

A7: Web Resources Documentation

The architecture of the web application to develop is documented indicating the catalogue of resources and the properties of each resource, including: references to the graphical interfaces, and the format of JSON responses. This page presents a documentation for Presto, including the following operations over data: create, read, update, and delete.

1. Overview

An overview of the web application to implement is presented in this section, where the modules are identified and briefly described. The web resources associated with each module are detailed in the individual documentation of each module.

M01: Authentication and Static Pages	Web resources associated with user authentication, includes the following system features: login/logout and registration. Web resources with static content are associated with this module: about and 404.
M02: Members	Web resources associated with members, such as individual profile management: view and edit personal information, manage their followers/following, their settings. Furthermore, this module also includes the notifications management.
M03: Content (Questions, Answers and Comments)	Web resources associated with content management, such as rating or reporting, creating new content, viewing specific content, editing or deleting it.
M04: Topics	Web resources associated with the topics (categories questions are associated with), includes creating, viewing, editing and deleting topics.
M05: Index and Search	Web resources associated with the main index and search. This module includes the main feed and search features.
M06: Administration	Web resources associates with member management, specifically: view and search members, ban member accounts, promote members to moderators and view and manage flagged users.

2. Permissions

This section defines the permissions used in the modules to establish the conditions of access to resources.

PUB	Public	Group of users without privileges
MEM	Member	Group of authenticated users that can create and search for content
OWN	Owner	Group of members that can update their profiles and have privileges regarding their content.
MOD	Moderator	Group of members that manage the site's content.
ADM	Administrator	System administrator

3. Modules

This section documents each web resource of each module, indicating the URL, HTTP methods, the possible parameters, interfaces with the user (referring to the A3 artefact) — or JSON responses in the event of APIs.

3.1. Module M01: Authentication and Static Pages

These are the endpoints available in the Authentication and Static Pages.

- **R101: Login Form** /login
- **R102: Login Action** /login
- **R103: Logout Action** /logout
- **R104: Sign up Form** /signup
- **R105: Sign up Action** /signup
- **R106: About** /about
- **R107: 404** /404

R101: Login Form

URL	/login
Description	Page with a form to login to a user account.
Method	GET
UI	UI02
SUBMIT	R102
Permissions	PUB

R102: Login Action

URL	/login	
Description	This web resource logs the user into the system. Redirects to the member's feed or to the admin page on success and the login form on failure.	
Method	POST	
Request Body	+email:string	Email
	+password:string	Password
Redirects	R502	Success
	R601	Success, if admin
	R101	Error
Permissions	PUB	

R103: Logout Action

URL	/logout	
Description	This web resource logs out the member or admin.	
Method	POST	
Redirects	R101	
Permissions	MEM or ADM	

R104: Sign up Form

URL	/signup	
Description	Page with a form to register a new member account.	
Method	GET	
UI	UI01	
SUBMIT	R105	
Permissions	PUB	

R105: Sign up Action

URL	/signup	
Description	This web resource inserts the user into the system. Redirects to the member's feed on success and the sign up form on failure.	
Method	POST	
Request Body	+username:string	Username
	+email:string	Email
	+password:string	Password
Redirects	R502	Success
	R601	Success, if admin
	R104	Error
Permissions	PUB	

R106: About

URL	/about	
Description	Get about page.	
Method	GET	
UI	UI03	
Permissions	PUB	
Observations	Obs1	

R107: 404

URL	/404	
Description	Get 404 page.	
Method	GET	
UI	UI18	
Permissions	PUB	

3.2. Module M02: Members

These are the endpoints available in the Users Module.

- **R201: View Profile** /profile/{username}
- **R202: Edit Profile Form** /profile/edit
- **R203: Get Edit Profile Form API** /api/members/{username}/edit-profile
- **R204: Edit Profile Action** /profile/{username}
- **R205: View Followers** /profile/{username}/followers
- **R206: View Following** /profile/{username}/following
- **R207: Follow/unfollow Member API** /api/member/{username}/toggle-follow
- **R208: View Settings** /settings
- **R209: Get Settings API** /api/members/{username}/settings
- **R210: Edit E-mail API** /api/members/{username}/settings/email
- **R211: Edit Password API** /api/members/{username}/settings/password
- **R212: Edit Google Account API** /api/members/{username}/settings/google-account
- **R213: Edit Facebook Account API**
/api/members/{username}/settings/facebook-account
- **R214: View Notifications** /notifications
- **R215: Get Notifications API** /api/members/{username}/notifications
- **R216: Mark a Notification as Read API**
/api/members/{username}/notifications/{notification_id}
- **R217: Flag Member API** /api/members/{username}/flag

