string
}
```

500 Failed to update Method ⇄

```
▾errorDefault {
  error: string
}
```

Try this operation

/eventv/{event_key}

DELETE /eventv/{event_key}

Event

Summary

Delete a Method

Parameters

Name	Located in	Description	Required	Schema
event_key	path	Method Id	Yes	⇄ `string`
X-MODULE-TOKEN	header	Module Token	Yes	⇄ `string`
X-AUTH-TOKEN	header	User Auth Token	Yes	⇄ `string`

Responses

Code	Description	Schema

200	OK	⇄ ▾ **`eventDeleteReply {`**
		`string: string`
		`}`

401	Unauthorized	⇄ ▾ **`errorDefault {`**
		`error: string`
		`}`

404	Event not Found	⇄ ▾ **`errorDefault {`**
		`error: string`
		`}`

500	Failed to delete Event	⇄ ▾ **`errorDefault {`**
		`error: string`
		`}`

Try this operation

/generalConfigType/list

GET /generalConfigType/list

General Configuration Types

Summary

Show a General Configuration detail

Parameters

Name	Located in	Description	Required	Schema
X-MODULE-TOKEN	header	Module Token	Yes	⇄ string
X-ENTITY-KEY	header	Entity Token	Yes	⇄ string
LANG-CODE	header	Language code	Yes	⇄ string
LANG-CODE-DEFAULT	header	Default Language code	Yes	⇄ string

Responses

Code	Description	Schema
200	Show the Method data	⇄
401	Unauthorized	⇄
404	Method not Found	⇄
500	Failed to retrieve Method	⇄

200

```
▾generalConfigTypeListReply {
   all of:
      ▾ {
        data: ▾[
                ▾generalConfigTypeReply {
                   all of:
                      ▾ {
                        id:                     integer
                        general_config_type_key: string
                        code:                   string
                        name:                   string
                        created_at:             string
                        updated_at:             string
                      }
                 }
              ]
        }
}
```

401

```
▾errorDefault {
   error: string
}
```

404

```
▾errorDefault {
   error: string
}
```

500

```
▾errorDefault {
   error: string
}
```

Try this operation

/methodGroup/{method_group_id}

`GET /methodGroup/{method_group_id}`

Method Groups

Summary

Show a Method Group detail

Parameters

Name	Located in	Description	Required	Schema
method_group_id	path	Method Group id	Yes	⇄ `string`
X-MODULE-TOKEN	header	Module Token	Yes	⇄ `string`
X-ENTITY-KEY	header	Entity Token	Yes	⇄ `string`
LANG-CODE	header	Language code	Yes	⇄ `string`
LANG-CODE-DEFAULT	header	Default Language code	Yes	⇄ `string`

Responses

Code **Description** **Schema**

200 Show the Method Group data ⇄

```
▾methodGroupReply {
  all of:
    ▾ {
       id:          integer
       name:        string
       description: string
       created_at:  string
       updated_at:  string
     }
}
```

401 Unauthorized ⇄

```
▾errorDefault {
   error: string
}
```

404 Method Group not Found ⇄

```
▾errorDefault {
   error: string
}
```

500 Failed to retrieve Method Group ⇄

