

Team 6: Zahra Jafari and Angela Popa

Hotel Booking API

Requirements

Design and implement a Hotel Booking RESTful API.

The API should contain minimum 10 different resources and at least one supported operation per resource.
Try serving real data from your API as much as possible.

Take into considerations the HATEOAS guidelines.

Resources and Endpoints

Resource	Endpoint	supported HTTP Request type	Remarks
hotels	/api/hotels	GET, POST	
	/api/hotels/<name>	GET, PUT, DELETE	
contacts	/api/hotels/<name>/contacts	GET	
	/api/hotels/<name>/contacts	POST	fields: email, telephone, faxNumber
rooms	/api/hotels/<name>/rooms	GET	
	/api/hotels/<name>/rooms/<room_name>/prices	GET	
images	/api/hotels/<name>/images	GET	
location	/api/hotels/<name>/location	GET	
payments	/api/hotels/<name>/payments	GET	
facilities	/api/hotel/<name>/facilities	GET	
	/api/hotel/<name>/facilities	POST	fields: name
bookings	/api/hotels/<name>/rooms/<room_name>/bookings	GET	
	/api/hotels/<name>/rooms/<room_name>/bookings	POST	fields: from, to, firstname, lastname, numberOfRooms
hotels at location	/api/locations/<location>/hotels	GET	all hotels at the named location
users	/api/users	GET	
	/api/users	POST	fields: *first_name, last_name, email, created_date
	/api/users/<userId>	GET	

Project setup:

- installation of the required tools (MongoDB, node.js, Postman). Installation details can be found under https://github.com/angelapopa/sws/tree/master/notes/environment_setup.md.
- import of the dataset (https://github.com/angelapopa/sws/tree/master/dataset/hotels_enriched.json)
- source code checkout: `git clone https://github.com/angelapopa/sws.git`
- installation of all needed node.js packages: `npm install` in a terminal window at `/sws/node-hotel-booking/`
- `npm start` - to start the node server
- for `POST` requests in Postman, don't forget to select `x-www-form-urlencoded` in the `Body` tab.

Used Dataset

We decided to use the dataset of the Mayrhofen and Seefeld schema.org annotated data. This dataset was provided to us in WS 2017.

From this dataset we filtered out only those entries that are hotels. For each entity we removed the annotated information (@type). We kept all field names as they were, namely schema.org properties.

This dataset was providing information like `hotel url`, `hotel name`, `hotel description`, `hotel location` (geo and postal address), `images`, `price range`, `rooms`.

This dataset was not providing information like `contact`, `bookings`, `facilities`, `users`, `navigation links`. The fields of this entities do not conform to the schema.org specification. This entities can be added via the API build during this project.

The `navigation urls` were generated and imported separately. Additionally, for every newly added hotel through the API, new `navigation links` are added automatically for the new hotel.

Restful API Pagination, Sorting, Filtering

This requirements were implemented inside the API logic and in this current version of the project they are not configurable outside the API.

Pagination: defaults to 10 entities per page

Sorting: by hotel name, ascending

Filtering: case dependent

Examples

Hotel Details

GET http://localhost:3000/api/hotels/Rauchenwalderhof

```
{
  "links": [
    {
      "_id": "5afcbcb1182d6018a87bf991",
      "href": "http://localhost:3000/api/hotels/Rauchenwalderhof",
      "rel": "self",
      "type": "GET"
    },
    {
      "_id": "5afcbcb1182d6018a87bf993",
      "href": "http://localhost:3000/api/hotels/Rauchenwalderhof/images",
      "rel": "images",
      "type": "GET"
    },
    {
      "_id": "5afcbcb1182d6018a87bf995",
      "href": "http://localhost:3000/api/hotels/Rauchenwalderhof/payments",
      "rel": "payments",
      "type": "GET"
    },
    {
      "_id": "5afcbcb1182d6018a87bf997",
      "href": "http://localhost:3000/api/hotels/Rauchenwalderhof/location",
      "rel": "location",
      "type": "GET"
    },
    {
      "_id": "5afcbcb1182d6018a87bf999",
      "href": "http://localhost:3000/api/hotels/Rauchenwalderhof/rooms",
      "rel": "rooms",
      "type": "GET"
    },
    {
      "_id": "5afcbcb1182d6018a87bf99b",
      "href": "http://localhost:3000/api/hotels/Rauchenwalderhof/contacts",
      "rel": "contacts",
      "type": "GET"
    },
    {
      "_id": "5afcbcb1182d6018a87bf99d",
      "href": "http://localhost:3000/api/hotels/Rauchenwalderhof/facilities",
      "rel": "facilities",
      "type": "GET"
    }
  ],
  "url": "http://www.rauchenwalderhof.at,http://maps.mayrhofen.at/?foreignResource=718BA2D7-2023-4353-9A23-C1B69BADEC06",
  "name": "Rauchenwalderhof",
  "description": "Das 2012 umgebaute und modern renovierte Landhotel Rauchenwalderhof liegt in sonniger, ruhiger TOP-Lage von Mayrhofen, nur wenige Meter von der Penkenbahn und Ahornba",
  "priceRange": "EUR 20.0 - 9999.0"
}
```