R201: View Profile

URL	/profile/{username}	
Description	Page with a member's profile info.	
Method	GET	
Parameters	+username:string	Member's username (unique key)
UI	UI11	
UI Observations	UI changes according to permissions. If member has OWN permissions, editing options are shown. If member has MOD permissions, member flagging option should be displayed on all profiles that are not one's own.	
Permissions	PUB (general access) or OWN (for editing) or MOD (for flagging)	
Observations	Obs1	

R202: Edit Profile Form

URL	/profile/edit
Description	Page with a form to edit one's profile info.
Method	GET
UI	UI05
Ajax Calls	R203
SUBMIT	R204
Permissions	OWN

R203: Get Edit Profile Form API

URL	/api/members/{username}/edit-profile	
Description	This resource gets member's information to use as default values in the edit form.	
Method	GET	
Parameters	+username:string	Member's username (unique key)
Response Body	JSON203	
Permissions	OWN	

R204: Edit Profile Action

URL	/profile/{username}	
Description	This web resource changes user profile info based on the input received. Redirects to the member profile on success and edit profile page on failure.	
Method	PUT	
Parameters	+username: string	Member's username (unique key)
Request Body	+name:string	New name
	+username:string	New username

	?bio:string	New bio
	?profile_pic:string	New profile picture
Redirects	R201	Success
	R202	Error
Permissions	OWN	

R205: View Followers

URL	/profile/{username}/followers	
Description	Page with the member's followers.	
Method	GET	
Parameters	+username:string	Target member's username (unique key)
UI	UI06	
Permissions	PUB	
Observations	Obs1	

R206: View Following

URL	/profile/{username}/following	
Description	Page with the members that another member follows.	
Method	GET	
Parameters	+username:string	Target member's username (unique key)
UI	UI07	
Permissions	PUB	
Observations	Obs1	

R207: Follow/unfollow Member API

URL	/api/member/{username}/toggle-follow	
Description	This web resource toggles following of another member.	
Method	PUT	
Request body	+username:string	Target member's username (unique key)
Returns	200 OK	The following status was successfully updated.
	400 Bad Request	Error. Error message is specified via a HTTP header.
	404 Not Found	Error. No member with the specified key exists.
Permissions	OWN	

R208: View Settings

URL	/settings	
Description	Page with the member's settings.	
Method	GET	
UI	UI12	
AJAX Calls	R209	R212
	R210	R213
	R211	
Permissions	OWN	
Observations	Obs1	

R209: Get Settings API

URL	/api/members/{username}/settings	
Description	Obtains a member's current settings	
Method	GET	
Parameters	+username:string	Member's username (unique key)
Response body	JSON209	
Permissions	OWN	

R210: Edit E-mail API

URL	/api/members/{username}/settings/email	
Description	This resource updates the member's e-mail account.	
Method	PUT	
Parameters	+username:string	Member's username (unique key)
Request Body	+e-mail:string	New e-mail
Returns	200 OK	The e-mail was successfully updated.
	400 Bad Request	Error. Error message is specified via a HTTP header.
	404 Not Found	Error. No member with the specified key exists.
Response body	JSON210	
Permissions	OWN	

R211: Edit Password API

URL	/api/members/{username}/settings/password	
Description	This resource updates the member's password.	
Method	PUT	
Parameters	+username:string	Member's username (unique key)
Request Body	+password:string	New password
Returns	200 OK	The password was successfully updated.
	400 Bad Request	Error. Error message is specified via a HTTP header.
	404 Not Found	Error. No member with the specified key exists.
Permissions	OWN	

R212: Edit Google Account API

URL	/api/members/{username}/settings/google-account	
Description	This resource updates the member's connected google account.	
Method	PUT	
Parameters	+username:string	Member's username (unique key)
Request Body	+account:string	New google account.
Returns	200 OK	The account was successfully updated.
	400 Bad Request	Error. Error message is specified via a HTTP header.
	404 Not Found	Error. No member with the specified key exists.
Permissions	OWN	

