API Documentation

Assessment webs5

Wouter van Geel
Gijsbert van Corstanje

Version 1.0

Authentication

Please send with every protected request an authentication token.
The provided authentication tokens can have user (read) or admin (crud) rights.

```
$.ajax({
        url : [request_url],
        headers: {
                'auth' : [token]
        }
});
```

Examples

Example of a jQuery Ajax call.

Default the API will give an JSON response.
Add *?format=html* to the URI to change the response data type to HTML.

```
$.ajax({
        url : "/venues",
        headers: {
                'auth' : "provideTokenHere"
        },
        dataType: "json",
        type : "GET",
        success : function(data) {
                // Do something here
        }
});
```

Overview

Overview of all current implemented API endpoints.
Base url: http://venuerace-gw.herokuapp.com/

Method	Endpoint	Description	Auth protection
GET	/venues	Return all venues	user
POST	/venues	Add venue	admin
GET	/venues/:Name	Returns venue details	user
PUT	/venues/:Name	Update venue details	admin
DELETE	/venues/:Name	Delete venue	admin
GET	/races	Return all races	user
POST	/races	Add race	admin
GET	/races/:Name	Returns race details	user
PUT	/races/:Name	Update race details	admin
DELETE	/races/:Name	Delete race	admin

General responses

Response Code	Return value
200 - OK	n/a
| 401 - UNAUTHORIZED | ```
{
        error: "no valid authorization key"
}
``` |

Requests

Venues

URL	/venues
Method	GET
URL Params	n/a
Data Params	n/a

Success response	**Code:** 200 OK
	Content:
	<pre>{
	  "count": 100,
	  "results": [
	    {
	      "name": "BlaBlaBla2",
	      "category": "Dutch"
	    },
	    {
	      "name": "Smaak d'Italia",
	      "category": "Ice parlor"
	    }
	    ...
	}</pre>

Error response	n/a

URL	/venues
Method	POST
URL Params	n/a

Data Params	```{
	  "name": "Example",
	  "category": "Dutch"
	}```

Success response	**Code:** 200 OK
	Content:
	``{
	  "count": 100,
	  "results": [
	    {
	      "name": "BlaBlaBla2",
	      "category": "Dutch"
	    },
	    {
	      "name": "Smaak d'Italia",
	      "category": "Ice parlor"
	    }
	    ...
	}```

Error response	n/a

URL	/venue/:Name
Method	GET
URL Params	n/a
Data Params	n/a

Success response	**Code:** 200 OK
	Content:
	``{
	  "name": "Example",
	  "category": "Dutch"
	}```

Error response	```{ "message": "Not Found" }```

URL	/venue/:Name
Method	PUT
URL Params	n/a
Data Params	```{ "name": "Example", "category": "Dutch" }```
Success response	**Code:** 200 OK **Content:** ```{ "redirect": "/venues" }```
Error response	```{ "message": "Not Found" }```

URL	/venue/:Name
Method	DELETE
URL Params	n/a
Data Params	n/a

Success response	**Code:** 200 OK
	Content:
	`{`
	`  "redirect": "/venues"`
	`}`

Error response	`{`
	`  "message": "Not Found"`
	`}`

Races

URL	/races
Method	GET
URL Params	n/a
Data Params	n/a

Success response	**Code:** 200 OK
	Content:
	`{`
	  `"count": 1,`
	  `"results": [`
	    `{`
	      `"name": "testRace"`
	    `}`
	    `...`
	  `]`
	`}`

Error response	n/a

URL	/races
Method	POST
URL Params	n/a

Data Params	```
	{
	  "name":"Example",
	  "Status":"not_started"
	}
	``

Success response	**Code:** 200 OK
	Content:
	``
	{
	  "count": 1,
	  "results": [
	    {
	      "name": "testRace"
	    }
	  ]
	}
	``

Error response	n/a

URL	/races/:Name
Method	GET
URL Params	n/a
Data Params	n/a

Success response	**Code:** 200 OK
	Content:
	``
	{
	  "name": "testRace",
	  "status": "not_started"
	}
	``

Error response	```
	{
	  "message": "Not Found"
	}
	``

URL	/races/:Name
Method	PUT
URL Params	n/a

Data Params	```
	{
	    "name": "Example",
	    "status": "started"
	}
	``

Success response	**Code:** 200 OK
	Content:
	``
	{
	  "redirect": "/races"
	}
	``

Error response	```
	{
	  "message": "Not Found"
	}
	``

URL	/races/:Name
Method	DELETE
URL Params	n/a
Data Params	n/a

Success response	**Code:** 200 OK
	Content:
	```{```
	``` "redirect": "/races"```
	```}```

Error response	```{```
	``` "message": "Not Found"```
	```}```