<

>

Hotel Images

GET http://localhost:3000/api/hotels/Rauchenwalderhof/images

```
[
  {
    "url": "http://resc.deskline.net/images/ZIL/1/0f2c4d9a-1bac-4828-88e5-bb717f09503b/99/image.jpg",
    "caption": "Aussenansicht Sommer"
  },
  {
    "url": "http://resc.deskline.net/images/ZIL/1/a4ea4d5c-360e-4718-af1e-2dc57bbfaa33/99/image.jpg",
    "caption": "Aussenansicht Winter"
  },
  {
    "url": "http://resc.deskline.net/images/ZIL/1/423d8bf3-3514-4412-b9a0-bcbeb1aecdde/99/image.jpg",
    "caption": "Rauchenwalderhof - Sommer"
  },
  {
    "url": "http://resc.deskline.net/images/ZIL/1/8674ce49-eddb-47ae-8df4-e733ee65039b/99/image.jpg",
    "caption": "Landhotel Rauchenwalderhof Mayrhofen"
  },
  {
    "url": "http://resc.deskline.net/images/ZIL/1/c76b1212-44cb-4eba-8228-4e1f20327c1b/99/image.png",
    "caption": "Rauchenwalderhof Logo"
  },
  {
    "url": "http://resc.deskline.net/images/ZIL/1/58efb8e9-48ff-49d3-abe6-c6dbbbe82449/99/image.jpg",
    "caption": "Rauchenwalderhof Terrasse"
  },
  [...]
]
```

Hotel Location

GET http://localhost:3000/api/hotels/Rauchenwalderhof/location

```
{
  "address": {
    "name": "Rauchenwalderhof",
    "streetAddress": "Rauchenwald 658",
    "addressLocality": "Mayrhofen",
    "postalCode": "6290",
    "addressCountry": "AT"
  },
  "geo": {
    "latitude": "47.1627288061039",
    "longitude": "11.8571770191193"
  }
}
```

Hotel Payment Methods

GET http://localhost:3000/api/hotels/Kammerlandhof/payments

```
{
  "paymentAccepted": [
    "MasterCard",
    "VISA"
  ],
  "currenciesAccepted": "EUR"
}
```

Hotel Facilities

GET http://localhost:3000/api/hotels/Haus%20Sailer/facilities

```
[
  {
    "name": "Spa"
  },
  {
    "name": "Restaurant"
  }
]
```

Hotel Rooms

GET http://localhost:3000/api/hotels/Tuxerhof/rooms

```
[
  {
    "bookings": [
      {
        "_id": "5af5fa2de7bcc42760efce78",
        "from": "2018.12.29",
        "to": "2018.12.31",
        "firstname": " Mathias",
        "lastname": "Mustermann"
      },
      {
        "_id": "5af5fa6885202f3a8859adc1",
        "from": "2018.12.24",
        "to": "2018.12.25",
        "firstname": " Martha",
        "lastname": "Mustermann"
      }
    ],
    "links": [
      {
        "_id": "5af9dd794910eb2bf4c64366",
        "href": "http://localhost:3000/api/hotels/Tuxerhof/rooms/Appartement%2FFewo",
        "rel": "self",
        "type": "GET"
      },
      {
        "_id": "5af9dd794910eb2bf4c64368",
        "href": "http://localhost:3000/api/hotels/Tuxerhof/rooms/Appartement%2FFewo/prices",
        "rel": "prices",
        "type": "GET"
      },
      {
        "_id": "5af9dd794910eb2bf4c6436a",
        "href": "http://localhost:3000/api/hotels/Tuxerhof/rooms/Appartement%2FFewo/bookings",
        "rel": "bookings",
        "type": "GET"
      }
    ],
    "name": "Appartement/Fewo",
    "description": "Unsere Ferienwohnung ist ca. 80 m² groß und bietet Platz für 2-7 Pers.!\nDie Wohnung besteht aus einer geräumigen Wohnküche, einem Vorraum, Dusche/WC und zwei se",
    "itemOffered": {
      "name": "Appartement/Fewo",
      "numberOfRooms": "5.0",
      "occupancy": {
        "maxValue": "9",
        "minValue": "2"
      }
    }
  }
]
```