R213: Edit Facebook Account API

URL	/api/members/{username}/settings/facebook-account	
Description	This resource updates the member's connected facebook account..	
Method	PUT	
Parameters	+username:string	Member's username (unique key)
Request Body	+account:string	New facebook account
Returns	200 OK	The account was successfully updated.
	400 Bad Request	Error. Error message is specified via a HTTP header.
	404 Not Found	Error. No member with the specified key exists.
Permissions	OWN	

R214: View Notifications

URL	/notifications
Description	Page with the member's notifications.
Method	GET
UI	UI10
AJAX Calls	R215
Permissions	OWN
Observations	Obs1

R215: Get Notifications API

URL	/api/members/{username}/notifications	
Description	This resource gets the member's notifications.	
Method	GET	
Parameters	+username:string	Member's username (unique key)
Response body	JSON215	
Permissions	OWN	

R216: Mark a Notification as Read API

URL	/api/members/{username}/notifications/{notification_id}	
Description	This web resource marks a notification as read.	
Method	PUT	
Parameter	+username:string	Member's username (unique key)
	+notification_id:integer	Notification's primary key
Returns	200 OK	The notification was successfully updated.
	400 Bad Request	Error. Error message is specified via a HTTP header.
	404 Not Found	Error. No member or notification with the specified primary key exists.
Permissions	OWN	

R217: Flag Member API

URL	/api/members/{username}/flag	
Description	This web resource flags a member.	
Method	PUT	
Parameter	+username:string	Flagged member's username (unique key)
Request Body	+mod_username:string	Mod's username (unique key)
	+reason: string	Reason for flagging user
Returns	200 OK	The member was successfully flagged.
	400 Bad Request	Error. Error message is specified via a HTTP header.
	404 Not Found	Error. No member or moderator with the specified key exists.
Permissions	MOD	

3.3. Module M03: Content (Questions, Answers and Comments)

These are the endpoints available in the Content Module.

- **R301: View Question** /questions/{id}
- **R302: Get Question API** /api/questions/{id}
- **R303: Add Question Action** /questions
- **R304: Edit Question API** /api/questions/{id}
- **R305: Delete Question API** /api/questions/{id}
- **R306: Close/open Question API** /api/questions/{id}/toggle-closed
- **R307: Rate Question API** /api/questions/{id}/rate
- **R308: Report Action API** /api/questions/{id}/report
- **R309: View Answer** /questions/{question_id}/answers/{answer_id}
- **R310: Get Answer API** /api/questions/{question_id}/answers/{answer_id}
- **R311: Add Answer API** /api/questions/{question_id}/answers
- **R312: Edit Answer API** /api/questions/{question_id}/answers/{answer_id}
- **R313: Delete Answer API** /api/questions/{question_id}/answers/{answer_id}
- **R314: Rate API** /api/questions/{question_id}/answers/{answer_id}/rate
- **R315: Report API** /api/questions/{question_id}/answers/{answer_id}/report
- **R316: Get Comment API** /api/comments/{id}
- **R317: Add Comment API** /api/comments

- **R318: Edit Comment API** /api/comments/{id}
- **R319: Delete Comment API** /api/comments/{id}
- **R320: Rate Comment API** /api/comments/{id}/rate
- **R321: Report Action API** /api/comments/{id}/report

R301: View Question

URL	/questions/{id}	
Description	Displays the question's page.	
Method	GET	
Parameters	+id:integer	Question primary key
AJAX Calls	R207	R311
	R302	R312
	R304	R313
	R305	R314
	R306	R315
	R307	R316
	R308	R317
	R309	R318
	R310	R319
	R320	
UI	UI13	
UI Observations	<p>UI changes according to permissions. If viewer has PUB permissions, no content options should be displayed. If MEM, report and rating options. If OWN, editing options only for the member's content. If MOD, editing options for all of the content and flagging options.</p> <p>If the question is closed, a "closed" tag is added and the options to add content are blocked.</p>	
Permissions	PUB (general access) or MEM (for rating and reporting content) or OWN (for editing one's content) or MOD (for editing any content and/or flagging)	
Observations	Obs1	

R302: Get Question API

URL	/api/questions/{id}	
Description	This web resource gets a question.	
Method	GET	
Parameters	+id:integer	Question primary key
Response Body	JSON302	The question's information when successful, an error message when unsuccessful
Permissions	MEM	

R303: Add Question Action

URL	/questions	
Description	This web resource adds a new question.	
Method	POST	
Request Body	+author_id:integer	Member primary key
	+title:string	Title
	+content:string	Content
Redirects	R301	Success
	R502	Error
Permissions	MEM	

