

A7: WEB RESOURCES DOCUMENTATION- LEARNIt (Q&A)

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 **Learn It**, 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	Web resources associated with user authentication, includes the following system features: login/logout, registration and recover password.
M02: Static pages	Web resources associated with immutable content pages, such as 404, about, terms, contacts.
M03: Users	Web resources associated with users, such as individual profile management: view and edit personal information, manage their followers/following, their settings. Furthermore, this module also includes the notifications management.
M04: Main content (questions, answers and comments)	Web resources associated with the categories, includes creating, viewing, editing and deleting questions, answers and comments.
M05: Feed and search	Web resources associated with the feed 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 moderator, manage reports, add new categories and faqs (Admin, moderator, faq).

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	Users that are owners of information (profile, questions, answers, comments) and can update their content.
ADM	Administrator	System administrator.
MOD	Moderator	Group of members that manage the website.

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

These are the endpoints available in the authentication module:

- **R101:** Login Form */login*
- **R102:** Login Action */login*
- **R103:** Logout Action */logout*
- **R104:** Register Form */register*
- **R105:** Register Action */register*
- **R106:** RecoverPassword Form */password/recover*
- **R107:** RecoverPassword Action */password/recover*

R101: Login Form

URL	<i>/login</i>
Description	Page with a form to login to a user account.
Method	GET
UI	UI01
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 admin or moderator page on success and to login form on failure.	
Method	POST	
Request Body	+username: string or +email: string	Username or Email
	+password: string	Password
Redirects	R502	Success, if member
	R601	Success, if admin
	R602	Success, if moderator
	R101	Error
Permissions	PUB	

R103: Logout Action

URL	/logout	
Description	This web resource logs out the authenticated user, the moderator or the administrator of the system. Redirects to the login form.	
Method	POST	
Redirects	UI04	
Permissions	MEM or ADM or MOD	

R104: Register Form

URL	/register	
Description	Page with a form to register a new user account.	
Method	GET	
UI	UI02	
SUBMIT	R105	
Permissions	PUB	

R105: Register Action

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

R106: RecoverPassword Form

URL	/password/recover	
Description	Page with a form to request a token to reset the password.	
Method	GET	
UI	UI03	
SUBMIT	R107	
Permissions	USR, MOD, ADM	

R107: RecoverPassword Action

URL	/password/recover	
Description	Web resource that sends a reset password link to the specified email.	
Method	POST	
Request Body	+email: string	Email
Redirects	R102	Success
	R106	Error
Permissions	USR, MOD, ADM	

3.2. MODULE M02: STATIC PAGES

These are the endpoints available in the static pages module:

- **R201:** 404 Page */404*
- **R202:** Contact Page Form */contacts*
- **R203:** Contact Page Action */contacts*
- **R204:** About Page */about*
- **R205:** Terms Page */terms*
- **R206:** FAQ Page */faq*
- **R207:** Home Page */*

R201: 404 Page

URL	/404
Description	Get the 404 page
Method	GET
UI	UI22
Permissions	PUB

R202: Contacts Page Form

URL	/contacts
Description	Get the contacts page.
Method	GET
UI	UI08
SUBMIT	R203
Permissions	PUB

R203: Contacts Page Action

URL	/contacts	
Description	This web resource sends a message to the site master.	
Method	POST	
Request Body	+username: string	Username
	+email: string	Email
	+subject: string	Subject
	+message: string	Message
Redirects	R204	Success
	R502	Success, if member
	R202	Error
Permissions	PUB	

R204: About Page

URL	/about
Description	Get about page.
Method	GET
UI	UI05
Permissions	PUB

R205: Terms Page

URL	/terms
Description	Get the terms of use page.
Method	GET
UI	UI06
Permissions	PUB

R206: FAQ Page

URL	/faq
Description	Get the faqs page.
Method	GET
UI	UI07
Permissions	PUB

R207: Home Page

URL	/
Description	Get the Home page.
Method	GET
UI	UI04
Permissions	PUB

3.3. MODULE M03: USERS

These are the endpoints available in the user's module:

