A7: Web Resources Documentation

Our project, answerly, is a web application for collaborative questions and answers.

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 the documentation for *Answerly*, including the following operations over data: create, read, update, and delete.

1. Overview

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

M01: Authentication and Individual Profile	Web resources associated with user authentication and indivifual profile management. Includes features: login/ logout and registration, view and edit personal profile information.
M02: Questions, Answers and Comments	Web resources associated with questions, answers and their comments. Includes features: add, view, upvote, downvote, follow and report questions, add, view, upvote, downvote and report answers, and add, view and report comments. Also deals with searches.
M03: Website Administration	Web resources associated with management of the website. Includes features: user promotion, demotion and banishment, posts management.
M04: Static Pages	Web resources with static content are associated with this module: About page.

2. Permissions

PUB	Guests	People just visiting the site
USR	Authenticated Users	Registered users with the ability of posting to the website.
OWN	Owner	Users that are owners of the information.
MOD	Moderatores	Group of Authenticated Users with Website Management Privileges
ADM	Administratores	Highest level of Website Management

3. Modules

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

3.1 Module M01: Authentication and Individual Profile

Endpoints of Module Authentication and Individual Profile:

- R101: Login Form [/login]
- R102: Login Action [/login]
- R103: Logout Action [/logout]
- R104: Register Form [/register]
- R105: Register Action [/register]
- R106: View Profile [/user/{id}]
- R107: Edit Profile [/api/user/{id}]

R101: Login Form

URL	/login
Description	Page with 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 home page on success and the login form on failure.
Method	POST
Request Body	+username:string (Username)
	+password:string (Password)
Redirects	R201 (on SUCCESS)
	R101 (on ERROR)
Permissions	PUB

R103: Logout Action

URL	/logout
Description	This web resource logs out the user from the system. Redirects to the login page.
Method	POST

Redirects	R101 (on SUCCESS)
Permissions	USR, MOD, ADM

R104: Register Form

URL	/register
Description	This web resource register a new user account.
Method	GET
UI	UI06
SUBMIT	R105
Permissions	PUB

R105: Register Action

URL	/register
Description	This web resource inserts the user into the system. Redirects to the home page on success and the register form on failure.
Method	POST
Request Body	+first_name:string (First Name)
	+last_name:string (Last Name)
	+email:string (Email)
	+username:string (Username)
	+password:string (Password)
	?bio:string (Description)
Redirects	R201 (on SUCCESS)
	R104 (on ERROR)
Permissions	PUB

R106: View Profile

URL	/user/{id}
Description	Shows a user's profile page
Method	GET
Parameters	+id:integer (user's primary key)

UI	UI05
Permissions	PUB

R107: Edit Profile

URL	/api/user/{id}
Description	This web resource edits a User's information.
Method	PUT
Parameters	+first_name: string (User's first name)
	+last_name: string (User's last name)
	+email: string (User's email)
	+bio: string (User's description)
	+username: string (User's username)
	+password: string (User's password)
Returns	200 OK (The User was successfully edited.)
	400 Bad request (Error. Error message is specified via a HTTP header.)
	404 Not Found (Error. Could not edit user's profile.)
Response Body	JSON107
Permissions	OWN

3.2 Module M02: Questions, Answers and Comments

Endpoints of Module Questions, Answers and Comments:

- R201: Home [/]
- R202: Search Form [/search]
- R203: Search Action [/search]
- R204: View Question, answers and comments [/question/{id}]
- R205: Add Question [/api/question]
- R206: Vote Question [/api/question/{id}/vote]
- R207: Follow Question [/api/question/{id}/follow]
- R208: Report Question [/api/question/{id}/report]
- R209: Add Answer [/api/answer]
- R210: Vote Answer [/api/answer/{id}/vote]
- R211: Report Answer [/api/answer/{id}/report]
- R212: Add Comment [/api/comment]
- R213: Report Comment [/api/comment/{id}/report]

R201: Home

URL	/
Description	Shows the home page, with some relevant questions and popular labels
Method	GET
UI	UI02
Permissions	PUB

R202: Search Form

URL	/search
Description	Page with a form regarding the search for specific questions
Method	GET
Parameters	?content:string (Question content, title or label)
	?answered:boolean (Answered)
	?positive_score:boolean (Positive Score)
	?start:date (Start)
	?end:date (End)
UI	UI07
AJAX Calls	R203
Permissions	PUB

R203: Search Action

URL	/search
Description	This web resource searches the system for questions and returns the results.
Method	POST
Parameters	?content:string (Question content, title or label)
	?answered:boolean (Answered)
	?positive_score:boolean (Positive Score)
	?start:date (Start)
	?end:date (End)
Response Body	JSON203
Permissions	PUB