R304: Edit Question API

URL	/api/questions/{id}	
Description	This web resource updates a question.	
Method	PUT	
Parameter	+id:integer	Question primary key
Request Body	+title:string	Title
	+content:string	Content
Response body	JSON304	The question's information when successful, an error message when unsuccessful
Permissions	OWN or MOD	

R305: Delete Question API

URL	/api/questions/{id}	
Description	This web resource deletes a question.	
Method	DELETE	
Parameters	+id:integer	Question primary key
Returns	200 OK	The question was successfully deleted.
	400 Bad Request	Error. Error message is specified via a HTTP header.
	404 Not Found	Error. No question with the specified primary key exists.
Redirects	R502	Success
Permissions	OWN or MOD	

R306: Close/open Question API

URL	/api/questions/{id}/toggle-closed	
Description	This web resource closes or opens a question.	
Method	PUT	
Parameters	+id:integer	Question primary key
Returns	200 OK	The question status was successfully updated.
	400 Bad Request	Error. Error message is specified via a HTTP header.
	404 Not Found	Error. No question with the specified primary key exists.
Permissions	OWN or MOD	

R307: Rate Question API

URL	/api/questions/{id}/rate	
Description	This web resource rates a question.	
Method	PUT	
Parameters	+id:integer	Question primary key
Request Body	+member_id:integer	Member primary key
	+rating:integer	Rating
Returns	200 OK	The question was successfully rated.
	400 Bad Request	Error. Error message is specified via a HTTP header.
	404 Not Found	Error. No question with the specified primary key exists.
Permissions	MEM	

R308: Report Question API

URL	/api/questions/{id}/report	
Description	This web resource reports a question.	
Method	POST	
Parameters	+id:integer	Question primary key
Request Body	+member_id:integer	Member primary key
	+reason:string	Reason
Returns	200 OK	The question was successfully reported.
	400 Bad Request	Error. Error message is specified via a HTTP header.
	404 Not Found	Error. No question with the specified primary key exists.
Permissions	MEM	

R309: View Answer

URL	/questions{question_id}/answers/{answer_id}	
Description	Displays the answer's page.	
Method	GET	
Parameters	+question_id:integer	Question primary key
	+answer_id:integer	Answer primary key
AJAX Calls	R207	R316
	R310	R317
	R312	R318
	R313	R319
	R314	R320
	R315	
UI	UI16	

UI Observations	UI changes according to permissions. If the viewer has PUB permissions, no content options should be displayed. If MEM, only rating options. If OWN, editing options only for the member's content. If MOD, editing options for all of the content and flagging options.
Permissions	PUB (general access) or MEM (for rating and/or reporting options) or OWN (for editing one's options) or MOD (for editing any content and/or flagging options)
Observations	Obs1

R310: Get Answer API

URL	/api/questions/{question_id}/answers/{answer_id}	
Description	This web resource gets an answer.	
Method	GET	
Parameters	+question_id:integer	Question primary key
	+answer_id:integer	Answer primary key
Response Body	JSON310	The answer's information when successful, an error message when unsuccessful
Permissions	MEM	

R311: Add Answer API

URL	/api/questions/{question_id}/answers	
Description	This web resource adds a new answer.	
Method	POST	
Parameters	+question_id:integer	Question primary key
Request Body	+author_id:integer	Member primary key
	+content:string	Content
Response Body	JSON311	The answer's information when successful, an error message when unsuccessful
Permissions	MEM	

R312: Edit Answer API

URL	/api/questions/{question_id}/answers/{answer_id}	
Description	This web resource updates an answer.	
Method	PUT	
Parameters	+question_id:integer	Question primary key
Request Body	+answer_id:integer	Answer primary key
	+content:string	Content
Response Body	JSON312	The answer's information when successful, an error message when unsuccessful
Permissions	OWN or MOD	

R313: Delete Answer API

URL	/api/questions/{question_id}/answers/{answer_id}	
Description	This web resource deletes an answer.	
Method	DELETE	
Parameters	+question_id:integer	Question primary key
	+answer_id:integer	Answer primary key
Returns	200 OK	The answer was successfully deleted.
	400 Bad Request	Error. Error message is specified via a HTTP header.
	404 Not Found	Error. No answer with the specified primary key exists.
Permissions	OWN or MOD	

