

starRating:1-5

recipe/reviewScore: 1-10

## 1.Endpoint /search-and-filter/categories

GET

Input: -

Output: toate categoriile

Status code 200

```
{
  "categoryTypes": [
    "Country",
    "Nationality",
    "GeneralCategory",
    "Category",
    "HealthLabel",
    "HealthRestriction",
    "Ingredient",
    "Fruit",
    "Vegetable",
    "NutritionInformation"
  ]
}
```

## 2.Endpoint /search-and-filter/categories/<category\_name>

GET

Input: category name (ex: Country)

Output: category instances list

Status code 200

```
{
  "Country": [
    "Poland",
    "Canada",
    "Bulgaria",
    "Romania",
    "Italy",
    ...
  ]
}
```

Exceptions:

1. If category\_name not found 404:

```
{
  "message": "Category gfgfg doesn't exist"
}
```

In case of filtering, for all endpoints (beverages, recipes, etc.) a single query parameter filter is used (with value a list of jsons, each json of type {"name": "..", "op": "..", "val": ".."}). Filtering schemas for each entity available on github.

### 3.Endpoint /search-and-filter/beverages

#### GET

##### Filtering criteria:

containsTextInBeverageName=[<beverageNameText>] (+negation)

beverageType=[<bigCategory>] (+negation)

provenanceCountry=[<country>] (+negation)

category=[<category>] (+negation)

containsHealthLabel=[<healthLabel>] (+negation)

containsTextInIngredientName=[<ingredientNameText>] (+negation)

isMadeOfFruit=[<fruit>](+negation)

isMadeOfVegetable=[<vegetable>](+negation)

LTEFilter, GTFilter, EQ. for Percentage of Nutritional Information (VitaminA, VitaminC, etc.)

LTEFilter, GTFilter, EQ for quantity

LTEFilter, GTFilter, EQ for calories

Input example:

```
http://127.0.0.1:5000/search-and-filter/beverages/?filter=[{"name": "hasProvenanceCountry", "op": "nc", "val": ["France", "Romania"]}, {"name": "beverageName", "op": "nc", "val": ["Choco", "Latte"]}, {"name": "isMadeOfFruit", "op": "c", "val": ["Apple"]}, {"name": "isMadeOfVegetable", "op": "c", "val": ["Carrot"]}]
```

Output example:(only general info regarding beverages)

```
[
  {
    "beverageImageUrl":
      "https://images.openfoodfacts.org/images/products/433/718/563/4727/front_de.11.400.jpg",
    "beverageName": "multivitaminsaft",
    "beverage": "4337185634727",
    "beverageUrl": "https://world.openfoodfacts.org/product/4337185634727"
  },
  {
    "beverageName": "frokostjuice",
    "beverage": "7032069712160",
    "beverageUrl": "https://world.openfoodfacts.org/product/7032069712160"
  },
  {
    "beverageImageUrl":
      "https://images.openfoodfacts.org/images/products/900/460/710/1208/front_en.4.400.jpg",
    "beverageName": "bioAppleUndCarrotJuiceNaturallyCloudy",
    "beverage": "9004607101208",
    "beverageUrl": "https://world.openfoodfacts.org/product/9004607101208"
  }
]
```

]

#### 4.Endpoint /search-and-filter/beverages/<beverage\_code>

**Input: beverage code (ex: 3368951148187)**

**Output:** All info regarding the beverage

```
{
  "hasGeneralCategory": [
    "Tea"
  ],
  "hasIngredient": [
    "almonds10",
    "caramelLaws5",
    "genmaicha15",
    "teaVert64",
    "flavoursNatural"
  ],
  "hasProvenanceCountry": [
    "France"
  ],
  "quantity": "100",
  "beverageName": "TeaAlmondCaramel",
  "ingredientImageUrl":
    "https://images.openfoodfacts.org/images/products/336/895/114/8187/ingredients_fr.11.400.jpg",
  "hasCategory": [
    "hot-beverages",
    "beverages",
    "plant-based-beverages",
    "plant-based-foods-and-beverages",
    "teas"
  ],
  "beverageImageUrl":
    "https://images.openfoodfacts.org/images/products/336/895/114/8187/front_fr.21.400.jpg",
  "hasHealthLabel": [
    "nuts",
    "milk"
  ],
  "nutritionImageUrl":
    "https://images.openfoodfacts.org/images/products/336/895/114/8187/nutrition_fr.23.400.jpg",