R204: View Question

URL	/question/{id}
Description	Shows the the question page, including its answers and comments
Method	GET
Parameters	+id:integer (question's primary key)
UI	UI03
AJAX Calls	R206
	R207
	R210
	R305
	R306
	R307
	R308
	R309
	R310
	R311
	R312
	R313
Permissions	PUB

R205: Add Question

URL	/api/question
Description	This web resource lets user add a question.
Method	PUT
Parameters	+title: string (Question title)
	+content: string (Question content)
Returns	200 OK (The question was successfully added.)
	400 Bad request (Error. Error message is specified via a HTTP header.)
	404 Not Found (Error. Could not create question.)
Permissions	USR

R206: Vote Question

URL	/api/question/{id}/vote
Description	This web resource lets user vote a question.
Method	PUT
Parameters	+id: integer (Question id)
Returns	200 OK (The question was successfully liked or disliked.)
	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	USR

R207: Follow Question

URL	/api/question/{id}/follow
Description	This web resource lets the user follow a question.
Method	PUT
Parameters	+id: integer (Question id)
Returns	200 OK (The question was successfully followed.)
	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	USR

R208: Report Question

URL	/api/question/{id}/report
Description	This web resource lets the user report a question.
Method	PUT
Parameters	+id: integer (Question id)
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	USR

R209: Add Answer

URL	/api/answer

Description	This web resource lets a user add a new Answer, based on the input received.
Method	PUT
Parameters	+id: integer (Question id)
	+content: string (Answer content)
	+question_index: string (Question id)
Returns	200 OK (The answer was successfully added.)
	400 Bad request (Error. Error message is specified via a HTTP header.)
	404 Not Found (Error. No question with the specified primary key exists)
Response Body	JSON209
Permissions	USR

R210: Vote Answer

URL	/api/answer/{id}/vote
Description	This web resource lets the user vote an answer.
Method	PUT
Parameters	+id: integer (Answer id)
Returns	200 OK (The answer was successfully liked or disliked.)
	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	USR

R211: Report Answer

URL	/api/answer/{id}/report
Description	This web resource lets the user report an answer.
Method	PUT
Parameters	+id: integer (Answer id)
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	USR

R212: Add Comment

URL	/api/comment
Description	This web resource lets a user add a new Comment, based on the input received.
Method	PUT
Parameters	+id: integer (Question id)
	+content: string (Comment content)
	+question_index: string (Question id)
	+answer_index: string (Answer id)
Returns	200 OK (The comment was successfully added.)
	400 Bad request (Error. Error message is specified via a HTTP header.)
	404 Not Found (Error. No question with the specified primary key exists.)
Response Body	JSON212
Permissions	USR

R213: Report Comment

URL	/api/comment/{id}/report
Description	This web resource lets user report a comment.
Method	PUT
Parameters	+id: integer (Comment id)
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	USR

3.3 Module M03: Website Administration and Ownership

Endpoints of Website Administration:

- R301: Access moderation area [/admin]
- R302: Ban user [/admin/user/{user_id}/ban]
- R303: Promote user [/admin/user/{user_id}/promote]
- R304: Demote user [/admin/user/{user_id}/demote]
- R305: Edit question form [/api/question/{id}]
- R306: Edit question action [/api/question/{id}/update]
- R307: Edit answer form [/api/answer/{id}]
- R308: Edit answer action [/api/answer/{id}/update]
- R309: Edit comment form [/api/comment/{id}]

- R310: Edit comment action [/api/comment/{id}/update]
- R311: Delete question [/api/question/{id}]
- R312: Delete answer [/api/answer/{id}]
- R313: Delete comment [/api/comment/{id}]

R301: Access moderation area

URL	/admin
Description	Shows moderation area page.
Method	GET
UI	UI04
Permissions	MOD, ADM

R302: Ban user

URL	/admin/user/{user_id}/ban
Description	This web resource bans a user.
Method	PUT
Parameters	+user_id: integer (User id)
Returns	200 OK (The user was successfully banned.)
	400 Bad request (Error. Error message is specified via a HTTP header.)
	404 Not Found (Error. No user with the specified primary key exists.)
Permissions	MOD, ADM

R303: Promote user

URL	/admin/user/{user_id}/promote
Description	This web resource promotes an authenticated user to moderator or a moderator to administrator.
Method	PUT
Parameters	+user_id: integer (User id)
Returns	200 OK (The user was successfully promoted.)
	400 Bad request (Error. Error message is specified via a HTTP header.)
	404 Not Found (Error. No user with the specified primary key exists.)
Permissions	ADM

R304: Demote user

URL	/admin/user/{user_id}/demote
Description	This web resource demotes a moderator to an authenticated user or an administrator to moderator.
Method	PUT
Parameters	+user_id: integer (User id)
Returns	200 OK (The user was successfully demoted.)
	400 Bad request (Error. Error message is specified via a HTTP header.)
	404 Not Found (Error. No user with the specified primary key exists.)
Permissions	ADM