R314: Rate Answer API

URL	/api/questions/{question_id}/answers/{answer_id}/rate	
Description	This web resource rates an answer .	
Method	PUT	
Parameters	+question_id:integer	Question primary key
	+answer_id:integer	Answer primary key
Request Body	+member_id:integer	Member primary key
	+rating:integer	Rating
Returns	200 OK	The answer was successfully rated.
	400 Bad Request	Error. Error message is specified via a HTTP header.
	404 Not Found	Error. No answer with the specified primary key exists.
Permissions	MEM	

R315: Report Answer API

URL	/api/questions/{question_id}/answers/{answer_id}/report	
Description	This web resource reports an answer .	
Method	PUT	
Parameters	+question_id:integer	Question primary key
	+answer_id:integer	Answer primary key
Request Body	+member_id:integer	Member primary key
	+reason:string	Reason
Returns	200 OK	The answer was successfully reported.
	400 Bad Request	Error. Error message is specified via a HTTP header.
	404 Not Found	Error. No answer with the specified primary key exists.

Permissions	MEM
--------------------	-----

R316: Get Comment API

URL	/api/comments/{id}	
Description	This web resource gets a comment.	
Method	GET	
Parameters	+id:integer	Comment primary key
Response Body	JSON316	The comment's information when successful, an error message when unsuccessful
Permissions	MEM	

R317: Add Comment API

URL	/api/comments	
Description	This web resource adds a new comment.	
Method	POST	
Request Body	+author_id:integer	Member primary key
	+question_id	Question primary key
	+answer_id	Answer primary key
	+content:string	Content
Response Body	JSON317	The comment's information when successful, an error message when unsuccessful
Permissions	MEM	

R318: Edit Comment API

URL	/api/comments/{id}	
Description	This web resource updates a comment.	
Method	PUT	
Parameters	+id:integer	Comment primary key
Request Body	+content:string	Content
Response Body	JSON318	The comment's information when successful, an error message when unsuccessful
Permissions	OWN or MOD	

R319: Delete Comment API

URL	/api/comments/{id}	
Description	This web resource deletes a comment.	
Method	DELETE	
Parameters	+id:integer	Comment primary key
Returns	200 OK	The comment was successfully deleted.
	400 Bad Request	Error. Error message is specified via a HTTP header.
	404 Not Found	Error. No comment with the specified primary key exists.
Permissions	OWN or MOD	

R320: Rate Comment API

URL	/api/comments/{id}/rate	
Description	This web resource rates a comment.	
Method	PUT	
Parameters	+id:integer	Comment primary key
Request Body	+member_id:integer	Member primary key
	+rating:integer	Rating
Returns	200 OK	The comment was successfully rated.
	400 Bad Request	Error. Error message is specified via a HTTP header.
	404 Not Found	Error. No comment with the specified primary key exists.
Permissions	MEM	

R321: Report Comment API

URL	/api/comments/{id}/report	
Description	This web resource reports a comment.	
Method	PUT	
Parameters	+id:integer	Comment primary key
Request Body	+member_id:integer	Member primary key
	+reason:string	Reason
Returns	200 OK	The comment was successfully reported.
	400 Bad Request	Error. Error message is specified via a HTTP header.
	404 Not Found	Error. No comment with the specified primary key exists.
Permissions	MEM	

3.4. Module M04: Topics

These are the endpoints available in the Topics Module.

- **R401: View Topic** /topics/{name}
- **R402: Get Topic API** /api/topics/{name}
- **R403: Add Topic API** /api/topics
- **R404: Edit Topic Form** /topics/{name}/edit
- **R405: Get Edit Topic Form API** /api/topics/{name}/edit-topic
- **R406: Edit Topic Action** /topics/{name}
- **R407: Delete Topic Action** /topics/{name}
- **R408: Follow/unfollow Topic API** /api//topics/{name}/toggle-follow

R401: View Topic

URL	/topics/{name}	
Description	Displays the topic's page.	
Method	GET	
Parameters	+name:String	Topic name
AJAX Calls	R404	
UI	UI17	
UI Observations	UI changes according to permissions. If the member has MOD permissions, topic deletion and editing options are displayed.	
Permissions	PUB (general access) or MOD (for editing)	
Observations	Obs1	

R402: Get Topic API

URL	/api/topics/{name}	
Description	This web resource gets a topic.	
Method	GET	
Parameters	+name:String	Topic Name
Response Body	JSON402	The topic's information when successful, an error message when unsuccessful
Permissions	MEM	

