**BEEZZY API – “BEEZZAPI”**

General structure  
To call a VK API method you need to make **POST**, **PUT**, **GET** or **DELETE** request to the specified URL using **HTTPS** protocol  
[https://beezzy.herokuapp.com/api/’’’METHOD\_NAME’’’?’’’PARAMETERS’’’&access\_token=ACCESS\_TOKEN](https://beezzy.herokuapp.com/api/'''METHOD_NAME'''?'''PARAMETERS'''&access_token=ACCESS_TOKEN)

**List of methods**

/users/**id** (GET)

**Parameters:**   
 fields – email, role, active, shops (list of coma-separated words)  
**Result:** an user object

/users (PUT)

**Parameters:**  
 email – user’s email  
 password - md5 of user password  
 role – name of role  
 active – is active  
**Result:** user id  
  
/users (POST)

**Parameters:**  
 id – user id  
 email – new user’s email  
 password - new md5 of user password  
 old\_password – md5 of user password (only to change password)  
 role – name of role  
 active – is active  
**Result:** boolean

/users/**id/shops** (GET)

**Parameters:**   
 fields – name, description, owner, goods (list of coma-separated words)  
**Result:** a list of shop objects

/shops/**id** (GET)

**Parameters:**   
 fields – name, description, owner, goods (list of coma-separated words)  
**Result:** an shop object  
  
  
/shops (PUT)

**Parameters:**  
 name – name of the shop   
 description - descript ion of the shop  
**Result:** shop id  
  
/shops/ (POST)

**Parameters:**  
 id – id of shop  
 name – name of the shop   
 description - descript ion of the shop  
**Result:** boolean  
  
/shops/**id** (DELETE)

**Parameters:** shop – name of the shop   
**Result:** boolean

/shops/**id**/goods (GET)

**Parameters:**  
 fields - name, description, category, varieties  
**Result:** a list of goods objects

/goods/**id** (GET)

**Parameters:**  
 fields - name, description, category, varieties  
**Result:** an goods object  
  
/goods (PUT)

**Parameters:**   
 shop\_id  
 name – name of the goods  
 description - descript ion of the goods  
 category\_id – id of the last subcategory  
**Result:** goods id  
  
/goods (POST)

**Parameters:**   
 id – goods id  
 name – new name of the goods  
 description - new description of the goods  
 category\_id – new subcategory id  
**Result:** boolean

/goods/**id** (DELETE)

**Parameters:** name – name of the goods  
**Result:** boolean   
  
/varieties/**id** (GET)

**Parameters:**   
 fields - description, amount, goods, p\_price, s\_price, code  
**Result:** an variety object  
  
/varieties (PUT)

**Parameters:**   
 description   
 amount  
 goods\_id  
 p\_price   
 s\_price  
 code  
**Result:** variety id  
  
/varieties (POST)

**Parameters:**   
 id  
 goods\_id  
 description   
 amount   
 p\_price   
 s\_price  
 code  
**Result:** boolean  
  
/varieties/**id** (DELETE)

**Parameters:** name – name of the variety  
**Result:** boolean

/categories/**id** (GET)

**Parameters:**   
 fields - name, parent, children  
**Result:** an category object  
  
/shops/**id**/categories (GET)

**Parameters:**   
 only\_parents – returns only parent categories (default true)  
**Result:** a list of all categories of the specified shop

/categories/ (PUT)

**Parameters:**   
 shop\_id  
 name   
 parent\_id (optional)   
**Result:** category id  
  
/categories/ (POST)

**Parameters:** id  
 name  
**Result:** boolean  
  
/categories/**id** (DELETE)

**Parameters:**   
 confirm – to delete al goods and varieties (default false)  
 (if there is any goods of this category it wouldn’t be deleted)  
**Result:** boolean