AMVARA DASHBOARD "tickets"

Extremly fast visualization on any device.

https://www.amvara.de/

February 22, 2021

Contents

1 Introduction

This dashboard loads data from a server via XHR requests and is able to visualize data in under 1 second. We really focused on the speed of visualization. This is why users enjoy using our dashboards at work to review company KPIs day by day. The usage of the dashboard is self explaining - so enduser training is not needed to understand the navigation path.

This documentation describes how to modify the data to fit your needs and how to change the skinning to meet your styleguide.

The dashboard uses Angular. Makeing changes to files inside the src directory requires to compile the project and update the dist directory. See README.md file for details.

The dashboard has a config-file where we specify the data endpoints to fetch data from. If fetching data from theses endpoints fails, it falls back to search files in the assests/reports directory.

The dashboard is able to read saved report output from **IBM Cognos (aka IBM Analytics)**. Which comes in handy as this dashboard uses the IBM Cognos REST API to just fetch the data and avoids to load the IBM Cognos Portal files reducing overhead to a minimum by focusing on speed and performance to visualize the raw data. Fetching data from **REST APIs** is of course not limited to IBM Cognos. We would be happy to know from which services or servers you load the dashboard data. See support chapter on how to contact us.

1.1 Dashboard's Main Page

The main page is divided in two parts. First we have the graphic, it is sorted by ticket type. Next, in the right part (If we are on desktop), or, in the bottom part (If we are on mobile), there are 4 tables with monthly information for each data type. It shows totals and percentages.

Experimental features can be enabled in the help section of the menu.

Data can be ordered by:

- By Priority [1,2,3,4] Being 1 the most critical and 4 the less critical
- By Ticket Type
- By Application/Service
- By Status

Note: All of these can be changed, for example. Instead of Ticket Type, Sales Channel, instead of By priority, by Clothes Size, etc. Everything can be modified according to your needs and the data you want to visualize.

1.2 Deployment and Compiling

The `src`-folder contains the project files that must be compiled. The compiled files reside in `dist`-folder. See `README.md` for details on how to compile the project.

We provide a `docker-compose.yml` file for you, to start everything needed inside a container.

If you have a webserver up and running, you could also just copy the files from "dist" directory to a directory on your webserver.

2 Prepare Skinning

In this section, we will see how to change the dashboard skinning to meet your styleguide.

2.1 Header

Header title, is edited in .src/app/components/ux/header/header.component.html.The original title is “AMVARA Code Is Poetry”, in order to change the title we will edit both span:

```
<div class="header">
    <div class="logo" (click)="navigate('/')" routerLinkActive="active">
        <span>AMVARA</span>
        <span>Code is Poetry</span>
    </div>
```

First span is shown in white color, the second one, in red. (How to change colors is explained later).

2.2 Window Title

Window title is edited on ./src/index.html. In order to edit it, simply change “AMVARA DASHBOARD” inside the title tag, with, “YOUR DASHBOARD TITLE”.

```
<head>
  <meta charset="utf-8">
  <title>AMVARA DASHBOARD</title>
  <meta name="viewport" content="width=device-width, initial-scale=1">
  <meta http-equiv="X-UA-Compatible" content="IE=edge" >
  <link rel="icon" type="image/x-icon" href="favicon.ico?v2">
  <link rel="manifest" href="manifest.json" crossorigin="use-credentials">
  <meta name="theme-color" content="#111111">
</head>
```

2.3 Loading Screen

Loading screen is edited on ./src/index.html.

2.3.1 Loading Screen Title

In order to change the title, div class name must be edited. The part inside “b” tag will be displayed with the color specified in “blue” class.

```
<div class="main">
  <div class="img"></div>
  <div class="name">AMVARA <b class="blue">Code is Poetry</b></div>
</div>
```

2.3.2 Loading Screen CSS

In order to change the “blue” class color, simply, find inside index.html style the blue selector and change the color to your needs:

```
.blue {
  color: #ef3340;
}
```

In order to change the progress bar color, you have to edit ".progress .progress-value" inside index.html styles. If you want the bar to have a different background, change background-color inside .progress. If the color in .progress is the same as the background color of the dashboard, the bar will not have a visible background.

```
.progress {
  width: 180px;
  position: fixed;
  height: 7px;
  top: 0;
  bottom: 0;
  left: 0;
  right: 0;
  margin: auto;
  border-radius: 7px;
  background-color: rgb(0,22,40);
}

.progress .progress-value {
  position: absolute;
  top: 0;
  width: 0%;
  height: 100%;
  border-radius: 7px;
  background-color: #ef3340;
  box-shadow: 0 0 15px #ef3340;
  transition: width .2s ease-in-out;
}
```

If you want to change the background color, edit the background-color property in the body style selector.

```
<style>
  body {
    height: 100%;
    width: 100%;
    background-color: rgb(0,22,40);
    color: rgba(255, 255, 255, .85);
    font-family: "CorpoS", "Segoe UI", "Arial", "sans-serif";
    font-size: 13px;
  }
```

For changing the color of the first part of the title, edit the color property in the name class style selector.

```
.name {
  width: 100%;
  text-align: center;
  height: 60px;
  padding-top: 20px;
  color: white;
  font-size: 1.7em;
  position: fixed;
  left: 0;
  right: 0;
  top: 65px;
  bottom: 0px;
  margin: auto;

}
```

2.4 General Information

General information is edited in ./src/assets/config.json. We will have to edit the contact part:

```
"contacts": {
  "dev": [
    {
      "name": "Name Surname",
      "description": "Description",
      "email": "name.surname@amvara.de",
      "telephones": [
        "+34 612 34 56 78",
        "+34 619 87 65 43"
      ]
    }
  ]
},
```

Note: Description can have more than one line.