R403: Add Topic API

URL	/api/topics	
Description	This web resource add a new topic.	
Method	POST	
Request Body	+name:string	Topic name
Response Body	JSON403	The topic's information when successful, an error message when unsuccessful
Permissions	MEM	

R404: Edit Topic Form

URL	/topics/{name}/edit	
Description	Page with a form to edit a topic.	
Method	GET	
Parameters	+name:String	Topic name
UI	UI19 (not in A3 originally)	
SUBMIT	R406	
Permissions	MOD	

R405: Get Edit Topic Form API

URL	/api/topics/{name}/edit-topic	
Description	This resource gets topic's information to use as default values in the edit form.	
Method	GET	
Parameters	+name:String	Topic name
Response Body	JSON405	
Permissions	MOD	

R406: Edit Topic Action

URL	/topics/{name}	
Description	This web resource updates the topic. Redirects to the topic page on success and the edit topic form on failure.	
Method	PUT	
Request Body	+name:string	Name
	?description:string	Description
	?image:string	Image url
Redirects	R401	Success
	R404	Error
Permissions	MOD	

R407: Delete Topic Action

URL	/api/questions/{name}	
Description	This web resource deletes a topic. Redirects to the moderator's feed on success and to the topic's page on failure.	
Method	DELETE	
Parameters	+name:String	Topic name
Redirects	R502	Success
	R401	Error
Permissions	MOD	

R408: Follow/unfollow Topic API

URL	/api/topics/{name}/toggle-follow	
Description	This web resource toggles following a topic.	
Method	PUT	
Parameter	+name:String	Topic name
Returns	200 OK	The topic was successfully updated.
	400 Bad Request	Error. Error message is specified via a HTTP header.
	404 Not Found	Error. No topic with the specified primary key exists.
Permissions	MEM	

3.5. Module M05: Index and Search

These are the endpoints available in the Index and Search Module.

- **R501: Index** /index
- **R502: View Feed** /index
- **R503: Get Feed API** /api/members/{username}/feed
- **R504: Search Page** /search
- **R505: Search API** /api/search

R501: Index

URL	/index
Description	Page with initial information.
Method	GET
UI	UI08
Permissions	PUB

R502: View Feed

URL	/index
Description	Page with the member's personal feed.
Method	GET
UI	UI09
AJAX Calls	R207
	R503
Permissions	MEM
Observations	Obs1

R503: Get Feed API

URL	/api/members/{username}/feed	
Description	This resource gets the member's personal feed.	
Method	GET	
Parameters	+username:string	Member's username (unique key)
Response Body	JSON503	
Permissions	MEM	

R504: Search Page

URL	/search
Description	Page to search all the content.
Method	GET
UI	UI14
UI Observations	When result type (question, answer, ...) filter is used, the author filter is blocked; When filtering users, the time filter is blocked
AJAX Calls	R505
Permissions	PUB

R505: Search API

URL	/api/search	
Description	This web resource searches the system for all content that matches the search query and filters it accordingly.	
Method	GET	
Request Body	+query:string	Search query.
	?types:string	Types of content to search.
	?author:string	Author of the content.
	?time:integer	Integer with the time boundaries to be considered.
Response Body	JSON505	
Permissions	PUB	

3.6. Module M06: Administration

These are the endpoints available in the Administrators Module.

- **R601: Admin Page** /admin
- **R602: Search for Members API** /api/members
- **R603: Ban Member API** /api/members/{username}/ban
- **R604: Promote Member API** /api/members/{username}/promote
- **R605: Dismiss Moderator API** /api/members/{username}/dismiss

R601: Admin Page

URL	/admin	
Description	Get admin page.	
Method	GET	
UI	UI04	
AJAX Calls	R602	
	R603	
	R604	
	R605	
Permissions	ADM	

R602: Search for Members API

URL	/api/members	
Description	This web resource searches the members whose data matches the search query.	
Method	GET	
Request Body	+query:string	Search query
Response Body	JSON602	
Permissions	ADM	

R603: Ban Member API

URL	/api/members/{username}/ban	
Description	This web resource bans a member.	
Method	POST	
Parameter	+username:string	Member's username (unique key)
Returns	200 OK	The member was successfully banned.
	400 Bad Request	Error. Error message is specified via a HTTP header.
	404 Not Found	Error. No member with the specified primary key exists.
Permissions	ADM	