```
▾errorDefault {
   error: string
}
```

Try this operation

PUT /methodGroup/{method_group_id}

Method Groups

Summary

Update a Method Group

Parameters

Name	Located in	Description	Required	Schema

Body	body	Method Group Update Data	Yes	⇄ ▾methodGroupCreate {
				all of:
				▾ {
				translations: ▸ []
				}
				}

method_group_id	path	Method Group id	Yes	⇄ integer
X-AUTH-TOKEN	header	User Auth Token	Yes	⇄ string
X-MODULE-TOKEN	header	Module Token	Yes	⇄ string
LANG-CODE	header	Language code	Yes	⇄ string
LANG-CODE-DEFAULT	header	Default Language code	Yes	⇄ string

Responses

Code	Description	Schema

200	The updated Method Group	⇄ ▾methodGroupReply {
		all of:
		▾ {
		id: integer
		name: string
		description: string
		created_at: string
		updated_at: string
		}
		}

401	Unauthorized	⇄ ▾errorDefault {
		error: string
		}

404	Method Group not Found	⇄ ▾errorDefault {
		error: string
		}

500	Failed to create Method Group	⇄ ▾errorDefault {
		error: string
		}

Try this operation

`DELETE /methodGroup/{method_group_id}`

Method Groups

Summary

Delete a Method Group

Parameters

Name	Located in	Description	Required	Schema
method_group_id	path	Method Group Id	Yes	⇄ `integer`
X-MODULE-TOKEN	header	Module Token	Yes	⇄ `string`
X-AUTH-TOKEN	header	User Auth Token	Yes	⇄ `string`

Responses

Code	Description	Schema

200	OK	⇄ ▾**methodGroupDeleteReply {**
		string: string
		}

401	Unauthorized	⇄ ▾**errorDefault {**
		error: string
		}

404	Method Group not Found	⇄ ▾**errorDefault {**
		error: string
		}

500	Failed to delete Method Group	⇄ ▾**errorDefault {**
		error: string
		}

Try this operation

`/methodGroup`

`POST /methodGroup`

Method Groups

Summary

Create a Method Group

Parameters

Name	Located in	Description	Required	Schema

Body	body	Method Group Created Data	Yes	⇄ ▾**methodGroupCreate {**
				all of:
				▾ **{**
				translations: ▸ []
				}
				}

X-AUTH-TOKEN	header	User Auth Token	Yes	⇄ `string`
X-MODULE-TOKEN	header	Module Token	Yes	⇄ `string`
LANG-CODE	header	Language code	Yes	⇄ `string`
LANG-CODE-DEFAULT	header	Default Language code	Yes	⇄ `string`

Responses

Code	Description	Schema
200	Created Method Group	⇄ ▾**methodGroupReply {** **all of:** ▾ **{** id: integer name: string description: string created_at: string updated_at: string } }

401	Unauthorized	⇄ ▾**errorDefault {**
		error: string
		}

500	Failed to update Method Group	⇄ ▾**errorDefault {**
		error: string
		}

Try this operation

`/methodGroup/{method_group_id}/listMethods`

`GET /methodGroup/{method_group_id}/listMethods`

Method Groups

Summary

List of methods of Method group

Parameters

Name	Located in	Description	Required	Schema
method_group_id	path	Method Group id	Yes	⇄ `string`
X-MODULE-TOKEN	header	Module Token	Yes	⇄ `string`
X-ENTITY-KEY	header	Entity Token	Yes	⇄ `string`
LANG-CODE	header	Language code	Yes	⇄ `string`
LANG-CODE-DEFAULT	header	Default Language code	Yes	⇄ `string`

Responses

Code	Description	Schema

200	Show the Method Group data	⇄ ▾**methodMethodGroupListReply {**
		all of:
		▾ **{**
		data: ▾**[**
		▸**methodReply { }**
		]
		}
		}

401	Unauthorized	⇄ ▾**errorDefault {**
		error: string
		}

404	Method Group not Found	⇄ ▾**errorDefault {**
		error: string
		}

500	Failed to retrieve Method Group	⇄ ▾**errorDefault {**
		error: string
		}

Try this operation

`/method/{method_id}`

`GET /method/{method_id}`

Method

Summary

Show a Method detail

Parameters

Name	Located in	Description	Required	Schema
method_id	path	Method id	Yes	⇄ `string`
X-MODULE-TOKEN	header	Module Token	Yes	⇄ `string`
X-ENTITY-KEY	header	Entity Token	Yes	⇄ `string`
LANG-CODE	header	Language code	Yes	⇄ `string`
LANG-CODE-DEFAULT	header	Default Language code	Yes	⇄ `string`

Responses

Code	Description	Schema
200	Show the Method data	⇄

```
▾methodGroupReply {
  all of:
    ▾ {
       id:          integer
       name:        string
       description: string
       created_at:  string
       updated_at:  string
    }
}
```

Code	Description	Schema
401	Unauthorized	⇄

```
▾errorDefault {
  error: string
}
```

Code	Description	Schema
404	Method not Found	⇄

```
▾errorDefault {
  error: string
}
```

Code	Description	Schema
500	Failed to retrieve Method	⇄

```
▾errorDefault {
  error: string
}
```

Try this operation

PUT /method/{method_id}

Method

Summary

Update a Method

Parameters

Name	Located in	Description	Required	Schema
Body	body	Method Update Data	Yes	⇄

```
▾methodCreate {
  all of:
    ▾ {
       group_method_id: integer
                        *
       translations:    ▸[]
    }
}
```

Name	Located in	Description	Required	Schema
method_id	path	Method Group id	Yes	⇄ `integer`
X-AUTH-TOKEN	header	User Auth Token	Yes	⇄ `string`
X-MODULE-TOKEN	header	Module Token	Yes	⇄ `string`
LANG-CODE	header	Language code	Yes	⇄ `string`
LANG-CODE-DEFAULT	header	Default Language code	Yes	⇄ `string`

Responses

Code	Description	Schema
200	The updated Method	⇄

