

Verteilte Systeme Master Labor

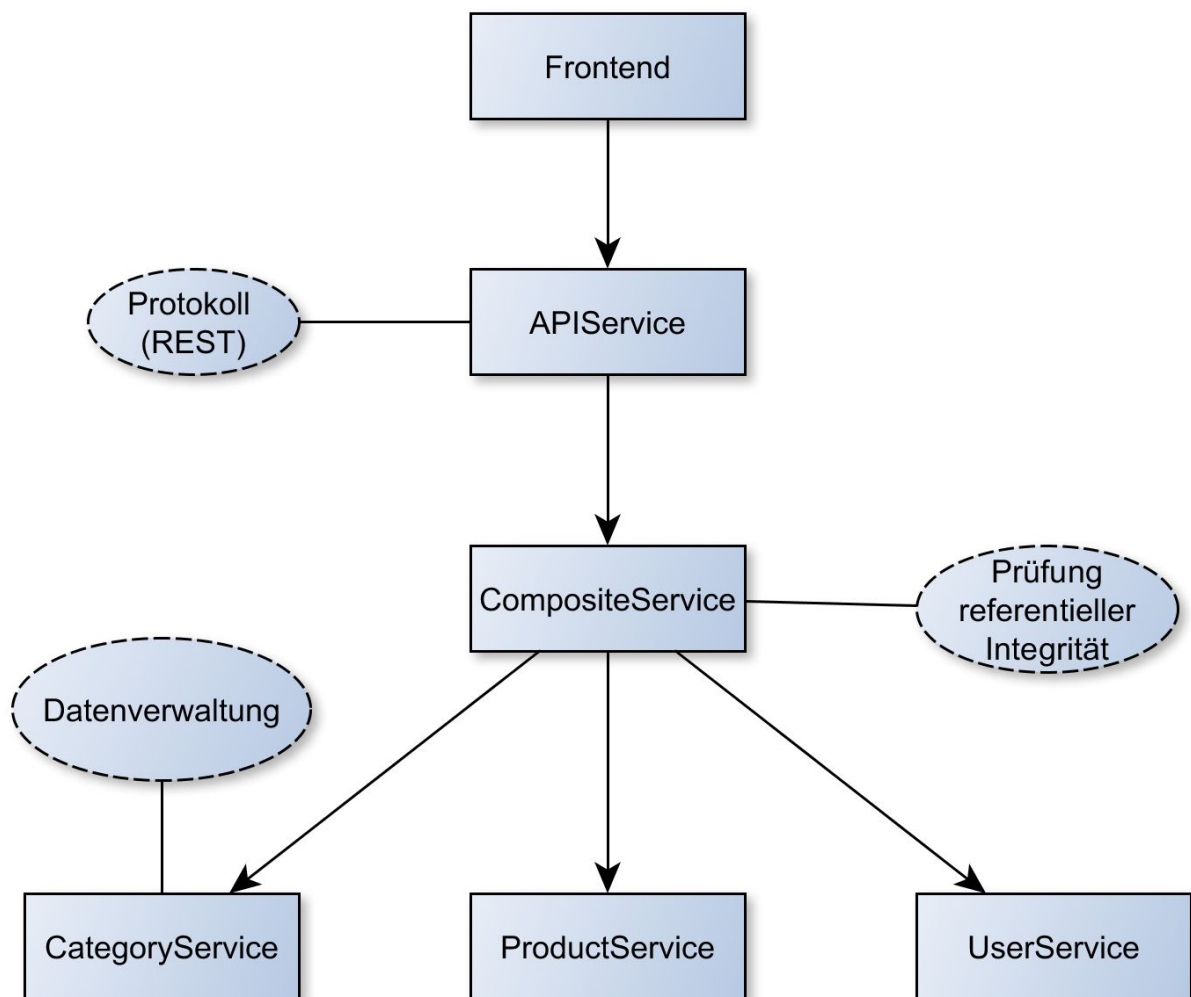
Aufgabe 2 - Microservice Architektur & REST API