- **R301:** View Profile */profile/{username}*
- **R302:** Edit Profile Form */profile/edit*
- **R303:** Get Edit Profile Form API */api/members/{username}/edit-profile*
- **R304:** Edit Profile Action */profile/{username}*
- **R305:** View Followers */profile/{username}/followers*
- **R306:** View Following */profile/{username}/following*
- **R307:** Follow/ unfollow user API */api/member/{username}/toggle-follow*
- **R308:** View Settings */username/settings*
- **R309:** Get Settings API */api/members/{username}/settings*
- **R310:** Edit E-mail API */api/members/{username}/settings/email*
- **R311:** Edit Password API */api/members/{username}/settings/password*
- **R312:** Edit Google Account API */api/members/{username}/settings/google-account*
- **R313:** Edit Facebook Account API */api/members/{username}/settings/facebook-account*
- **R314:** Delete account */api/members/{username}/settings/delete-account*
- **R315:** Get Notifications API */api/members/{username}/notifications*
- **R316:** Mark a Notification as Read API */api/members/{username}/notifications/{notification_id}*

R301: View Profile

URL	<i>/profile/{username}</i>	
Description	Page with a member's profile info.	
Method	GET	
Parameters	+ username: string	Member's username
UI	UI09	
UI observations	UI changes according to permissions. If member has OWN permissions, editing options are shown. If member has MEM permissions, follow option should be displayed on all profiles that are not one's own.	
Permissions	PUB (general access) or OWN (for editing)	

R302: Edit Profile Form

URL	<i>/profile/edit</i>
Description	Page with a form to edit one's profile info.
Method	GET
UI	UI10
Ajax calls	R203
SUBMIT	R204
Permissions	OWN

R303: 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
Response Body	JSON303	
Permissions	OWN	

R304: 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
Request Body	+ username: string	username
	+ email: string	email
	+ password: string	password
	? bio	Bio description
	? picture	Profile photo
Redirects	R301	Success
	R303	Error
Permissions	OWN	

R305: View Followers

URL	/profile/{username}/followers	
Description	Page with the member's followers.	
Method	GET	
Parameters	+ username: string	Member's username
UI	UI11	
Permissions	PUB	

R306: View Following

URL	profile/{username}/following	
Description	Page with the members that another member follows.	
Method	GET	
Parameters	+ username: string	Member's username
UI	UI11	
Permissions	PUB	

R307: Follow/ unfollow user 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
Returns	200 Ok	The following status was updated
	400 Bad request	Error. Error message is specified via http header.
	404 Not found	Error. No member with the specified username exists.
Permissions	OWN	

R308: View Settings

URL	/{username}/settings	
Description	Page with the member's settings.	
Method	GET	
UI	UI23	
Ajax Calls	R309, R310, R311, R312, R313	
Permissions	OWN	

R309: Get Settings API

URL	/api/members/{username}/settings	
Description	Obtains a member's current settings.	
Method	GET	
Parameters	+ username: string	Member's username.
Response body	JSON309	
Permissions	OWN	

R310: 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.
Request Body	+ email: string	New email
Returns	200 Ok	The email was updated.
	400 Bad request	Error. Error message is specified via http header.
	404 Not found	Error. No member with the specified username exists.
Response body	JSON310	
Permissions	OWN	

R311: 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.
Request Body	+ password: string	New password
Returns	200 Ok	The password was updated.
	400 Bad request	Error. Error message is specified via http header.
	404 Not found	Error. No member with the specified username exists.
Permissions	OWN	

R312: 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.
Request Body	+ account: string	New google account
Returns	200 Ok	The account was updated.
	400 Bad request	Error. Error message is specified via http header.
	404 Not found	Error. No member with the specified username exists.
Permissions	OWN	

R313: 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.
Request Body	+ account: string	New facebook account.
Returns	200 Ok	The account was successfully updated.
	400 Bad request	Error. Error message is specified via http header.
	404 Not found	Error. No member with the specified username exists.
Permissions	OWN	

R314: Delete Account

URL	/api/members/{username}/settings/delete-account	
Description	This resource deletes the member's account.	
Method	DELETE	
Parameters	+ username: string	Member's username.
	200 Ok	The account was successfully deleted.
	400 Bad request	Error. Error message is specified via http header.
	404 Not found	Error. No member with the specified username exists.
Permissions	OWN	

