

IOP Ass 5b

Paul Bartl

May 26, 2022

REST Base path: <https://wwmlab.cs.univie.ac.at/~bartlp20/rest/index.php>

Contents

1	HATEOAS Experience	4
2	GET Endpoints	4
2.1	Teams with less than X employees	5
2.1.1	Resource Path	5
2.1.2	Purpose	5
2.1.3	SOAP operation signature	5
2.1.4	Example Call	5
2.1.5	Return data	5
2.1.6	Covered Requirements	5
2.2	Teams by Countries X	6
2.2.1	Resource Path	6
2.2.2	Purpose	6
2.2.3	SOAP operation signature	6
2.2.4	Example Call	6
2.2.5	Return data	6
2.2.6	Covered Requirements	7
2.3	Team with smallest Id	8
2.3.1	Resource Path	8
2.3.2	Purpose	8
2.3.3	SOAP operation signatur	8
2.3.4	Example Call	8
2.3.5	Return data	8
2.3.6	Covered Requirements	8

2.4	Get Customer by Zip	9
2.4.1	Resource Path	9
2.4.2	Purpose	9
2.4.3	SOAP operation signatur	9
2.4.4	Example Call	9
2.4.5	Return data	9
2.4.6	Covered Requirements	9
2.5	Get Streets by Zip	10
2.5.1	Resource Path	10
2.5.2	Purpose	10
2.5.3	SOAP operation signature	10
2.5.4	Example Call	10
2.5.5	Return data	10
2.5.6	Covered Requirements	11
2.6	Get Bars by Amount of Minifridges or Color	12
2.6.1	Resource Path	12
2.6.2	Purpose	12
2.6.3	SOAP operation signature	12
2.6.4	Example Call	12
2.6.5	Return data	12
2.6.6	Covered Requirements	13
2.7	Get planes by color	14
2.7.1	Resource Path	14
2.7.2	Purpose	14
2.7.3	SOAP operation signature	14
2.7.4	Example Call	14
2.7.5	Return data	14
2.7.6	Covered Requirements	15
2.8	Get planes by livery	16
2.8.1	Resource Path	16
2.8.2	Purpose	16
2.8.3	SOAP operation signature	16
2.8.4	Example Call	16
2.8.5	Return data	16
2.8.6	Covered Requirements	17
2.9	Get Protocol by Date	18
2.9.1	Resource Path	18
2.9.2	Purpose	18
2.9.3	SOAP operation signature	18
2.9.4	Example Call	18

2.9.5	Return data	18
2.9.6	Covered Requirements	19
2.10	Get Testdate Of Protocol	20
2.10.1	Resource Path	20
2.10.2	Purpose	20
2.10.3	SOAP operation signature	20
2.10.4	Example Call	20
2.10.5	Return data	20
2.10.6	Covered Requirements	20
3	POST Endpoints	21
3.1	Post a new bar to plane	22
3.1.1	Resource Path	22
3.1.2	Purpose	22
3.1.3	SOAP operation signatur	22
3.1.4	Example Call	22
3.1.5	Return data	22
3.1.6	Covered Requirements	23
4	PUT Endpoints	24
4.1	Update the Address Of Customer owning Plane	25
4.1.1	Resource Path	25
4.1.2	Purpose	25
4.1.3	SOAP operation signature	25
4.1.4	Example Call	25
4.1.5	Return data	25
4.1.6	Covered Requirements	26
5	PATCH Endpoints	27
5.1	Patch Flowrate of Toilets	28
5.1.1	Resource Path	28
5.1.2	Purpose	28
5.1.3	SOAP operation signature	28
5.1.4	Example Call	28
5.1.5	Return data	28
5.1.6	Covered Requirements	29
5.2	Patch Beverage	30
5.2.1	Resource Path	30
5.2.2	Purpose	30
5.2.3	SOAP operation signature	30

5.2.4	Example Call	30
5.2.5	Return data	30
5.2.6	Covered Requirements	30
6	DELETE Endpoints	31
6.1	Delete customer by Id	32
6.1.1	Resource Path	32
6.1.2	Purpose	32
6.1.3	SOAP operation signature	32
6.1.4	Example Call	32
6.1.5	Return data	32
6.1.6	Covered Requirements	32

1 HATEOAS Experience

<https://wwwlab.cs.univie.ac.at/~bartlp20/rest/index.php/hateaos/customers>

I recommend to pick user 495, because this customer has a special case in that he has two planes, one with protocols and one without, to showcase the included HATEOAS logic. All other customers should of course work as well.

