

Web Engineering Project: NRP REST-API

Radu Rebeja S4051297, r.rebeja@student.rug.nl

Eugen Falca S4341392, e.falca@student.rug.nl

Edwin Geertsma S4423453, a.e.geertsma@student.rug.nl

December 7, 2021

Introduction

Project NRP-API's purpose is to provide a database system management software that is accessible through the web. The interface is provided by a frontend layer implemented in React and using the bootstrap framework. The backend is created with Node.js (Express.js) and uses MySQL as the database management system.

In this project we provide a tool with a defined set of operations that can be applied to a dataset. Our app provides an interface to the data contained in the Netherlands Rental Properties dataset.

Naming convention: We will interchangeably refer to a database **entry** which contains fields from a Kamernet post as an **article** .

Supported operations:

- Create & store new entries in the database
- Retrieve (and list), delete, update entries by quarrying the ID / Location (Lat-Long) / [...]
- Retrieve (and list) entries by applying filters : City, Active / Non-active , Cost (Interval) , CPSM (Cost per square meter)
- Retrieve statistical data (mean , median, sd) of : rental cost / deposit
- Customize number entries to be retrieved
- Customize pagination

Database schema

TABLE: properties

Field	Type	Null	Key	Default	Extra
<i>externalId</i>	char(36)	NO	PRI	NULL	
address	varchar(255)	YES		Empty	
postalCode	varchar(255)	YES		Empty	
city	varchar(255)	YES		Empty	
areaSqm	int	NO		0	
rent	int	NO		0	
deposit	int	NO		0	
isRoomActive	tinyint(1)	NO		1	
latitude	varchar(255)	YES		NULL	
longitude	varchar(255)	YES		NULL	
createdAt	datetime	NO		NULL	
updatedAt	datetime	NO		NULL	

API Design

Version

V1.0

Base-URL

http://localhost:8083/nrp

Content-Type

API endpoints use the following header fields:

- (Default) *Content-Type* : **application\json**
- *Content-Type* : **text\csv**

The API allows specifying in the query parameters the response's Content-Type. The parameter is optional and hence - if missing, the default format will be used. The response's header specification for Content-Type will be configured (based on the request) and the appropriate response body will be generated in the backend. The browser will display the converted response body into formatted HTML labels. For GET requests an export option will be provided to download the response's body (JSON or CSV).

HTTP status code summary and method usage

200 - OK : Returns expected value; *[*]*

204 - OK : 0 Results; *[GET]*

400 - Bad Request : Wrong request syntax; *[*]*

404 - Not Found : Requested resource doesn't exist; *[PUT, DELETE]*

500 - Method Error* : Error caught during method execution *[*]*

**Note:* This error indicates that the backend layer method is not handling the request correctly.

Resource usage

POST	/articles/new <i>Create a new article entry in the database</i>
Parameter	
Header Parameters <i>no header parameters</i>	
Body Parameters	
address	: property address
postalCode	: property postalCode
city	: property city
areaSqm	: property area size in square meters
rent	: property monthly rent
deposit	: property deposit
isRoomActive	: property true if not occupied
latitude	: property location latitude
longitude	: property location longitude
Response	application/json
200 ok	<pre>{ "address": "Dr. Benthemstraat", "postalCode": "7514CL", "city": "Enschede", "areaSqm": 125, "rent": 995, "deposit": 0, "isRoomActive": true, "latitude": "52.2255530000", "longitude": "6.8971690000", "createdAt": "2021-11-29T11:29:13.000Z", "updatedAt": "2021-11-29T11:29:13.000Z" }</pre>
500 error: method exception	<pre>{ "message": "An error occurred while creating new rental post" }</pre>

GET	/articles/id/{id} <i>Get specific article* with external id or a "not-found error" if the article is with the requested is missing</i>
Parameter	

