Assignment

FRONTEND IN GUNG

GUNG.

Gung AB

GUNG AB | WWW.GUNG.SE

Assignment

Background

When we work with customization of the Gung platform, data from underlying business systems is sometimes not of the same structure and does not always match the situation where it needs to be used in the customer-specific solution.

Assignment

In the following assignment, data is delivered as if it came from Gung's backend with help functions that extract typical data from the business system. Your assignment is to display this data in a good way using Angular (https://angular.io/). The data we collect in this case are categories including underlying products, as well as product information using two methods as described.

The categories are downloaded in a tree structure, with nodes and sub nodes, where nodes can be either a new category or a product. In order to distinguish a category from a product, we can check if the id starts with a lower case "s".

The task is to create a page where we list all products in a list that can be sorted and filtered.

1. It should be possible to filter on:
    a. Name
    b. Id
    c. Price (Exists in the field extra.PRI on the product object)
2. It should be possible to sort and filter for:
    a. If the product is in stock (Field LGA in the extra-object > 0).
    b. Volume (Field = extra.VOL) in an intervall (from - to).
    c. Category

We assume that the task is solved with Angular (https://angular.io/) using RxJs (https://angular.io/guide/rx-library), as well as Bootstrap for a little easier css styling. If there are any questions, just email: support@gung.se

Lycka till / Good luck !

Extra

To test how your solution works with a more production-like data volume, you can use the categoryService.getAlotOfCategories() function, as well as productService.getRandomProduct(id) instead of the corresponding below.

Help functions

Two Angular services are provided to fetch data. The data is returned wrapped in a RxJs Observable to simulate making real asynchronous API-calls.

categoryService.getCategories() - Retrieves products and categories in the following format:

```
{
     "id": "s0405",
     "name": "Avfettning",
     "children": [
          {
               "id": "AV300-1006",
               "name": "Slangupprullare Avfettning 10m-1/4\""
          },
          .....
     ]
}
```

productService.getProduct(id, callback) - Retrieves products information for one product, for example:

```
{
     "id": "AV300-1006",
     "name": "Slangupprullare Avfettning 10m-1/4\"",
     "extra": {
          "AGA": {
               "APE": "      1",
               "KAT": "TU",
               "LGA": "       0.00",
               "PRI": "    4170.00",
               "TYP": "",
               "VOL": "    0.000",
               "VPE": "   14.500",
               "XP1": "          0",
               "XP2": "          0"
          }
     }
}
```