Team:

Böhlke, Bela Hraban; MatrNr.: 64048
Chaudhry, Sajeel Ahmed; MatrNr.: 63427
Göckel, Moritz; MatrNr.: 63557
Krieger, Peter; MatrNr.: 60832
Machurich, Mirko; MatrNr.: 63853

1. Übersicht	2
2. Gesamt API	3
3. Details	4

1. Übersicht



2. Gesamt API

categories



POST	/categories	Adds a new category
GET	/categories	Queries categories. If no name is provided all categories will be returned
DELETE	/categories/{id}	Deletes a category
GET	/categories/{id}	Retrieves a category

userRole



GET	/users	Get users
POST	/users	Create user
GET	/users/{id}	Get user by Id
PUT	/users/{id}	Set user by Id
DELETE	/users/{id}	Delete user with Id
GET	/roles	Get roles
POST	/roles	Create role
GET	/roles/{id}	Get role by Id
PUT	/roles/{id}	Set role by Id
DELETE	/roles/{id}	Delete role with Id

products



POST	/products	Adds a new product
GET	/products	Queries products. If no name is provided all products will be returned
DELETE	/products/{id}	Deletes a product
GET	/products/{id}	Retrieves a product

Models		⌵
Product	>	⌵
Category	>	⌵
User	>	⌵
Role	>	⌵

3. Details

Models		⌵
Product	⌵ { id integer(\$int64) name string }	⌵
Category	⌵ { id integer(\$int64) name string }	⌵
User	⌵ { id integer(\$int64) username string firstName string lastName string roleID integer(\$int32) }	⌵
Role	⌵ { id integer(\$int64) typ string level integer(\$int32) }	⌵

POST

/products

Adds a new product

Parameters

Try it out

Name	Description
name * required string (header)	The name of the product

Responses

Response content type application/json

Code	Description
200	<div>Successful operation</div> <div>Example Value Model</div> <div><pre>{ "id": 0, "name": "string" }</pre></div>
400	<div>Invalid input</div>
405	<div>Product already exists</div>

GET **/products** Queries products. If no name is provided all products will be returned

Parameters

Try it out

Name	Description
name string (header)	The name of the product

Responses

Response content type **application/json**

Code	Description
200	<div>Successful operation</div> <div>Example Value Model</div> <pre>[{ "id": 0, "name": "string" }]</pre>

DELETE **/products/{id}** Deletes a product

Parameters

Try it out

Name	Description
id * required integer(\$int64) (path)	The id of the to be deleted product

Responses

Response content type **application/json**

Code	Description
200	<div>Successful operation</div>
405	<div>Product does not exist</div>

GET

/products/{id}

Retrieves a product

Parameters

Try it out

Name	Description
id * required integer(\$int64) (path)	The id of the to be retrieved product

Responses

Response content type

application/json

Code	Description
200	<div>Successful operation</div> <div><div>Example Value</div><div>Model</div><div><pre>{ "id": 0, "name": "string" }</pre></div></div>
405	<div>Product does not exist</div>

POST

/categories

Adds a new category

Parameters

Try it out

Name	Description
name * required string (header)	The name of the category

Responses

Response content type

application/json

Code	Description
200	<div>Successful operation</div> <div>Example Value Model</div> <div><pre>{ "id": 0, "name": "string" }</pre></div>
400	<div>Invalid input</div>
405	<div>Category already exists</div>

GET

/categories

Queries categories. If no name is provided all categories will be returned

Parameters

Try it out

Name	Description
name string (header)	The name of the category

Responses

Response content type

application/json

Code	Description
200	<div>Successful operation</div> <div>Example Value Model</div> <div><pre>[{ "id": 0, "name": "string" }]</pre></div>

DELETE

/categories/{id}

Deletes a category

Parameters

Try it out

Name	Description
id * required integer(\$int64) (path)	The id of the to be deleted category