R315: Get Notifications API

URL	/api/members/{username}/notifications	
Description	This resource gets the member's notifications.	
Method	GET	
Parameters	+ username: string	Member's username
Response body	JSON315	
Permissions	OWN	

R316: Mark a Notification as Read API

URL	/api/members/{username}/notifications/{id_notification}	
Description	This resource marks a notification as read.	
Method	PUT	
Parameters	+ username: string	Member's username.
	+ id_notification: integer	Notifications primary key
Returns	200 Ok	The account was successfully deleted.
	400 Bad request	Error. Error message is specified via http header.
	404 Not found	Error. No member with the specified username exists.
Permissions	OWN	

3.4. MODULE M04: MAIN CONTENT

These are the endpoints available in the main content module:

Topic:

- **R401:** View Topic */topic/{name}*

Questions:

- **R402:** View Question */questions/{id_question}*
- **R403:** Add Question From */questions/add*
- **R404:** Add Question Action */questions/add*
- **R405:** Edit Question API */api/questions/{id_question}*
- **R406:** Delete Question API */api/questions/{id_question}*
- **R407:** Rate Question API */api/questions/{id_question}/rate*
- **R408:** Report Action API */api/questions/{id_question}/report*

Answers:

- **R409:** Get Answer */questions/{id_question}/answers/{id_answer}*
- **R410:** Add Answer API */api/questions/{id_question}/answers*
- **R411:** Edit Answer API */api/questions/{id_question}/answers/{id_answer}*
- **R412:** Delete Answer API */api/questions/{id_question}/answers/{id_answer}*
- **R413:** Rate Answer API */api/questions/{id_question}/answers/{id_answer}/rate*
- **R414:** Report API */api/questions/{id_question}/answers/{id_answer}/report*
- **R415:** Assign/unassign as best answer API */api/{id_question}/answers/{id_answer}/best-answer*

Comment:

- **R416:** Get Comment API */api/questions/{id_question}/comments/{id_comment}*
- **R417:** Add Comment API */api/questions/{id_question}/comments*
- **R418:** Edit Comment API */api/questions/{id_question}/comments/{id_comment}/edit*

- **R419:** Delete Comment API *api/questions/{id_question}/comments/{id_comment}*
- **R420:** Rate Comment API
api/questions/{id_question}/comments/{id_comment}/rate
- **R421:** Report Comment API
api/questions/{id_question}/comments/{id_comment}/rate

R401: View Topic

URL	/topics /{name}	
Description	Displays the topic's page.	
Method	GET	
Parameters	+ name: String	Topic name
UI	UI19	
Permissions	PUB	

R402: View Question

URL	/questions/{id_question}	
Description	Displays the question's page.	
Method	GET	
Parameters	+ id: integer	Question primary key
Ajax Calls	R307, R403, R405, R406, R407, R408, R409, R410, R411, R412, R413, R414, R415, R416, R417, R418, R419, R420, R421	
UI	UI21	
Ui observations	UI changes according to permissions. If viewer has PUB permissions, no content options should be displayed. If MEM, report, rating and comment/ answer options. If OWN, editing options only for the member's content.	
Permissions	PUB (general access) or MEM (rating, reporting content, comment) or OWN (for editing one's content)	

R403: Add Question Form

URL	/questions/add	
Description	Page with a form to add a new question.	
Method	GET	
Parameters	+ category: string	Question's category.
	+ text: string	Question
	? description: string	Question's description.
	? photo: string	Question's photo or file.
UI	UI18	
SUBMIT	R303	
Permissions	MEM	

R404: Add Question Action

URL	/questions/add	
Description	This web resource adds a new question.	
Method	POST	
Request Body	+ id_author: integer	Member primary key
	+ category: string	Question's category.
	+ text: string	Question
	? description: string	Question's description.
	? photo: string	Question's photo or file.
Redirects	R402	Success
	R403	Error
Permissions	MEM	

R405: Edit Question API

URL	/api/questions/{id_question}	
Description	This web resource updates a question.	
Method	PUT	
Parameter	+ id_question: integer	Question primary key
Request Body	+ category: string	Question's category.
	+ text: string	Question
	? description: string	Question's description.
	? photo: string	Question's photo or file.
Response body	JSON405	The question's information when successful, an error message when unsuccessful.
Permissions	OWN	