2 GET Endpoints

2.1 Teams with less than X employees

2.1.1 Resource Path

`https://wwwlab.cs.univie.ac.at/~bartlp20/rest/index.php/teams/{numEmployees}`

2.1.2 Purpose

Returns all teams in 1_data.xml with less than or equal x amount of employees.

2.1.3 SOAP operation signature

`getTeamsByEmployees`

2.1.4 Example Call

```
curl --request GET \
  --url 'https://wwwlab.cs.univie.ac.at/~bartlp20/rest/index.php/teams/3'
```

2.1.5 Return data

```
{
  "Team": [
    {
      "@attributes": {
        "teamid": "609",
        "employees": "3"
      },
      "ISO_Code": "IT",
      "HourlyRate": "199.61",
      "Units": {
        ...
      }
    }
  ]
}
```

2.1.6 Covered Requirements

Operation	R1	R2	R3	R4	R5	R6	R7
<code>getTeamsByEmployees</code>		X					X

2.2 Teams by Countries X

2.2.1 Resource Path

<https://wwwlab.cs.univie.ac.at/~bartlp20/rest/index.php/teams/bycountries>

2.2.2 Purpose

Returns all teams in 1_data.xml which are based in countries provided by array X.

2.2.3 SOAP operation signature

getTeamsByCountries

2.2.4 Example Call

```
curl --request GET \
  --url 'https://wwwlab.cs.univie.ac.at/~bartlp20/rest/index.php/teams/
    bycountries' \
  --header 'Content-Type: application/json' \
  --data '{
    "countries": [
      {"country": "IT"},
      {"country": "IS"}
    ]
  }'
```

2.2.5 Return data

```
{
  "Team": [
    {
      "@attributes": {
        "teamid": "609",
        "employees": "3"
      },
      "ISO_Code": "IT",
      "HourlyRate": "199.61",
      "Units": {
        ...
      }
    }
  ]
}
```

2.2.6 Covered Requirements

Operation	R1	R2	R3	R4	R5	R6	R7
getTeamsByCountries	X				5.1		X

2.3 Team with smallest Id

2.3.1 Resource Path

<https://wwwlab.cs.univie.ac.at/~bartlp20/rest/index.php/teams/smallest>

2.3.2 Purpose

Returns the team with the smallest teamId in 1_data.xml.

2.3.3 SOAP operation signatur

getSmallestTeamID

2.3.4 Example Call

```
curl --request GET \  
  --url 'https://wwwlab.cs.univie.ac.at/~bartlp20/rest/index.php/teams/  
        smallest'
```

2.3.5 Return data

13

2.3.6 Covered Requirements

Operation	R1	R2	R3	R4	R5	R6	R7
getSmallestTeamID							

2.4 Get Customer by Zip

2.4.1 Resource Path

<https://wwwlab.cs.univie.ac.at/~bartlp20/rest/index.php/customers/byzip/zip>

2.4.2 Purpose

Returns all customers in 2_data.xml whose address has a zip of less than x. Example zip is 10000.

2.4.3 SOAP operation signatur

getCustomersByZip

2.4.4 Example Call

```
curl --request GET \
  --url 'https://wwwlab.cs.univie.ac.at/~bartlp20/rest/index.php/customers/
  byzip/10000'
```

2.4.5 Return data

```
{
  "Customer": [
    {
      "Firstname": "Stine",
      "Lastname": "Schumacher",
      "Address": {
        "@attributes": {
          "zip": "2728",
          "city": "Davidton"
        },
        ...
      }
    }
  ]
}
```

2.4.6 Covered Requirements

Operation	R1	R2	R3	R4	R5	R6	R7
getCustomersByZip		X				X	

2.5 Get Streets by Zip

2.5.1 Resource Path

<https://wwwlab.cs.univie.ac.at/~bartlp20/rest/index.php/streets/byzip/zip>

2.5.2 Purpose

Returns all streets in 3_data.xml which are based city whose zip code is less than x. Example zip is 10000.