```

```

        "beverage": "3368951148187",
        "isMadeOfFruit": [],
        "isMadeOfVegetable": [],
        "beverageUrl": "https://world.openfoodfacts.org/product/3368951148187"
    }

```

#### Exception:

##### 1. Beverage code invalid 404:

```

{
    "message": "Beverage code 3368 doesn't exist"
}

```

## 5.Endpoint /search-and-filter/recipes

### Filtering criteria:

containsTextInRecipeName=[<recipeNameText>] (+negation)

correspondsToBeverageType=[<bigCategory>] (+negation)

LTEFilter, GTFilter, EQ. for Percentage of Nutritional Information (VitaminA, VitaminC, etc.)

LTEFilter, GTFilter, EQ for quantity

LTEFilter, GTFilter, EQ for calories

containsTextInIngredientName=[<ingredientNameText>] (+negation)

containsHealthLabel=[<healthLabel>] (+negation)

Input example: [http://127.0.0.1:5000/search-and-filter/recipes/?filter=\[{"name": "hasHealthLabel", "op": "c", "val": \["milk", "for-vegetarians", "peanuts-free"\]}, {"name": "fatContent", "op": "lteq", "val": "10"}, {"name": "recipeName", "op": "c", "val": \["Orange", "SMOOTHIE"\]}"\]](http://127.0.0.1:5000/search-and-filter/recipes/?filter=[{)

Output: (only general info regarding recipes)

```

[
    {
        "recipeName": "orangeStrawberrySmoothie",
        "recipeShareAs":
"http://www.edamam.com/recipe/orange-strawberry-smoothie-1bac9cff4d4305865dc482f4aa04f3a5/juice/alcohol-free",
        "recipeImageUrl":
"https://www.edamam.com/web-img/160/1608a226178750bfbd11e5d0074b3503.jpg",
        "beverage": "Juice",
        "recipe": "recipeNumber10139"
    },
    {
        "recipeName": "orangeBananaSmoothie",

```

```

        "recipeShareAs":
"http://www.edamam.com/recipe/orange-banana-smoothie-7cc800fcelc4eafebfb86bba72b8cca5/j
uice/alcohol-free",
        "recipeImageUrl":
"https://www.edamam.com/web-img/583/583b50cd2afbe2671d0738eebfa48ae8.jpg",
        "beverage": "Juice",
        "recipe": "recipeNumber10361"
    },
    ..
]

```

## 6.Endpoint /search-and-filter/recipes/<recipe\_code>

Input example: <http://127.0.0.1:5000/search-and-filter/recipes/recipeNumber17490>  
(beverage code)

Output example: (output=all info of recipe with given code)

```

{
    "vitaminB6Content": "3.623678333334406E-4",
    "hasIngredient": [

"1CupOfOrganicNonfatUnsweetenedAlmondMilkWeLoveLoveLoveBlueDiamondAlmondBreeze",
        "1TableSpoonOfHoney",
        "1TeaSpoonOfGroundCinnamon",
        "1TeaSpoonOfOrganicCocoa",
        "1TeaSpoonOfPureVanillaExtract",
        "12FrozenBananaBroken",
        "12NavalOrangePeeledAndSliced",
        "14CupOfNonfatPlainGreekYogurt",
        "23PiecesOfRipeStrawberriesChopped"
    ],
    "fiberContent": "7.2252833333366957",
    "recipe": "recipeNumber17490",
    "fatContent": "3.579073333345783",
    "isRecipeFor": "Tea",
    "calciumContent": "0.6187331666667829",
    "ironContent": "0.001803286666679262",
    "hasHealthLabel": [
        "alcohol-free",
        "celery-free",
        "fish-free",
        "for-pescatarians",
        "for-vegetarians",
        "lupine-free",
        "molluscs-free",
        "mustard-free",