R406: Delete Question API

URL	/api/questions/{id_question}	
Description	This web resource deletes a question.	
Method	DELETE	
Parameters	+ id_question: 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	

R407: Rate Question API

URL	/api/questions/{id_question}/rate	
Description	This web resource rates a question.	
Method	PUT	
Parameters	+ id_question: integer	Question primary key
	+ id_author: integer	User who rated the question.
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	

R408: Report Question API

URL	/api/questions/{id_question}/report	
Description	This web resource reports a question.	
Method	POST	
Parameters	+ id_question: integer	Question 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	

R409: Get Answer

URL	/questions{id_question}/answers/{id_answer}	
Description	This web resource gets an answer.	
Method	GET	
Parameters	+ id_question: integer	Question primary key
	+ id_answer: integer	Answer primary key
UI	UI21	
Response Body	JSON409	The answer's information when successful, an error message when unsuccessful
Permissions	MEM	

R410: Add Answer API

URL	/api/questions/{id_question}/answers	
Description	Page with a form to add a new answer.	
Method	POST	
Parameters	+ id_question: integer	Question primary key
Request Body	+ id_author: integer	Member primary key
	+ content: string	Content
Response Body	JSON410	The answer's information when successful, an error message when unsuccessful
Permissions	MEM	

R411: Edit Answer API

URL	/api/questions/{id_question}/answers/{id_answer}	
Description	This web resource updates an answer.	
Method	PUT	
Parameter	+ id_question: integer	Question primary key
Request Body	+ id_author: integer	Member primary key
	+ content: string	Content
Response body	JSON411	The answer's information when successful, an error message when unsuccessful.
Permissions	OWN	

R412: Delete Answer API

URL	/api/questions/{id_question}/answers/{id_answer}	
Description	This web resource deletes an answer.	
Method	DELETE	
Parameters	+ id_question: integer	Question primary key
	+ id_answer: 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 question with the specified primary key exists.
Permissions	OWN or MOD	

R413: Rate Answer API

URL	/api/questions/{id_question}/answers/{id_answer}/rate	
Description	This web resource rates an answer.	
Method	PUT	
Parameters	+ id_question: integer	Question primary key
	+ id_answer: integer	Answer's primary key
	+ id_author: integer	User who rated the question.
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	

R414: Report Answer API

URL	/api/questions/{id_question}/answers/{id_answer}/report	
Description	This web resource reports an answer.	
Method	PUT	
Parameters	+ id_question: integer	Question primary key
	+ id_answer: integer	Answer primary key
Request body	+ id_user: 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	

R415: Assign/unassign as best answer API

URL	/api/{id_question}/answers/{id_answer}/best-answer	
Description	This web resource assign/ unassign as best answer	
Method	PUT	
Parameters	+ id_question: integer	Question primary key
Returns	200 Ok	The answer was successfully Assign/unassign as best answer.
	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 (question's author)	

R416: Get Comment API

URL	/api/questions/{id_question}/comments/{id_comment}	
Description	This web resource gets a comment.	
Method	GET	
Parameters	+ id_comment: integer	Comment primary key
UI	UI21	
Response body	JSON416	The comment's information when successful, an error message when unsuccessful
Permissions	MEM	

R417: Add Comment API

URL	/api/questions/{id_question}/ comments	
Description	This web resource adds a new comment.	
Method	POST	
Parameters	+ id_question: integer	Question primary key
Request Body	+ id_author: integer	Member primary key
	+ id_answer: integer	Answer primary key
	+ content: string	Content
Response Body	JSON417	The comment's information when successful, an error message when unsuccessful
Permissions	MEM	

R418: Edit Comment API

URL	/api/questions/{id_question}/comments/{id_comment}/edit	
Description	This web resource updates a comment.	
Method	PUT	
Parameter	+ id_comment: integer	Comment primary key
Request Body	+ id_author: integer	Member primary key
	+ content: string	Content
Response body	JSON418	The comment's information when successful, an error message when unsuccessful.
Permissions	OWN	

R419: Delete Comments API

URL	/api/questions/{id_question}/ comments/{id_comment}	
Description	This web resource deletes a comment.	
Method	DELETE	
Parameters	+ id_question: integer	Question primary key
	+ id_answer: integer	Answer primary key
	+ id_comment: integer	Comment 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 question with the specified primary key exists.