2.5.3 SOAP operation signature

getStreetsByZip

2.5.4 Example Call

```
curl --request GET \  
  --url 'https://wwwlab.cs.univie.ac.at/~bartlp20/rest/index.php/streets/  
        byzip/10000'
```

2.5.5 Return data

```
{  
  "Street": [  
    "Mitchell Gateway",  
    "Julie Parkways",  
    "Alexander Road",  
    "Michael Corner",  
    "Jeremy Island",  
    "Beard Roads",  
    "Norton Haven",  
    "Hall Centers",  
    "Kevin Rest",  
    "Shannon Square",  
    "Shannon Square"  
  ]  
}
```

2.5.6 Covered Requirements

Operation	R1	R2	R3	R4	R5	R6	R7
getStreetsByZip		X				X	

2.6 Get Bars by Amount of Minifridges or Color

2.6.1 Resource Path

<https://wwwlab.cs.univie.ac.at/~bartlp20/rest/index.php/bars/byminifridgesorcolor/>

2.6.2 Purpose

Returns all registrations of planes which match the input criteria of array x. Array x consists of dict fridgesInterval and array colors. Every plane with at least one bar that has the amount of fridges between fridgesInterval min/max or matches one color in the colors array are included in the returned data.

2.6.3 SOAP operation signature

getBarMinifridgeAndColor

2.6.4 Example Call

```
curl --request GET \
  --url 'https://wwwlab.cs.univie.ac.at/~bartlp20/rest/index.php/bars/
    byminifridgesorcolor/' \
  --header 'Content-Type: application/json' \
  --data '{
    "colors": [
      "white",
      "maroon"
    ],
    "fridgesInterval": {
      "minFridges": 8,
      "maxFridges": 10
    }
  }'
```

2.6.5 Return data

```
{
  "Registration": [
    "CKN7593",
    "TEX8779",
    "WPA4428",
```

"MPF3551",
"YWK5335",
"FMW2950",
"BVK3060",
"ZIM6990",
"FAJ8688",

2.6.6 Covered Requirements

Operation	R1	R2	R3	R4	R5	R6	R7
getBarMinifridgeAndColor					5.3		X

2.7 Get planes by color

2.7.1 Resource Path

<https://wwwlab.cs.univie.ac.at/~bartlp20/rest/index.php/planes/bycolor/>

2.7.2 Purpose

Returns all planes which match one of the colors supplied by array x. Searches in 4_data.xml

2.7.3 SOAP operation signature

getPlanesByColor

2.7.4 Example Call

```
curl --request GET \
  --url 'https://wwwlab.cs.univie.ac.at/~bartlp20/rest/index.php/planes/
    bycolor/' \
  --header 'Content-Type: application/json' \
  --data '{
    "colors": [
      "white",
      "maroon"
    ]
  }'
```

2.7.5 Return data

```
<?xml version="1.0"?>
<Planes>
  <Plane registration="TEX8779">
    <Color>white</Color>
    <Livery>eurowhite</Livery>
    <Bars>
      <Bar unitid="8882156c-f456-40e3-a564-2c55e1d28b68"
        teamid="767">
        <Minifridges>
          <Amount>5</Amount>
        </Minifridges>
      </Bar>
    </Bars>
  </Plane>
  ...
</Planes>
```

2.7.6 Covered Requirements

Operation	R1	R2	R3	R4	R5	R6	R7
getPlanesByColor	X				5.1		X

2.8 Get planes by livery

2.8.1 Resource Path

<https://wwwlab.cs.univie.ac.at/~bartlp20/rest/index.php/planes/bylivery/>

2.8.2 Purpose

Returns all planes which match one of the liveries supplied by array x. Searches in 4_data.xml

2.8.3 SOAP operation signature

getPlanesByLivery

2.8.4 Example Call

```
curl --request GET \
  --url 'https://wwwlab.cs.univie.ac.at/~bartlp20/rest/index.php/planes/
    bylivery/' \
  --header 'Content-Type: application/json' \
  --data '{
    "liveries": [
      "eurowhite",
      "jellybean"
    ]
  }'
```

2.8.5 Return data

```
{
  "Plane": [
    {
      "@attributes": {
        "registration": "DAJ7178"
      },
      "Color": "maroon",
      "Livery": "eurowhite",
      "Bars": {}
    },
    ...
  ]
}
```


2.8.6 Covered Requirements

Operation	R1	R2	R3	R4	R5	R6	R7
getPlanesByLivery	X				5.1	X	

2.9 Get Protocol by Date

2.9.1 Resource Path

`https://wwwlab.cs.univie.ac.at/~bartlp20/rest/index.php/protocol/bydate/{date}`

2.9.2 Purpose

Returns protocols of given date x in 2_data.xml. Example date ist '2017-06-12'.