```

```

        "oil-free",
        "peanuts-free",
        "pork-free",
        "red-meat-free",
        "gluten",
        "salt",
        "soybeans",
        "wheat-derivatives",
        "none"
    ],
    "recipeSource": "bestdessertsonlinecom",
    "potassiumContent": "0.6978275000013849",
    "recipeUrl":
"http://www.bestdessertsonline.com/healthy-dessert-recipe-strawberry-orange-cinnamon-co
coa-smoothie/",
    "sugarContent": "35.10340416666826",
    "recipeUri":
"http://www.edamam.com/ontologies/edamam.owl#recipe_a26e7d6e967af460ea3178b501900fb3",
    "quantity": "4",
    "vitaminAContent": "1.0406333333333333E-5",
    "vitaminCContent": "0.05982975",
    "cholesterolContent": "0.0037",
    "calories": "259.1211666668739",
    "carbohydrateContent": "49.88040500005262",
    "sodiumContent": "0.1998085833333524",
    "recipeName": "strawberryOrangeCinnamonCocoaSmoothieRecipes",
    "magnesiumContent": "0.06289975000045347",
    "recipeShareAs":
"http://www.edamam.com/recipe/strawberry-orange-cinnamon-cocoa-smoothie-recipes-a26e7d6
e967af460ea3178b501900fb3/tea/alcohol-free",
    "proteinContent": "10.94830000001781",
    "phosphorusContent": "0.2519686333340004",
    "recipeImageUrl":
"https://www.edamam.com/web-img/cbc/cbc72505b0dd67d2f0b2d1c9f5272755",
    "vitaminDContent": "0",
    "vitaminEContent": "0.006001757333333422",
    "vitaminKContent": "1.884075000002272E-6",
    "vitaminB12Content": "5.549999999999999E-7",
    "zincContent": "9.528541666728553E-4"
}

```

## 7.Endpoint /search-and-filter/offers

### Query parameters:

**hasOfferOwner**= email

**hasOfferOwner**!= [email]

**isOfferFor**= bigcategory

**isOfferFor** != [bigcategory]

**offerDescription** C [text]

**offerDescription** Nc [text]

LTEFilter, GTFilter, EQ. for **offerPrice**

**priceCurrency**=[currency]

**priceCurrency**!= [currency]

**offerQuantity** = quantitytype

**offerQuantity** != [quantity types]

FiltersForTheUser (familyName, givenName, livesInCountry, hasNationality, gender, age)

**Input:** `http://127.0.0.1:5000/search-and-filter/offers?filter=[{"name":"hasOfferOwner","op":"neq","val":["bhaycockm_at_e-recht24.de","alebatteurz_at_wisc.edu"]}, {"name":"isOfferFor","op":"neq","val":["Tea","Coffee","Milkshake"]}, {"name":"familyName","op":"eq","val":"a"}, {"name":"livesInCountry","op":"neq","val":["France","India"]}, {"name":"hasNationality","op":"eq","val":"Romanian"}]`

### Output :

```
[
  {
    "offer":
      "http://www.semanticweb.org/ontologies/bem#offerhbalassill_at_ucoz.ruSmoothie",
      "priceCurrency": "EUR",
      "offerPrice": "17.5",
      "offerDescription": "I'm very proud of my smoothie. It's made with
high-quality organic fruits, natural products, and it has a lot of sweet flavours. I
think it's the best thing to drink when you feel like you need some energy boost. I
also offer various smoothie flavours: strawberry, banana, apple, pear, cherry.",
      "hasOfferOwner":
        "http://www.semanticweb.org/ontologies/bem#hbalassill_at_ucoz.ru",
      "offerQuantity": "1.51",
      "isOfferFor": "Smoothie"
  }
]
```