Permissions	OWN or MOD	
--------------------	------------	--

R420: Rate Comment API

URL	/api/questions/{id_question}/ comments/{id_comment}/rate	
Description	This web resource rates an answer.	
Method	PUT	
Parameters	+ id_question: integer	Question primary key
	+ id_answer: integer	Answer's primary key
	+ id_author: integer	User who rated the question.
	+ id_comment: integer	Comment's primary key
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	

R421: Report Comment API

URL	/api/questions/{id_question}/comments/{id_comment}/report	
Description	This web resource reports a comment.	
Method	PUT	
Parameters	+ id_question: integer	Question primary key
	+ id_answer: integer	Answer primary key
	+ id_comment: integer	Comment primary key
Request body	+ id_user: 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	

3.5. MODULE M05: FEED AND SEARCH

These are the endpoints available in the feed and search module:

- **R501:** View Feed */feed*
- **R502:** Get Feed API */api/members/{username}/feed*
- **R503:** Search API */api/search*

R501: View Feed

URL	/feed
Description	Page with the member's personal feed
Method	GET
UI	UI20
Ajax Calls	R307, R502
Permissions	MEM

R502: 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
Response Body	JSON502	
Permissions	MEM	

R503: 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
	+ name: string	Category's content
	? filter_option: integer	Filters option
Response body	JSON503	
Permissions	PUB	

3.6. MODULE M06: ADMINISTRATION

These are the endpoints available in the administration module:

- **R601:** Admin Page */admin*
- **R602:** Moderator Page */moderator*
- **R603:** Search for Members API */api/members*
- **R604:** Add Category API */api/admin/add-category*
- **R605:** Edit Category API */api/admin/category/{id}*
- **R606:** Ban Member API */api/members/{username}/ban*
- **R607:** Promote Member API */api/members/{username}/promote*
- **R608:** Dismiss Moderator API */api/members/{username}/dismiss*

R601: Admin Page

URL	/admin
Description	Get admin page.
Method	GET
UI	UI12
Ajax Calls	R603 R604, R605, R606, R607, R608
Permissions	ADM

R602: Moderator Page

URL	/moderator
Description	Get moderator page.
Method	GET
UI	UI16
Ajax Calls	R603, R606
Permissions	MOD

R603: 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	JSON603	
Permissions	ADM or MOD	

R604: Add Category API

URL	/api/admin/add-category	
Description	This web resource adds a new category.	
Method	POST	
Request Body	+ name: string	Category's name
	+ icon: text	Category's icon
Response body	JSON604	The category's information when successful, an error message when unsuccessful
Permissions	ADM	

R605: Edit Category API

URL	/api/admin/category/{id}	
Description	This web resource updates a category.	
Method	PUT	
Parameters	+ id_category: integer	Category primary key
Request body	+ name: string	Category's name
	+ icon: text	Category's icon
Response body	JSON606	The category's information when successful, an error message when unsuccessful
Permissions	ADM	

R606: Ban Member API

URL	/api/members/{username}/ban	
Description	This web resource bans a member.	
Method	POST	
Parameters	+ username: string	Member's username
Returns	200 Ok	The following status was updated
	400 Bad request	Error. Error message is specified via http header.
	404 Not found	Error. No member with the specified username exists.
Permissions	ADM or MOD	

R607: Promote Member API

URL	/api/members/{username}/promote	
Description	This web resource promotes a member to acquire moderator privileges.	
Method	PUT	
Parameters	+ username: string	Member's username
Returns	200 Ok	The following status was updated
	400 Bad request	Error. Error message is specified via http header.
	404 Not found	Error. No member with the specified username exists.
Permissions	ADM or MOD	

R608: Dismiss Moderator API

URL	/api/members/{username}/dismiss	
Description	This web resource removes the moderator privileges of a member.	
Method	PUT	
Parameters	+ username: string	Member's username
Returns	200 Ok	The following status was updated
	400 Bad request	Error. Error message is specified via http header.
	404 Not found	Error. No member with the specified username exists.
Permissions	ADM	