2.9.3 SOAP operation signature

`getProtocolByDate`

2.9.4 Example Call

```
curl --request GET \
  --url 'https://wwwlab.cs.univie.ac.at/~bartlp20/rest/index.php/planes/
    bylivery/' \
  --header 'Content-Type: application/json' \
  --data '{
    "liveries": [
      "eurowhite",
      "jellybean"
    ]
  }'
```

2.9.5 Return data

```
<?xml version="1.0"?>
<Protocols>
  <Protocol protocolid="UFGUPCARTE">
    <Testdate>2017-06-12</Testdate>
    <results>
      <test>wings</test>
      <vibrations loc="cockpit">
        <scale>low</scale>
        <critical>false</critical>
      </vibrations>
    ...
```

2.9.6 Covered Requirements

Operation	R1	R2	R3	R4	R5	R6	R7
getProtocolByDate				X			X

2.10 Get Testdate Of Protocol

2.10.1 Resource Path

<https://wwwlab.cs.univie.ac.at/~bartlp20/rest/index.php/testdate/byprotocol/protocolId>

2.10.2 Purpose

Returns test date of protocol with protocolId = x.

2.10.3 SOAP operation signature

getTestdateOfProtocol

2.10.4 Example Call

```
curl --request GET \  
  --url 'https://wwwlab.cs.univie.ac.at/~bartlp20/rest/index.php/testdate/  
        byprotocol/KPTDTWVNUZ'
```

2.10.5 Return data

"2017-02-19"

2.10.6 Covered Requirements

Operation	R1	R2	R3	R4	R5	R6	R7
getTestdateOfProtocol			X			X	

3 POST Endpoints

3.1 Post a new bar to plane

3.1.1 Resource Path

<https://wwwlab.cs.univie.ac.at/~bartlp20/rest/index.php/planes/bars/postnewbar>

3.1.2 Purpose

Creates a new bar resource for plane

3.1.3 SOAP operation signatur

getBarsOfPlane

3.1.4 Example Call

```
curl --request POST \
  --url 'https://wwwlab.cs.univie.ac.at/~bartlp20/rest/index.php/planes/bars
    /postnewbar' \
  --header 'Content-Type: application/json' \
  --data '{
    "registration": "OLH2122",
    "teamid": "50",
    "Bar": {
      "Minifridges": {
        "Amount": 5
      },
      "Glasses": {
        "Type": "flute",
        "Amount": 10
      }
    }
  }'
```

3.1.5 Return data

200 if successful, 404 if error

3.1.6 Covered Requirements

Operation	R1	R2	R3	R4	R5	R6	R7
getBarsOfPlane			X			X	

4 PUT Endpoints

4.1 Update the Address Of Customer owning Plane

4.1.1 Resource Path

<https://wwwlab.cs.univie.ac.at/~bartlp20/rest/index.php/planes/update>

4.1.2 Purpose

Updates the address of the customer who owns the plane specified by the registration in the body.

4.1.3 SOAP operation signature

getCustomerOfPlane

4.1.4 Example Call

```
curl --request PUT \
  --url 'https://wwwlab.cs.univie.ac.at/~bartlp20/rest/index.php/planes/
  update' \
  --header 'Content-Type: application/json' \
  --data '{
    "registration": "OLH2122",
    "Address": {
      "@attributes": {
        "zip": "21360",
        "city": "South Erintown"
      },
      "Street": "Spencer Shoals",
      "Housenumber": "450"
    }
  },'
```

4.1.5 Return data

Code 204 on success, 404 on error

4.1.6 Covered Requirements

Operation	R1	R2	R3	R4	R5	R6	R7
getCustomerOfPlane			X			X	

5 PATCH Endpoints

5.1 Patch Flowrate of Toilets

5.1.1 Resource Path

<https://wwwlab.cs.univie.ac.at/~bartlp20/rest/index.php/planes/toiletunits/betweenflowrate>

5.1.2 Purpose

This patch searches and changed the flowrate of all toilet units within an interval supplied by the request body. Furthermore checks if a specific registration was in the affected toilets, if not will add a new toilet with given specs.