```
▾methodGroupReply {
  all of:
    ▾ {
       id:          integer
       name:        string
       description: string
       created_at:  string
       updated_at:  string
    }
}
```

Code	Description	Schema
401	Unauthorized	⇄

```
▾errorDefault {
  error: string
}
```

Code	Description	Schema
404	Method Group not Found	⇄

```
▾errorDefault {
  error: string
}
```

Code	Description	Schema

500	Failed to create Method	⇄ ▾**errorDefault {**
		error: string
		}

Try this operation

`DELETE /method/{method_id}`

Method

Summary

Delete a Method

Parameters

Name	Located in	Description	Required	Schema
method_id	path	Method Id	Yes	⇄ `integer`
X-MODULE-TOKEN	header	Module Token	Yes	⇄ `string`
X-AUTH-TOKEN	header	User Auth Token	Yes	⇄ `string`

Responses

Code	Description	Schema

200	OK	⇄ ▾**methodDeleteReply {**
		string: string
		}

401	Unauthorized	⇄ ▾**errorDefault {**
		error: string
		}

404	Method Group not Found	⇄ ▾**errorDefault {**
		error: string
		}

500	Failed to delete Method Group	⇄ ▾**errorDefault {**
		error: string
		}

Try this operation

`/method`

POST /method

Method

Summary

Create a Method

Parameters

Name	Located in	Description	Required	Schema
Body	body	Method Created Data	Yes	⇄

```
▾methodCreate {
  all of:
    ▾ {
      group_method_id: integer
                       *
      translations:     ▸[]
    }
}
```

Name	Located in	Description	Required	Schema
X-AUTH-TOKEN	header	User Auth Token	Yes	⇄ `string`
X-MODULE-TOKEN	header	Module Token	Yes	⇄ `string`
LANG-CODE	header	Language code	Yes	⇄ `string`
LANG-CODE-DEFAULT	header	Default Language code	Yes	⇄ `string`

Responses

Code	Description	Schema
200	Created Method	⇄

```
▾methodReply {
  all of:
    ▾ {
      id:              integer
      method_group_id: integer
      name:            string
      description:     string
      created_at:      string
      updated_at:      string
    }
}
```

Code	Description	Schema
401	Unauthorized	⇄

```
▾errorDefault {
  error: string
}
```

Code	Description	Schema
500	Failed to update Method	⇄

```
▾errorDefault {
  error: string
}
```

Try this operation

Models

translationsConfigs

```
▾translationsConfigs {
  all of:
    ▾ {
      language_code: string
      name:          string
      description:   string
    }
}
```

configurationReply

```
▾configurationReply {
  all of:
    ▾ {
      id:             integer
      method_id:      integer
      code:           string
      parameter_type: string
      name:           string
      description:    string
      created_at:     string
      updated_at:     string
    }
}
```

translationsConfiguration

```
▾translationsConfiguration {
  all of:
    ▾ {
      language_code: string
      name:          string
      description:   string
    }
}
```

configurationCreate

```
▾configurationCreate {
  all of:
    ▾ {
      method_id:      integer *
      code:           string
      parameter_type: string
      translations:   ▾[
                         ▾translationsConfiguration {
                           all of:
                             ▾ {
                               language_code: string
                               name:          string
                               description:   string
                             }
                         }
                       ]
    }
}
```

configurationDeleteReply

```
▾configurationDeleteReply {
   string: string
}
```

errorDefault

```
▾errorDefault {
   error: string
}
```

eventErrorDefault

```
▾eventErrorDefault {
   error: string
}
```

eventMethodReply

```
▾eventMethodReply {
  all of:
    ▾ {
      id:              string
      code:            string
      method_group_id: integer
      name:            string
      description:     string
      created_at:      string
      updated_at:      string
    }
}
```

eventConfigurationsReply

```
▾eventConfigurationsReply {
  all of:
    ▾ {
      id:                      integer
      configuration_event_key: string
      configuration_id:        integer
      event_id:                integer
      general_config_id:       integer
      value:                   string
      name:                    string
      parameter_type:          string
      configuration_code:      string
      created_at:              string
      updated_at:              string
    }
}
```

eventShowReply