4. JSON/XML TYPES**JSON303: Edit profile (default values for form)**

```
{
  "data":
  {
    "id": 1,
    "username": "SmilingOwl",
    "bio_description": "Hi! I love this forum!",
    "profile_photo": "sophie.jpg"
  },
  "message": "Example error message when editing a profile"
}
```


JSON309: Get current member settings

```
{
  "data": {
    "id": 1,
    "email": "sophie2000@gmail.com",
    "google_account": "myfacebookaccount",
    "facebook_account": "mygoogleaccount"
  },
  "message": "Example error message when getting a member settings"
}
```

JSON310: Edit email

```
{
  "data": {
    "email": "sophie2000@gmail.com"
  },
  "message": "Example error message when editing an email"
}
```

JSON315: Get notifications

```
{
  "notifications": [
    {
      "id_notification": 1,
      "type": "comment",
      "date": "2017-04-23 01:57:00",
      "description": "Peter left a comment on your question:
'How do a black holes impact time?'",
      "view": true
    },
    {
      "id_notification": 2,
      "type": "answer",
      "date": "2017-04-23 01:57:00",
      "description": "Peter answered your question: 'How do a
black holes impact time?'",
      "view": true
    },
    {
      "id_notification": 1,
      "type": "follow",
      "date": "2017-05-31 02:35:13",
      "description": "Diogo start follow you",
      "view": false
    }
  ],
  "message": "Example error message when getting notifications"
}
```

JSON405: Edit question

```
{
  "DATA": {
    "ID_QUESTION": 4,
    "TITLE": "WHAT IS  $\sin(x) + \cos(x)$ ?",
    "DESCRIPTION": "PLEASE HELP ME!",
    "DATE": "2017-01-01 05:00:00",
    "VOTES": 1000,
    "PHOTO": "YVES.JPG",
    "DELETED": FALSE,
    "ID_CATEGORY": 5,
    "ID_USER": 20
  },
  "MESSAGE": "EXAMPLE ERROR MESSAGE WHEN GETTING A QUESTION"
}
```

JSON409, JSON410 and JSON411: Get/ add or edit an answer

```
{
  "DATA": {
    "ID_ANSWER": 4,
    "TEXT": "MAYBE 1",
    "DATE": "2018-01-02 12:35:01",
    "VOTES": 5,
    "PHOTO": NULL,
    "DELETED": "FALSE",
    "ID_QUESTION": 10,
    "USER_POST": "JOANA"
  },
  "MESSAGE": "EXAMPLE ERROR MESSAGE WHEN GETTING/ADDING/EDITING AN ANSWER"
}
```

JSON416, JSON417 and JSON418: Get, add or edit a comment

```
{
  "DATA": {
    "FIRSTANSWER": 1,
    "SECONDANSWER": 2,
    "SECONDANSWERDATA": {
      "ID_ANSWER": 4,
      "TEXT": "YOUR ANSWER HELP ME A LOT!",
      "DATE": "2018-01-02 12:35:01",
      "VOTES": 5,
      "PHOTO": NULL,
      "DELETED": "FALSE",
      "ID_QUESTION": 10,
      "USER_POST": "ANNA"
    }
  },
  "MESSAGE": "EXAMPLE ERROR MESSAGE WHEN GETTING/ADDING/EDITING A COMMENT"
}
```