5.1.3 SOAP operation signature

getToiletsBetweenFlowrates

5.1.4 Example Call

```
curl --request PATCH \
  --url 'https://wwwlab.cs.univie.ac.at/~bartlp20/rest/index.php/planes/
        toiletunits/betweenflowrate' \
  --header 'Content-Type: application/json' \
  --data '{
    "registration": "DAJ7178",
    "newFlowrate": 50,
    "flowrates": {
      "minRate": 2.0,
      "maxRate": 2.2
    }
  }'
```

5.1.5 Return data

```
{
  "Unit": [
    {
      "operation": "modified",
      "registration": "AMQ2317",
      "unitid": "35ec9671-2a96-46b7-8b0e-67c479683cce"
    },
    {
```

```

        "operation": "modified",
        "registration": "ARX5204",
        "unitid": "f1aaf0df-98e8-43b8-aaaf-7877f102fc0b"
    },
    {
        "operation": "modified",
        "registration": "DDR5762",
        "unitid": "8a846cfd-e070-45f8-a948-e989a5287bfe"
    },
    {
        "operation": "modified",
        "registration": "LQV1108",
        "unitid": "25378482-369c-4c71-8e3e-dd1a5d370db3"
    },
    {
        "operation": "modified",
        "registration": "HSV8858",
        "unitid": "6715cd21-08f2-4140-aa6c-c380376ea5c6"
    },
    {
        "operation": "created",
        "registration": "DAJ7178",
        "unitid": "0d59b5f6-bea6-4b7c-b630-04ae5c85e4df"
    }
]
}

```

5.1.6 Covered Requirements

Operation	R1	R2	R3	R4	R5	R6	R7
getToiletsBetweenFlowrates		X			5.2		X

5.2 Patch Beverage

5.2.1 Resource Path

<https://wwwlab.cs.univie.ac.at/~bartlp20/rest/index.php/planes/bars/patchmeadrink>

5.2.2 Purpose

Was on accident, doesn't cover a xpath from 5a. Changes beverage in bar or adds if not existent.

5.2.3 SOAP operation signature

None, is new

5.2.4 Example Call

```
curl --request PATCH \  
  --url 'https://wwwlab.cs.univie.ac.at/~bartlp20/rest/index.php/planes/bars/  
    /patchmeadrink' \  
  --header 'Content-Type: application/json' \  
  --data '{  
    "unitid": "69e605d0-08cd-4a6f-bc0f-e22192dcc856",  
    "Beverage": [  
      {  
        "Drink": "Spirits",  
        "Cost": 500  
      },  
      {  
        "Drink": "Beer",  
        "Cost": 100  
      }  
    ]  
  }'
```

5.2.5 Return data

200 if success, 404 if error

5.2.6 Covered Requirements

None

6 DELETE Endpoints

6.1 Delete customer by Id

6.1.1 Resource Path

<https://wwwlab.cs.univie.ac.at/~bartlp20/rest/index.php/customers/delete/customerId>

6.1.2 Purpose

Deletes a customer by Id

6.1.3 SOAP operation signature

getCustomerById

6.1.4 Example Call

```
curl --request DELETE \  
  --url 'https://wwwlab.cs.univie.ac.at/~bartlp20/rest/index.php/customers/  
  delete/19'
```

6.1.5 Return data

Code 204 on success, 404 on error

6.1.6 Covered Requirements

Operation	R1	R2	R3	R4	R5	R6	R7
getCustomerById		X				X	

Requirement summary

Operation	R1	R2	R3	R4	R5	R6	R7
getTeamsByEmployees		X					X
getTeamsByCountries	X				5.1		X
getSmallestTeamID							
getCustomerByID		X				X	
getCustomersByZip		X				X	
getStreetsByZip		X				X	
getBarsOfPlane			X			X	
getBarMinifridgeAndColor					5.3		X
getPlanesByColor	X				5.1		X
getPlanesByLivery	X				5.1	X	
getProtocolByDate				X			X
getToiletsBetweenFlowrates		X			5.2		X
getCustomerOfPlane			X			X	
getTestdateOfProtocol			X			X	