R604: Promote Member API

URL	/api/members/{username}/promote	
Description	This web resource promotes a member to acquire moderator privileges.	
Method	PUT	
Parameter	+username:string	Member's username (unique key)
Returns	200 OK	The member was successfully promoted.
	400 Bad Request	Error. Error message is specified via a HTTP header.
	404 Not Found	Error. No member with the specified primary key exists.
Permissions	ADM	

R605: Dismiss Moderator API

URL	/api/members/{username}/promote	
Description	This web resource removes the moderator privileges of a member.	
Method	PUT	
Parameter	+username:string	Member's username (unique key)
Returns	200 OK	The member (moderator) was successfully dismissed.
	400 Bad Request	Error. Error message is specified via a HTTP header.
	404 Not Found	Error. No member with the specified primary key exists.
Permissions	ADM	

Observations:

- **Obs1:** There exists a modal on these pages containing a form to add a question for members (MEM) that when submitted triggers the add question action (R303).

4. JSON/XML Types

JSON203 - Edit profile (default values for form)

```
{
  "data":
    {
      "id": 0,
      "username": cyrilico,
      "name": Cyril Damas,
      "bio" : "I'm a Cyril boy, in a Cyril world",
      "profile_picture": "https://dummyimage.com/300"
    },
  "message": "Example error message when editing a profile"
}
```

JSON209 - Get current member settings

```
{
  "data":
    {
      "id": 0, //User id
      "email" : "dummy@dummy.org",
      "google_account": "myfacebookaccounthash",
      "facebook_account": "mygoogleaccounthash",
    },
  "message": "Example error message when getting a member settings"
}
```

JSON210 - Edit email

```
{
  "data":
    {
      "email": dummy@dummy.org
    },
  "message": "Example error message when editing an email"
}
```

JSON215 - Get notifications

```
{
  "notifications": [
    {
      "id": 0,
      "type": "Answer",
      "date": "2017-04-23 01:57:00",
      "content": "Cyril Damas answered your question: 'What is love?'",
      "read": true
    },
    {
      "id": 1,
      "type": "Follow",
      "date": "2017-05-31 02:35:13",
      "content": "Diogo Torres followed you",
      "read": false
    },
    {
      "id": 2,
```

```

    "type": "Comment",
    "date": "2018-01-01 05:00:00",
    "content": "Bruno Piedade left a comment on your question: 'How to do this?'",
    "read": false
  }
],
"message": "Example error message when getting notifications"
}

```

JSON302 - Get a question, **JSON303** - Add question, **JSON304** - Edit question

```

{
  "data":
    {
      "id": 4,
      "author_id": 1,
      "author_name" : "Joana Lima",
      "title" : "What is 1+2?",
      "content": "Seriously, I can't figure it out",
      "date": "2018-01-01 05:00:00",
      "views": 1000,
      "upvotes": 200,
      "downvotes": 100,
      "solved": false,
      "topics": [
        "Maths",
        "Science"
      ]
    },
  "message": "Example error message when getting a question"
}

```

JSON310 - Get an answer, **JSON311** - Add answer, **JSON312** - Edit answer

```

{
  "data":
    {
      "id": 4,
      "author_id": 1,
      "author_name" : "Joana Lima",
      "question_id" : 4,

```

```

        "content": "1+2 is obviously four.",
        "date": "2018-01-02 12:35:01",
        "views": 1000,
        "upvotes": 200,
        "downvotes": 100
    },
    "message": "Example error message when getting/adding/editing an answer"
}

```

JSON316 - Get a comment, **JSON317** - Add comment, **JSON318** - Edit comment

```

{
  "data":
    {
      "id": 4,
      "author_id": 1,
      "author_name" : "Joana Lima",
      "question_id" : null,
      "answer_id" : 4,
      "content": "I'm sorry but that is wrong. 1+2 is 42",
      "date": "2018-01-03 20:34:30",
      "upvotes": 200,
      "downvotes": 100
    },
  "message": "Example error message when getting/adding/editing a comment"
}

```

JSON402 - Get a topic, **JSON403** - Add topic, **JSON405** - Edit topic form

```

{
  "data":
    {
      "id": 4,
      "name": "History",
      "description": "The study of the past as it is described in
written documents."
      "picture" : "default.png",
      "views": 100000,
      "followers": 200,
      "questions": 100,
      "answers": 1000
    },
  "message": "Example error message when getting/adding/editing a topic"
}