JSON502: Get feed

```

{
  "FEED": {
    "QUESTIONS": [
      {
        "ID_QUESTION": 10,
        "TITLE": "WHAT IS  $\sin(x) + \cos(x)$ ?",
        "DESCRIPTION": "PLEASE HELP ME!",
        "DATE": "2017-01-01 05:00:00",
        "VOTES": 1000,
        "PHOTO": "YVES.JPG",
        "DELETED": FALSE,
        "ID_CATEGORY": 5,
        "ID_USER": 20
      },
      {
        "ID_QUESTION": 4,
        "TITLE": "WHAT IS REQUIRED TO START SCULPTING CLAY (TOOLS,
EXPERIENCE...)?",
        "DESCRIPTION": "PLEASE HELP ME!",
        "DATE": "2017-06-01 14:00:00",
        "VOTES": 500,
        "PHOTO": "YVES.JPG",
        "DELETED": FALSE,
        "ID_CATEGORY": 2,
        "ID_USER": 5
      }
    ],
    "ANSWERS": [
      {
        "ID_ANSWER": 4,
        "TEXT": "MAYBE 1",
        "DATE": "2018-01-02 12:35:01",
        "VOTES": 5,
        "PHOTO": NULL,
        "DELETED": "FALSE",
        "ID_QUESTION": 10,
        "USER_POST": "JOANA"
      },
      {
        "ID_ANSWER": 7,
        "TEXT": "WHEN YOU ARE SCULPTING IT IS IMPORTANT TO HAVE A WELL-
ORGANIZED.",
        "DATE": "2019-01-02 12:35:01",
        "VOTES": 5,
        "PHOTO": NULL,
        "DELETED": "FALSE",
        "ID_QUESTION": 4,
        "USER_POST": "PETER"
      }
    ]
  },
  "MESSAGE": "EXAMPLE ERROR MESSAGE WHEN GETTING FEED"
}

```

JSON503: Search results

```

{
  "SEARCH": {
    "QUESTIONS": [
      {
        "ID_QUESTION": 10,
        "TITLE": "WHAT IS  $\sin(x) + \cos(x)$ ?",
        "DESCRIPTION": "PLEASE HELP ME!",
        "DATE": "2017-01-01 05:00:00",
        "VOTES": 1000,
        "PHOTO": "YVES.JPG",
        "DELETED": FALSE,
        "ID_CATEGORY": 5,
        "ID_USER": 20
      },
      {
        "ID_QUESTION": 4,
        "TITLE": "WHAT IS REQUIRED TO START SCULPTING CLAY (TOOLS,
EXPERIENCE...)?",
        "DESCRIPTION": "PLEASE HELP ME!",
        "DATE": "2017-06-01 14:00:00",
        "VOTES": 500,
        "PHOTO": "YVES.JPG",
        "DELETED": FALSE,
        "ID_CATEGORY": 2,
        "ID_USER": 5
      }
    ],
    "ANSWERS": [
      {
        "ID_ANSWER": 4,
        "TEXT": "MAYBE 1",
        "DATE": "2018-01-02 12:35:01",
        "VOTES": 5,
        "PHOTO": NULL,
        "DELETED": "FALSE",
        "ID_QUESTION": 10,
        "USER_POST": "JOANA"
      },
      {
        "ID_ANSWER": 7,
        "TEXT": "WHEN YOU ARE SCULPTING IT IS IMPORTANT TO HAVE A WELL-
ORGANIZED.",
        "DATE": "2019-01-02 12:35:01",
        "VOTES": 5,
        "PHOTO": NULL,
        "DELETED": "FALSE",
        "ID_QUESTION": 4,
        "USER_POST": "PETER"
      }
    ]
  },
  "MESSAGE": "EXAMPLE ERROR MESSAGE WHEN SEARCHING"
}

```

JSON603: Search user

```
{
  "DATA": {
    "ID": 0,
    "USERNAME": "SOPHIE"
  },
  "MESSAGE": "EXAMPLE ERROR MESSAGE GETTING MATCHING USER"
}
```

JSON604 and JSON606: Get, add or edit a category

```
{
  "DATA": {
    "ID_CATEGORY": 2,
    "NAME": "BIOLOGY",
    "ICON": "BIOLOGY-01.svg"
  },
  "MESSAGE": "EXAMPLE ERROR MESSAGE WHEN GETTING/ADDING/EDITING A CATEGORY"
}
```

REVISION HISTORY

Changes made to the first submission:

1. Changed the URL of the home page to /;
2. Added missing UI interfaces in some resources;

GROUP1834:

- João Augusto dos Santos Lima, up201605314@fe.up.pt
- José Diogo da Cunha Moreira Trindade Martins, up201504761@fe.up.pt
- Juliana Maria Cruz Marques, up201605568@fe.up.pt (Editor)
- Leonor Ribeiro e Sousa Mendes de Freitas, up201207603@fe.up.pt