## 8.Endpoint /search-and-filter/reviews

### Query parameters:

**isReviewFor** = <beverage\_code>

**isReviewFor** != [<beverage\_code>]

**hasReviewOwner** = <review\_owner>

**hasReviewOwner** = [<review\_owner>]

**reviewDescription** C [text]

**reviewDescription** NC [text]

**starRating** LTEFilter, GTFilter, EQ.

FiltersForTheUser (familyName, givenName, livesInCountry, hasNationality, gender, age)

**Input:** [http://127.0.0.1:5000/search-and-filter/reviews?filter=\[{"name": "givenName", "op": "eq", "val": "an"}, {"name": "gender", "op": "neq", "val": "female"}, {"name": "starRating", "op": "lteq", "val": "4"}, {"name": "reviewDescription", "op": "nc", "val": \["love", "expensive"\]}, {"name": "hasReviewOwner", "op": "neq", "val": \["dwoolaston9p\\_at\\_artisteer.com", "askermanbb\\_at\\_chron.com"\]}"\]](http://127.0.0.1:5000/search-and-filter/reviews?filter=[{)

**Output example:**

```
[
  {
    "hasReviewOwner": "cstubbington6k@infoseek.co.jp",
    "reviewDescription": "It's so refreshing and the taste is out of this
world. It has become my go-to drink whenever I am thirsty.",
    "beverageName": "originalSparklingWaterOriginal",
    "review":
"http://www.semanticweb.org/ontologies/bem#reviewcstubbington6k_at_infoseek.co.jp465670
22376",
    "isReviewFor": "46567022376",
    "starRating": "4"
  },
  {
    "hasReviewOwner": "ktatlow5y@facebook.com",
    "reviewDescription": "It has a terrible taste and I could barely enjoy it.
I think it tastes so gross, so my friends agreed. We tried it together too and nobody
liked the taste. Maybe people have different opinions but I don't think there are many
who actually enjoy the taste of this drink as much as they would like to believe",
    "beverageName": "beerWithoutAlcohol",
    "review":
"http://www.semanticweb.org/ontologies/bem#reviewktatlow5y_at_facebook.com8480000275684
",
    "isReviewFor": "8480000275684",
    "starRating": "2"
  },
  {
    "hasReviewOwner": "porridge7c@google.co.jp",
    "reviewDescription": "It is one of the brands that I've always wanted to
try for a long time. The branding is so sleek and cool, I feel like it's the type of
drink you should be having at a party. But when I finally tried it, it felt like an
overpriced waste of money. It wasn't anything special in terms of taste",
    "beverageName": "siropMentheToSheets",
```



```

        "review":
"http://www.semanticweb.org/ontologies/bem#reviewporridge7c_at_google.co.jp761133066099
9",
        "isReviewFor": "7611330660999",
        "starRating": "3"
    }
]

```

## 9.Endpoint /search-and-filter/users

Query parameters:

FiltersForTheUser

**Input example :** `http://127.0.0.1:5000/search-and-filter/users?filter=[{"name": "familyName", "op": "neq", "val": ["De Bruijn"]}, {"name": "givenName", "op": "neq", "val": ["Adaline"]}, {"name": "hasNationality", "op": "neq", "val": ["Bulgarian"]}, {"name": "livesInCountry", "op": "neq", "val": ["Romania"]}, {"name": "hasHealthRestriction", "op": "c", "val": ["cancer", "diabetes"]}, {"name": "gender", "op": "neq", "val": "male"}]`

**Output example:**

```

[
  {
    "hasCategoryPreference": [
      "juice",
      "legume-milks",
      "rice-milks"
    ],
    "hasHealthRestriction": [
      "diabetes",
      "cancer"
    ],
    "gender": "female",
    "livesInCountry": "http://www.semanticweb.org/ontologies/bem/Germany",
    "familyName": "Oliver",
    "givenName": "Burl",
    "hasNationality": "http://www.semanticweb.org/ontologies/bem/Belizean",
    "email": "boliver31@yandex.ru",
    "age": "73"
  },
  {
    "hasCategoryPreference": [
      "potatoes",
      "artificial-sugar-substitutes"
    ],