```

JSON602 - Get members whose username matches input

```
{
  "data":
    {
      "id": 0,
      "username": cyrillico,
      "name": Cyril Damas
    }
  "message": "Example error message getting matching user"
}
```

JSON 503 - Feed

```
{
  "feed":
    {
      "questions" : [
        {"id": 4,
          "author_id": 1,
          "author_name" : "Joana Lima",
          "content": "Example content of the edited content itself",
          "date": "25-03-2012",
          "views": 1000,
          "upvotes": 200,
          "downvotes": 100,
          "solved": false,
          "topics": [
            "Maths",
            "Science"
          ]
        },
        {"id": 4,
          "author_id": 1,
          "author_name" : "Joana Lima",
          "content": "Example content of the edited content itself",
          "date": "25-03-2012",
          "views": 1000,
          "upvotes": 200,
          "downvotes": 100,
          "solved": false,
          "topics": [
            "Maths",
            "Science"
          ]
        }
      ]
    }
}
```

```

    ],
    "answers" : [
      {
        "id": 4,
        "author_id": 1,
        "author_name" : "Joana Lima",
        "content": "Example content of the edited content itself",
        "date": "25-03-2012",
        "views": 1000,
        "upvotes": 200,
        "downvotes": 100
      },
      {
        "id": 4,
        "author_id": 1,
        "author_name" : "Joana Lima",
        "content": "Example content of the edited content itself",
        "date": "25-03-2012",
        "views": 1000,
        "upvotes": 200,
        "downvotes": 100
      }
    ],
    },
    "message": "Example error message when getting feed"
  }

```

JSON505 - Search results

```

{
  "search":
    {
      "questions" : [
        {
          "id": 4,
          "author_id": 1,
          "author_name" : "Joana Lima",
          "content": "Example content of the edited content itself",
          "date": "25-03-2012",
          "views": 1000,
          "upvotes": 200,
          "downvotes": 100,
          "solved": false,
          "topics": [

```

```

        "Maths",
        "Science"
    ]
},
{
    "id": 4,
    "author_id": 1,
    "author_name" : "Joana Lima",
    "content": "Example content of the edited content itself",
    "date": "25-03-2012",
    "views": 1000,
    "upvotes": 200,
    "downvotes": 100,
    "solved": false,
    "topics": [
        "Maths",
        "Science"
    ]
}
],
"answers" : [
{
    "id": 4,
    "author_id": 1,
    "author_name" : "Joana Lima",
    "content": "Example content of the edited content itself",
    "date": "25-03-2012",
    "views": 1000,
    "upvotes": 200,
    "downvotes": 100
},
{
    "id": 4,
    "author_id": 1,
    "author_name" : "Joana Lima",
    "content": "Example content of the edited content itself",
    "date": "25-03-2012",
    "views": 1000,
    "upvotes": 200,
    "downvotes": 100,
}
],
"topics" : [
{
    "id": 4,

```

```

    "name": "History",
    "description": "The study of the past as it is described in
written documents."
    "picture" : "default.png"
  },
  {
    "id": 4,
    "name": "History",
    "description": "The study of the past as it is described in
written documents."
    "picture" : "default.png"
  }
],
"message": "Example error message when searching"
}

```

Revision history

Changes made to the first submission:

1. Add missing necessary permissions on some resources for special UI cases (when UI is different according to the user permissions). Updated resources: R201 (View profile), R301 (View question), R309 (View answer), R401 (View topic)
2. Update on necessary permissions: when more than one is present, made clear(er) that only one of them is necessary (and not all at once)
3. Update mandatory status on some fields in some request bodies (are now optional (?) as they were not strictly mandatory (+) in the request). Updated resources: R204 (Edit Profile Action), R406 (Edit Topic Action): in both the now optional fields correspond to database columns that allow null values; R505 (Search API): search results filtering is not always present
4. Implemented edit topic UI and added its reference to the respective resource (R404)

GROUP1725, 13/04/2018

António Cunha Seco Fernandes de Almeida, up201505836@fe.up.pt

Bruno Manuel Nascimento Costa Galvinas Piedade, up201505668@fe.up.pt

Diogo Luis Rey Torres, up201506428@fe.up.pt

João Paulo Madureira Damas, up201504088@fe.up.pt