VIS-Labor Webshop API

API-Beschreibung der Webshop-Implementierung im Rahmen des VIS-Labors, November 2015.

Version 0.8.0

Contact information
Sebastian Schork & Vitali Dettling

Paths

/account/{user}

GET /account/{user}

login Account

Description

handles the login of a given user account

Parameters

Name	Located in	Description	Required	Schema
user	path	user name	Yes	⇄ string
pass	body	user's password	Yes	⇄ string

Responses

Code	Description
200	login procedure successfull
401	unauthorized login
404	user not found
500	internal service error

Try this operation

PUT /account/{user}

Description

updates existing account

Parameters

Name	Located in	Description	Required	Schema
user	path	user name of the account	Yes	⇄ string
pass	body	user' password of the account	Yes	⇄ string

Responses

Code	Description
200	account successfully updated
401	authorization error
404	user not found
500	internal server error

Try this operation

DELETE /account/{user}

Description

deletes an account permanently

Parameters

Name	Located in	Description	Required	Schema
user	path	user name of the account to be deleted	Yes	⇄ string
pass	body	user password of the account to be deleted	Yes	⇄ string

Responses

Code	Description
200	account deleted
401	authorization error
404	user not found
500	internal server error

Try this operation

/account

POST /account

create Account

Description

creates a new account

Parameters

Name	Located in	Description	Required	Schema
user	body	user name of the new account	Yes	⇄ string
pass	body	user' password of the new account	Yes	⇄ string

Responses

Code	Description
201	account successfully created
500	internal server error

Try this operation

/category

POST /category

add Category

Description

add new category

Parameters

Name	Located in	Description	Required	Schema
id	body	ID of new category	Yes	⇄ string

Responses

Code	Description
201	category successfully added
500	internal server error

Try this operation

GET /category

Description

return list of all existing categories

Responses

Code	Description	Schema
200	array of strings	⇄ ▶ [ ]
500	internal server error	

Try this operation

/category/{id}

DELETE /category/{id}

Description

delete an existing category

Parameters

Name	Located in	Description	Required	Schema
id	path	ID of the category to be deleted	Yes	⇄ string

Responses

Code	Description
200	category successfully deleted
404	category not found
500	internal server error

Try this operation

/product/{id}

GET /product/{id}

Description

return the current product information

Parameters

Name	Located in	Description	Required	Schema
id	path	id of the product	Yes	⇄ string

Responses

Code	Description	Schema

200	product object	⇄ ▼ {
		  data: ▼ **Product** {
		    product object
		    id: integer
		    price: double
		    info: string
		    category: string
		  }
		}

404	product not found	
500	internal server error	

Try this operation

PUT /product/{id}

update Product

Description

update an existing product

Parameters

Name	Located in	Description	Required	Schema
id	path	product id to be updated	Yes	⇄ undefined
product	body	new product specification to update	Yes	⇄

```
▼ {
    data: ▼ Product {
            product object
            id:       integer
            price:    double
            info:     string
            category: string
        }
}
```

Responses

Code	Description
201	product successfully updated
404	product not found
500	internal server error

Try this operation

DELETE /product/{id}

Description

delete a product

Parameters

Name	Located in	Description	Required	Schema
id	path	id of the product to be deleted	Yes	⇄ string

Responses

Code	Description
200	product successfully deleted
404	product not found
500	internal server error

Try this operation

/product/

POST /product/

Description

add a new product

Parameters

Name	Located in	Description	Required	Schema
product	body	ID of the product	Yes	⇄

```
▼ {
    data: ▼ Product {
              product object
              id:       integer
              price:    double
              info:     string
              category: string
          }
  }
```

Responses

Code	Description
201	product successfully added
500	internal server error

Try this operation

GET /product/

get Products

Description

return list of all existing products

Responses

Code	Description	Schema

▼ [
   data: ▼ **Product** {
        product object
        id: integer

200   array of product objects   ⇄   price: double
        info: string
        category: string
    }
]

---

500   internal server error

Try this operation

Models

Product

▼ **Product** {
  product object
  id: integer
⇄ price: double
  info: string
  category: string
}