Header Parameters

id : *externalId* value of the Kamernet article

Quarry Parameters

format : json or csv response format ; default = json ; optional parameter

Response

application/json

200 ok

```
{
  "externalId": "apartment-1775319",
  "address": "Dr. Benthemstraat",
  "postalCode": "7514CL",
  "city": "Enschede",
  "areaSqm": 125,
  "rent": 995,
  "deposit": 0,
  "isRoomActive": true,
  "latitude": "52.2255530000",
  "longitude": "6.8971690000",
  "createdAt": "2021-11-29T11:29:13.000Z",
  "updatedAt": "2021-11-29T11:29:13.000Z"
}
```

204 error: article not found

```
{ "message": "Article with id '11' not found!" }
```

500 error: method exception

```
{ "message": "An error occurred while retrieving article." }
```

GET

/articles/location?latitude={lat}&longitude={long}

get the article(s) at the provided latitude-longitude location

Parameter

Header Parameters

no header parameters

Quarry String Parameters

latitude : the latitude specified in the article

longitude : the longitude specified in the article

format : json or csv response format ; default = json ; optional parameter

Response

application/json

200 ok

```
[
  {
    "externalId": "apartment-1750096",
    "address": "West-Varkenoordseweg",
    "postalCode": "3074HN",
```

		<pre> "city": "Rotterdam", "areaSqm": 80, "rent": 1100, "deposit": 1100, "isRoomActive": true, "latitude": "51.8966010000", "longitude": "4.5149930000", "createdAt": "2021-11-29T11:29:13.000Z", "updatedAt": "2021-11-29T11:29:13.000Z" }, ...] </pre>
204	error: 0 matches	<pre> { "message": "0 properties found at the provided location" } </pre>
500	error: method exception	<pre> { "message": "Some error occurred while retrieving property." } </pre>

GET	/articles/search-budget?{options} <i>Get all articles with rent price between the requested min-max bounds</i>
Parameter	
Header parameters <i>no header parameters</i>	
Quarry Parameters	
min	: minimum rent price
max	: maximum rent price
format	: json or csv response format ; default = json ; optional parameter
Response	application/json
200	ok
	<pre> [{ "externalId": "apartment-1619636", "address": "Erve Kloppe", "postalCode": "7577SR", "city": "Oldenzaal", "areaSqm": 76, "rent": 901, "deposit": 0, "isRoomActive": 1, "latitude": "52.3250610000", </pre>

	<pre> "longitude": "6.9008880000", "createdAt": "2021-11-29T11:29:13.000Z", "updatedAt": "2021-11-29T11:29:13.000Z" }, ...] </pre>
204	error: 0 matches <pre> { "message": "0 Articles found for the specified quarry" } </pre>
500	error: method exception <pre> { "message": "Some error occurred while retrieving property." } </pre>

GET	/articles/top-list/{city}?options <i>Get N or all articles specifying active rentals sorted in either ascending or descending order and ordered by a specified options parameter (which must be numerical)</i>
Parameter	
Header parameters	
city	: filter the search list by city; optional parameter; if absent, use all cities
Quarry Parameters	
amount	: top N articles to be retrieved from the search
ascending	: = true if response should be in ascending order; optional parameter; default = false
format	: json or csv response format ; default = json ; optional parameter
Response	application/json
200	ok <pre> [{ "externalId": "room-1702909", "address": "Spilsluizen", "postalCode": "9712NT", "city": "Groningen", "areaSqm": 9, "rent": 50, "deposit": 50, "isRoomActive": true, "latitude": "53.2212490000", "longitude": "6.5636370000", "createdAt": "2021-11-29T11:29:13.000Z", "updatedAt": "2021-11-29T11:29:13.000Z" }, </pre>

	<pre>] ... </pre>
204	error: 0 matches <pre> { "message": "0 Articles found for the specified quarry" } </pre>
500	error: method exception <pre> { "message": "Some error occurred while retrieving property." } </pre>