R305: Edit question form

URL	/api/question/{id}
Description	This web resource exchanges question's title and description by input fields.
Method	PUT
Parameters	+id: integer (Question id)
	+title: string (Question title)
	+description: string (Question content)
Returns	200 OK (The elements were successfully exchanged.)
	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, ADM

R306: Edit question action

URL	/api/question/{id}/update
Description	This web resource edits a question.
Method	PUT
Parameters	+id: integer (Question id)
	+title: string (Question title)
	+content: string (Question content)
Returns	200 OK (The question was successfully edited.)
	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, ADM

R307: Edit answer form

URL	/api/answer/{id}
Description	This web resource exchanges answer's content by an input field.
Method	PUT
Parameters	+id: integer (Answer id)
	+content: string (Answer content)
Returns	200 OK (The element was successfully exchanged.)
	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, ADM

R308: Edit answer action

URL	/api/answer/{id}/update
Description	This web resource edits an answer.
Method	PUT
Parameters	+id: integer (Answer id)
	+content: string (Answer content)
Returns	200 OK (The answer was successfully edited.)
	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, ADM

R309: Edit comment form

URL	/api/comment/{id}
Description	This web resource exchanges comment's content by an input field.
Method	PUT
Parameters	+id: integer (Comment id)
	+content: string (Comment content)

Returns	200 OK (The element was successfully exchanged.)
	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, ADM

R310: Edit comment action

URL	/api/comment/{id}/update
Description	This web resource edits a comment of an answer.
Method	PUT
Parameters	+id: integer (Comment id)
	+content: string (Comment content)
Returns	200 OK (The comment was successfully edited.)
	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, ADM

R311: Delete a question

URL	/api/question/{id}
Description	This web resource deletes a question.
Method	DELETE
Parameters	+id: integer (Question id)
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.)
Permissions	OWN, ADM

R312: Delete an answer

URL	/api/answer/{id}
Description	This web resource deletes an answer.
Method	DELETE
Parameters	+id: integer (Answer id)

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, ADM

R313: Delete comment

URL	/api/comment/{id}
Description	This web resource deletes a comment of a question.
Method	DELETE
Parameters	+id: integer (Comment id)
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, ADM

3.4 Module M04: Static Pages

Endpoints of Static pages:

- R401: About [/about]
- R402: 404 [/404]

R401: About

URL	/about
Description	Get about page.
Method	GET
UI	UI08
Permissions	PUB

R402: 404

URL	/404
Description	Get 404 page.
Method	GET
UI	UI10

Permissions   PUB

4. JSON/XML Types

The structure of the JSON formatted answers must be documented as illustrated below.

JSON107: User Profile: {user}

```
{
  "user":
    {
      "id": "7",
      "first_name": "Paulo",
      "last_name": "Cardoso",
      "email": "pcardoso@gmail.com",
      "bio": "caixa",
      "username": "pcardoso",
    }
}
```

JSON203: Search Questions: {questions}[]

```
{
  "question": [
    {
      "id": "1",
      "user_id": "4",
      "title": "Name some great method actors",
      "nr_likes": "7",
      "nr_dislikes": "2",
    },
    {
      "id": "7",
      "user_id": "5",
      "title": "Running a program from within java code",
      "nr_likes": "7",
      "nr_dislikes": "5",
    }
  ]
}
```

JSON209: Answer: {answer}

```json
{
  "answer":
    {
      "id": "4",
      "user_id": "2",
      "question_id": "4",
      "answer_date": "2019-05-01",
      "content": "I'm glad you realize there is a difference, because many people don't. When you do something daily, let's say, then you do it every day. Every. Day. Two words. Two separate words. "Every day" means, refers to, and includes all days, just like "every cantaloupe" means, refers to, and includes all cantaloupes. You get the idea. When something is commonplace, ordinary, mundane, routine, average, run-of-the-mill, plain, or typical, then it can be accurately described as everyday. One word. One. "Everyday" is an adjective, a descriptor.",
      "nr_likes": "30",
      "nr_dislikes": "2",
    }
}
```

JSON212: Comment: {comment}

```json
{
  "comment":
    {
      "id": "1",
      "user_id": "3",
      "question_id": "1",
      "answer_id": "NULL",
      "comment_date": "2019-12-07",
      "content": "I believe it would also be interesting if people commented on the worse method actors as well."
    }
}
```

Revision history

1. First submission (13/04/2020).
2. Changed some endpoints, fixed minor errors (03/05/2020).

GROUP2064, 03/05/2020

- Antonio Pedro Reis Ribeiro Sousa Dantas, up201703878@fe.up.pt
- [Editor] Eduardo João Santana Macedo, up201703658@fe.up.pt

- Nuno Miguel Teixeira Cardoso, up201706162@fe.up.pt
- Paulo Roberto Dias Mourato, up201705616@fe.up.pt