```
▾eventShowReply {
  all of:
    ▾ {
      key:            string
      name:           string
      method_id:      integer
      created_by:     string
      start_date:     string
      end_date:       string
      start_time:     string
      end_time:       string
      created_at:     string
      updated_at:     string
      method:         ▾ {
                        all of:
                          ▾eventMethodReply {
                            all of:
                              ▾ {
                                id:              string
                                code:            string
                                method_group_id: integer
                                name:            string
                                description:     string
                                created_at:      string
                                updated_at:      string
                              }
                          }
                      }
```

```
        configurations: ▾[
                          ▾eventConfigurationsReply {
                            all of:
                              ▾ {
                                  id:                        integer
                                  configuration_event_key: string
                                  configuration_id:         integer
                                  event_id:                  integer
                                  general_config_id:        integer
                                  value:                     string
                                  name:                      string
                                  parameter_type:           string
                                  configuration_code:       string
                                  created_at:               string
                                  updated_at:               string
                              }
                          }
                        ]
    }
}
```

eventConfigurationsCreate

```
▾eventConfigurationsCreate {
  all of:
    ▾ {
⇄       configuration_id: integer
        value:            string
    }
}
```

eventCreate

```
▾eventCreate {
  all of:
    ▾ {
      method_id:      integer
      start_date:     string
      end_date:       integer
      start_time:     string
      end_time:       string
      created_at:     string
      updated_at:     string
      configurations: ▾ {
                        all of:
                          ▾eventConfigurationsCreate {
                            all of:
                              ▾ {
                                configuration_id: integer
                                value:            string
                              }
                          }
                      }
    }
}
```

eventReply

```
▾eventReply {
  all of:
    ▾ {
      key:        string
      name:       string
      method_id:  integer
      created_by: string
      start_date: string
      end_date:   string
      start_time: string
      end_time:   string
      created_at: string
      updated_at: string
    }
}
```

eventDeleteReply

```
▾eventDeleteReply {
  string: string
}
```

generalConfigTypeReply

```
   ▾generalConfigTypeReply {
     all of:
       ▾ {
         id:                      integer
         general_config_type_key: string
⇄        code:                    string
         name:                    string
         created_at:              string
         updated_at:              string
       }
   }
```

generalConfigTypeListReply

```
   ▾generalConfigTypeListReply {
     all of:
       ▾ {
         data: ▾[
                 ▾generalConfigTypeReply {
                   all of:
                     ▾ {
                       id:                      integer
                       general_config_type_key: string
⇄                      code:                    string
                       name:                    string
                       created_at:              string
                       updated_at:              string
                     }
                 }
               ]
       }
   }
```

translationsMethodGroup

```
   ▾translationsMethodGroup {
     all of:
       ▾ {
         language_code: string
⇄        name:          string
         description:   string
       }
   }
```

methodGroupCreate

```
▾methodGroupCreate {
  all of:
    ▾ {
      translations: ▾[
                      ▾translationsMethodGroup {
                        all of:
                          ▾ {
                            language_code: string
                            name:          string
                            description:   string
                          }
                      }
                    ]
    }
}
```

methodGroupReply

```
▾methodGroupReply {
  all of:
    ▾ {
      id:          integer
      name:        string
      description: string
      created_at:  string
      updated_at:  string
    }
}
```

methodMethodGroupListReply

```
▾methodMethodGroupListReply {
  all of:
    ▾ {
      data: ▾[
              ▾methodReply {
                all of:
                  ▾ {
                    id:              integer
                    method_group_id: integer
                    name:            string
                    description:     string
                    created_at:      string
                    updated_at:      string
                  }
              }
            ]
    }
}
```

methodGroupDeleteReply

```
▾methodGroupDeleteReply {
   string: string
}
```

translationsMethod

```
▾translationsMethod {
  all of:
    ▾ {
       language_code: string
       name:          string
       description:   string
    }
}
```

methodCreate

```
▾methodCreate {
  all of:
    ▾ {
       group_method_id: integer *
       translations:    ▾[
                          ▾translationsMethod {
                            all of:
                              ▾ {
                                 language_code: string
                                 name:          string
                                 description:   string
                              }
                          }
                        ]
    }
}
```

methodReply

```
▾methodReply {
  all of:
    ▾ {
       id:              integer
       method_group_id: integer
       name:            string
       description:     string
       created_at:      string
       updated_at:      string
    }
}
```

methodDeleteReply

```
▾methodDeleteReply {
   string: string
}
```