GET	/articles/statistics/{city}?{options} <i>Get statistical data (entry count, mean, median, standard-deviation) for rent and deposit values for a specific city or all cities</i>
Parameter	
Header parameters	
city	: filter the search list by city; optional parameter; if absent, use all cities
Quarry Parameters	
format	: json or csv response format ; default = json ; optional parameter
Response	application/json
200	ok <pre> { "entries_n": 5084, "mean_cost": "531.0334", "sd_cost": 276.75173793908016, "mean_deposit": "358.5024", "sd_deposit": 415.3804553576052, "median_cost": 410, "median_deposit": 320, "city": "Groningen" } </pre>
204	error: got 0 matches <pre> { "message": "The quarried city name was not found in the database" } </pre>
500	error: method exception <pre> { "message": "Failed to fetch database statistics ." } </pre>

PUT	/articles/id/id <i>Update the database article entry with the provided externalId by supplying a json request body with new data</i>
Parameter	
Header parameters	
id	article's externalId
Body parameters	
*	all parameters specified in the database model section below
Body	application/json
<pre> { "externalId": "apartment-1775319", "address": "API Doc Example Request", "areaSqm": "999", "postalCode": "2021WE", "rent": "0" } </pre>	
Response	
application/json	
200	ok
<pre> { "message": "Article was updated successfully." } </pre>	
404	error: article not found
<pre> { "message": "\"Cannot update property with id= apartment.Property not found or input is empty !\"" } </pre>	
500	method exception
<pre> { "message": "Error updating property with id= apartment" } </pre>	

PUT	/articles/location?latitude={lat}&longitude={long} <i>Update the database articles with the provided location in latitude and longitude by supplying a json request body with new data</i>
Parameter	
Header Parameters	
<i>no header parameters</i>	
Quarry String Parameters	
latitude	: the latitude specified in the article
longitude	: the longitude specified in the article
Body Parameters	
(* : mandatory fields)	
*externalId	: article external id
address	: property address
postalCode	: property postalCode
city	: property city
areaSqm	: property area size in square meters
rent	: property monthly rent
deposit	: property deposit
isRoomActive	: property true if not occupied
latitude	: property location latitude
longitude	: property location longitude
Body	application/json
<pre>{ "externalId": "apartment-1775319", "address": "API Doc Example #2 Request", "areaSqm": "303", "postalCode": "2021WE", "rent": "0" }</pre>	
Response	
application/json	
200	ok
<pre>{ "message": "Updated 20 properties with lat=51.1066010000 & long= 4.5149930000" }</pre>	
204	error: 0 articles updated
<pre>{ "message": "Found 0 articles to update" }</pre>	
500	error: method exception

```
{
  "message": "Encountered an error while updating
            articles by location"
}
```

DELETE	/articles/{id} <i>remove article from the database with external id</i>
Parameter	
Header Parameters	
id	: external Id of article
Response application/json	
200	ok
	<pre>{ "message": "Article was deleted successfully!" }</pre>
404	error: storage not found
	<pre>{ "message": "Cannot delete article with id= apartment. Article not found!" }</pre>
500	error: method exception
	<pre>{ "message": "An error occurred while deleting article with id=apartment" }</pre>

DELETE	/articles/location?latitude={lat}&longitude={long} <i>remove article with specified location options latitude and longitude</i>
Parameter	
Header Parameters <i>no header parameters</i>	
Quarry String Parameters	
latitude	: the latitude specified in the article
longitude	: the longitude specified in the article
Response	application/json
200	ok
	<pre>{ "message": "Deleted 20 properties with lat=51.1066010000 & long= 4.5149930000." }</pre>
404	error: 0 matches
	<pre>{ "message": "Could not find any articles to delete." }</pre>
500	error: method exception
	<pre>{ "message": "Error occurred while deleting articles with lat=51.1066010000 & long= 4.5149930000." }</pre>