Note: If more telephones are needed, just add a comma behind the last one and add another number.

2.5 Footer

Left footer text, is composed by copyright and appTitle, located in the config.json.

```
"version": "1.24.0",
"language": "en",
"appTitle": "AMVARA Dashboard",
"scenario": "dev",
"delay": 0,
"delayRequests": 0,
"copyright": "AMVARA",
```

Mid and right footer text, are edited in .src/app/components/ux/footer/footer.component.html:

```
<div routerLink="help" class="flexed copyright">
  <div class="footer">{{ (config$ | async).copyright }} &middot; {{ (config$ | async).appTitle }}</div>
</div>
<div class="flexed info">
  <div class="desktop">Code is Poetry</div>
  <div class="desktop">amvara-example@amvara.de</div>
  <div class="mobile">Confidential</div>
</div>
<div routerLink="about" class="flexed version">
  <div class="desktop execution_date">Execution Date: 15.02.2021 - 09:00h</div>
  <div>v{{ (config$ | async).version }} {{ (config$ | async).language | uppercase }}</div>
</div>
```

Inside div class flexed, is located the middle text, the text shown will depend if it is accesed from a desktop or a mobile. Then we have the flexed version, that display the execution date and the version of the dashboard.

2.6 Colors

Dasboard's colors, are edited in ./src/app/common/_colors.scss.

```
// Main colors
$bg-color: rgb(0,22,40);
$semi-white: rgba(white, .85);
$semi-black-1: lighten($bg-color, 5);
$semi-black-2: lighten($bg-color, 8);
$header-box: $semi-black-1;
$content-box: $semi-black-2;

// Colors
$blue: #ef3340;
$grey: #d3d3d3;
$orange: rgb(247,174,21);
```

The color variables to edit in order to change dashboard colors are:

- $bg-color: Dashboard's background color.
- $semi-white: Dashboard's text color.
- $blue: Dashboard's colors for title and total numbers.
- $grey: General information telephone color.
- $orange: Will change percentage text color and general information link.

2.7 Chart Legend and colors

In order to change chart legend and colors, the file config.json must be edited. it is located in ./src/assets/config.json.

```
"colorScheme": [
  {
    "color": "#1e88e5",
    "name": "Online"
  },
  {
    "color": "#d32f2f",
    "name": "Outlet"
  },
  {
    "color": "#d3d3d3",
    "name": "Retail"
  },
  {
    "color": "#efb810",
    "name": "Whitebrand"
  }
],
```

Colors are changed in "color" and legend names are changed in "name".

3 Introducing Data

The data insertion or edition is done by editing the file Mobile_List_Chart.csv (Located in .src/assets/reports/), it is mandatory to use the same name in the ticket type column inside the csv and in the colorscheme inside config.json (Located in .src/assets/). If not, when clicking in the bar chart, data will not be displayed correctly.

3.1 dashboardhelper.sh

Every time Mobile_List_Chart.csv is modified, it is important to execute dashboardhelper.sh (Located in .scripts/), this script will read the csv and will complete the other CSV files such as Mobile_Tickets_Priority.csv or Mobile_Tickets_Service.csv, among others. it is very simple to run the script, just open a terminal in ./scripts folder and run ./dashboard-helper.sh -m. For futher information, run ./dashboardhelper -h or ./dashboardhelper –help

3.2 Table Column Names

In order to change column names to your needs, edit “cell_header” in the language files inside src/assets/i18n.

4 Configure XHR

In order to configure XHR, “fullUrl” and “portalFolder” must be edited inside config.json. In **fullUrl** you have to specify the protocol and the domain. In **portalFolder** you have to specify the route to the main page. These will be concatenated and form a complete URL. Example:

- **fullUrl:** https://www.yourdomain.com
- **portalFolder:** /your/route
- **Result:** https://www.yourdomain.com/your/route

If **you are not using IBM Cognos**, change “protectedUrl” to “” :

```
"fullUrl": "https://www.yourdomain.com",
"portalFolder": "/your/route/",
"protectedUrl":"",
```

Note: If fullUrl is empty, it will take his own domain (web root).

Note: portalFolder can be empty if main page is located in **web root**.

Note: It is important to be careful with “/” in fullUrl and portalFolder, for example, if fullUrl is **“https://www.yourdomain.com/”** and portalFolder is **“/your/route”** your final URL will be **“https://www.yourdomain.com/your/route”**. So you can specify it at the end of fullUrl or at the start of portalFolder, but not in both of them.

4.1 Configure IBM Cognos as datasource

IBM Cognos (aka IBM Analytics) is a BI solution that can do it all: clean, connect, aggregate and visualize with enterprise security. What many developers like us appreciate a lot is the powerful **REST API** exposed by swagger. This dashboard perfectly runs with IBM Cognos. It comes with integrated security prompts and can fetch data directly from saved report outputs. This dashboard is not capable of fetching data reports run interactively. If you are interested in doing so and need help how to accomplish this, please contact us.

Configuring IBM Cognos is just as easy as configuring XHR, the only thing to do is specify the “fullUrl”, the “portalFolder” and “protectedUrl” in config.json. “protectedUrl” is used by IBM Cognos to check if the user has access and is used to maintain the connection too. By default is “v1/notifications”, but in case of doubt leave it empty.

```
"fullUrl": "",
"portalFolder": "/analytics/bi/",
"protectedUrl":"v1/notifications",
```

Also, in config.json edit “apiType” value and set it to “cognos”, by default is empty. This will bring up the IBM Cognos license in the licenses section of about menu.

5 Support

Please do not hesitate to contact us on **Discord** in case of questions or by email: tec_dev@amvara.de

We would love to hear from you.