Responses

Response content type

application/json

Code	Description
200	<div>Successful operation</div>
405	<div>Category does not exist</div>

GET

/categories/{id}

Retrieves a category

Parameters

Try it out

Name	Description
id * required integer(\$int64) (path)	The id of the to be retrieved category

Responses

Response content type

application/json

Code	Description
200	<div>Successful operation</div> <div>Example Value Model</div> <div><pre>{ "id": 0, "name": "string" }</pre></div>
405	<div>Category does not exist</div>

GET /users Get users	
Parameters Try it out	
Name	Description
username	
string	
(query)	
firstname	
string	
(query)	
lastname	
string	
(query)	
roleId	
integer(\$int64)	
(query)	
Responses Response content type application/json	
Code	Description
200	<div>success</div> <div>Example Value Model</div> <div>[{ "id": 0, "username": "string", "firstName": "string", "lastName": "string", "roleID": 0 }]</div>
403	<div>error</div>

POST

/users

Create user

Parameters

Try it out

Name	Description
user <small>★ required</small> (body)	<div>Example Value Model</div> <pre>{ "id": 0, "username": "string", "firstName": "string", "lastName": "string", "roleID": 0 }</pre> <div>Parameter content type application/json</div>

Responses

Response content type application/json

Code	Description
200	success
403	error

GET

/users/{id}

Get user by Id

Parameters

Try it out

Name	Description
Id <small>★ required</small> integer(\$int64) (path)	

Responses

Response content type application/json

Code	Description
200	<div>success</div> <div>Example Value Model</div> <pre>{ "id": 0, "username": "string", "firstName": "string", "lastName": "string", "roleID": 0 }</pre>
403	error

PUT

/users/{id} Set user by Id

Parameters

Try it out

Name	Description
Id * required integer(\$int64) (path)	
user * required (body)	<div>Example Value Model</div> <pre>{ "id": 0, "username": "string", "firstName": "string", "lastName": "string", "roleID": 0}</pre> <div>Parameter content type application/json</div>

Responses

Response content type application/json

Code	Description
200	success
403	error

DELETE

/users/{id} Delete user with Id

Parameters

Try it out

Name	Description
Id * required integer(\$int64) (path)	

Responses

Response content type application/json

Code	Description
200	Success
403	error

GET

/roles

Get roles

Parameters

Try it out

Name	Description
typ	
string	
(query)	
level	
integer(\$int32)	
(query)	

Responses

Response content type

application/json

Code	Description
200	<div>success</div> <div>Example Value Model</div> <div><pre>[{ "id": 0, "typ": "string", "level": 0 }]</pre></div>
403	<div>error</div>

POST

/roles

Create role

Parameters

Try it out

Name	Description
role ★ required	
(body)	

Example Value | Model

```
{  
  "id": 0,  
  "typ": "string",  
  "level": 0  
}
```

Parameter content type

application/json

Responses

Response content type

application/json

Code	Description
200	<div>success</div>
403	<div>error</div>

GET

/roles/{id} Get role by Id

Parameters

Try it out

Name	Description
Id * required integer(\$int64) (path)	

Responses

Response content type application/json

Code	Description
200	<div>success</div> <div>Example Value Model</div> <div><pre>{ "id": 0, "typ": "string", "level": 0 }</pre></div>
403	<div>error</div>

PUT

/roles/{id} Set role by Id

Parameters

Try it out

Name	Description
Id * required integer(\$int64) (path)	
role * required (body)	<div>Example Value Model</div> <div><pre>{ "id": 0, "typ": "string", "level": 0 }</pre></div> <div>Parameter content type application/json</div>

Responses

Response content type application/json

Code	Description
200	<div>success</div>
403	<div>error</div>

DELETE

/roles/{id} Delete role with Id

Parameters

Try it out

Name	Description
Id * required integer(\$int64) (path)	

Responses

Response content type application/json

Code	Description
200	Success
403	error