```

```

        "hasHealthRestriction": [
            "diabetes",
            "cancer"
        ],
        "gender": "female",
        "livesInCountry": "http://www.semanticweb.org/ontologies/bem/VietNam",
        "familyName": "Trahar",
        "givenName": "Erie",
        "hasNationality": "http://www.semanticweb.org/ontologies/bem/American",
        "email": "etrahar5g@howstuffworks.com",
        "age": "27"
    }
}
]

```

### Exceptions:

#### 1. Operator not allowed (400):

```

{
    "message": "Operator newq cannot be used for criteria familyName "
}

```

### FiltersForTheUser:

```

firstName eq <firstName>
firstName neq [<firstName>]
lastName eq <lastName>
lastName neq [<lastName>]
nationality eq <nationality>
nationality neq [<nationality>]
livesInCountry eq <country>
livesInCountry neq [<country>]
Age eq, gteq, Lteq <age>
gender eq <gender>
gender neq <gender>
healthRestriction c [<healthRestriction>]
healthRestriction nc [<healthRestriction>]
categoryPreference c [<categoryPreference>]
categoryPreference nc [<categoryPreference>]

```

### Sorting:

#### Beverages and recipes all info (sort by only fields available)

[http://127.0.0.1:5000/search-and-filter/beverages/?filter=\[{"name": "hasProvenanceCountry", "op": "nc", "val": \["France", "Romania"\]}, {"name": "beverageName", "op": "nc", "val": \["Choco", "Latte"\]}, {"name": "isMadeOfFruit", "op": "c", "val": \["Apple"\]}, {"name": "isMadeOfVegetable", "op": "c", "val": \["Carrot"\]}\]&sort\\_by=beverageName.d](http://127.0.0.1:5000/search-and-filter/beverages/?filter=[{)

### **Users (sort by all singular fields)**

[http://127.0.0.1:5000/search-and-filter/users?filter=\[{"name": "familyName", "op": "neq", "val": \["De Bruijn"\]}, {"name": "givenName", "op": "neq", "val": \["Adaline"\]}, {"name": "hasNationality", "op": "neq", "val": \["Bulgarian"\]}, {"name": "livesInCountry", "op": "neq", "val": \["Romania"\]}, {"name": "gender", "op": "neq", "val": "male"}\]&sort\\_by=age.a](http://127.0.0.1:5000/search-and-filter/users?filter=[{)

### **Reviews (sort by all singular fields)**

[http://127.0.0.1:5000/search-and-filter/reviews?filter=\[{"name": "givenName", "op": "eq", "val": "an"}, {"name": "gender", "op": "neq", "val": "female"}, {"name": "starRating", "op": "lteq", "val": "4"}, {"name": "reviewDescription", "op": "nc", "val": \["love", "expensive"\]}, {"name": "hasReviewOwner", "op": "neq", "val": \["dwoolaston9p\\_at\\_artisteer.com", "askermanbb\\_at\\_chron.com"\]}\]&sort\\_by=hasReviewOwner.a](http://127.0.0.1:5000/search-and-filter/reviews?filter=[{)

### **Offers (sort by all singular fields)**

[http://127.0.0.1:5000/search-and-filter/offers?filter=\[{"name": "hasOfferOwner", "op": "neq", "val": \["bhaycockm\\_at\\_e-recht24.de", "alebatteurz\\_at\\_wisc.edu"\]}, {"name": "isOfferFor", "op": "neq", "val": \["Tea", "Coffee", "Milkshake"\]}, {"name": "familyName", "op": "eq", "val": "a"}, {"name": "livesInCountry", "op": "neq", "val": \["France", "India"\]}\]&sort\\_by=offerPrice.a](http://127.0.0.1:5000/search-and-filter/offers?filter=[{)