Hotel Room Prices

GET http://localhost:3000/api/hotels/Tuxerhof/rooms/Appartement%2FFewo/prices

```
[
  {
    "description": "Pax 2",
    "minPrice": 45,
    "maxPrice": 60,
    "priceCurrency": "EUR"
  },
  {
    "description": "Pax 3",
    "minPrice": 60,
    "maxPrice": 70,
    "priceCurrency": "EUR"
  },
  [...]
]
```

All Hotels at Location Hippach, page 9

GET http://localhost:3000/api/locations/Hippach/hotels/9

```
{
  "links": [
    {
      "_id": "5afcbcb1182d6018a87bfa0f",
      "href": "http://localhost:3000/api/hotels/Haus%20Kirchler",
      "rel": "self",
      "type": "GET"
    },
    {
      "_id": "5afcbcb1182d6018a87bfa11",
      "href": "http://localhost:3000/api/hotels/Haus%20Kirchler/images",
      "rel": "images",
      "type": "GET"
    },
    {
      "_id": "5afcbcb1182d6018a87bfa13",
      "href": "http://localhost:3000/api/hotels/Haus%20Kirchler/payments",
      "rel": "payments",
      "type": "GET"
    },
    {
      "_id": "5afcbcb1182d6018a87bfa15",
      "href": "http://localhost:3000/api/hotels/Haus%20Kirchler/location",
      "rel": "location",
      "type": "GET"
    },
    {
      "_id": "5afcbcb1182d6018a87bfa17",
      "href": "http://localhost:3000/api/hotels/Haus%20Kirchler/rooms",
      "rel": "rooms",
      "type": "GET"
    },
    {
      "_id": "5afcbcb1182d6018a87bfa19",
      "href": "http://localhost:3000/api/hotels/Haus%20Kirchler/contacts",
      "rel": "contacts",
      "type": "GET"
    },
    {
      "_id": "5afcbcb1182d6018a87bfa1b",
      "href": "http://localhost:3000/api/hotels/Haus%20Kirchler/facilities",
      "rel": "facilities",
      "type": "GET"
    }
  ],
  "url": "http://",
  "address": {
    "streetAddress": "Schwendberg 365",
    "addressLocality": "Hippach",
    "postalCode": "6283",
    "addressCountry": "AT",
    "telephone": "(0043) 664 1616583",
    "email": "patriciakirchler93@gmx.at"
  },
  "name": "Haus Kirchler",
  "description": "1 Schlafzimmer (1 Doppelbett und einem Einzelbett) 1 Bad mit Dusche, Inklusive: Heizung, 1 Stellplatz für PKW, TV-Sat, Wlan, Handtücher, Klopapier, Geschirrtücher,
  "priceRange": "EUR 27.0 - 27.0"
},
  [...]
}
```

Hotel Bookings

GET http://localhost:3000/api/hotels/Tuxerhof/rooms/Appartement%2FFewo/bookings

```
[
  {
```

```
    "from": "2018.12.29",
    "to": "2018.12.31",
    "firstname": " Mathias",
    "lastname": "Mustermann"
  },
  {
    "from": "2018.12.24",
    "to": "2018.12.25",
    "firstname": " Martha",
    "lastname": "Mustermann"
  },
  {
    "from": "2018.12.24",
    "to": "2018.12.25",
    "firstname": " Martha",
    "lastname": "Mustermann",
    "numberOfRooms": 2
  },
  [...]
]
```

Hotel Contact Data

```
GET http://localhost:3000/api/hotels/KnechtIhof/contacts
```

```
[
  {
    "telephone": "555-888-222",
    "faxNumber": "55-12121-9977",
    "email": "knechtelhof@test.com"
  }
]
```

Hotel API Users

```
GET http://localhost:3000/users/
```

```
[
  {
    "created_date": "2018-05-16T19:37:47.516Z",
    "first_name": "User12",
    "last_name": "Lastname2",
    "email": "test2@test.com"
  },
  {
    "created_date": "2018-05-17T00:53:44.225Z",
    "first_name": "Matthias",
    "last_name": "Mustermann",
    "email": "test_mm@test.com"
  }
]
```

Hotel API Users

```
GET http://localhost:3000/api/users/5afc888bed29471e54bce756
```

Results in

```
{
  "created_date": "2018-05-16T19:37:47.516Z",
  "first_name": "User12",
  "last_name": "Lastname2",
  "email": "test2@test.com"
}
```

Future work

- the dataset contains contact data inside the postal address element. Merge this into the current `/contact` endpoint data.
- implement the remaining operations for the endpoints
- make pagination and sorting configurable from outside
- find a solution for Umlaute in hotel names w hen using them in endpoints