

Documentation: Vanilla HTML/CSS/JS Implementation of Figma Design with Bootstrap 5

1. Project Overview

This document outlines the key details of implementing a Figma design into static HTML. The website is structured into two sections, each implemented in a separate HTML file:

[index.html](#) and [section-2.html](#).

The website utilizes **Bootstrap 5.3.3** for styling and form elements, while **JavaScript** handles interactive functionalities such as filtering, managing popups, handling active states, and managing tabs and buttons.

SCSS Integration

The project includes **SCSS code**, which is particularly useful for:

1. Integration into other projects.
2. Simplified updates to styles.
3. Compatibility with frameworks that support SCSS.

While the page functions without the SCSS files, they provide added flexibility for maintainability and customization.

- **Color Variables:** The SCSS files define variables for page colors, enabling quick updates to the design palette.
- **Bootstrap References:** The SCSS files include commented references to Bootstrap components. These references are optional and can be uncommented to integrate Bootstrap's native styles into the project's CSS.

CSS and Bootstrap Setup

In its current form, Bootstrap is installed via **Node Package Manager (NPM)** and compiled into a standard CSS file located at [css/styles.css](#). This approach ensures that custom styles and Bootstrap are combined in a single, optimized CSS file.

The pages are responsive, however, the design was not supplied with a design for smaller screens. The layout of the smaller screens is determinate by the standard bootstrap layout for these screens.

The tables will have a horizontal scrolling capability on smaller screens. On the second section, the left side filtering section will hide offscreen on smaller screens and by clicking a button with

the text of “Filters weergeven” and id of “#showFiltersMobile” will reveal the hidden section in a section that will slide in from the left side of the screen. The slide in element can be closed with a close button “X”, or by clicking outside the filtering section. The button mentioned above is hidden on the desktop version of the pages. The script responsible for this behavior is a standard bootstrap functionality that uses “offcanvas” from bootstrap. There is no separate script for it.

2. File Structure

/css

- styles.css: Contains custom CSS for the website.
- styles.css.map

/images

- europe-flag-icon.png: Flag icon used in tables.
- flanders-flag-icon.png: Flag icon used in tables.
- header-placeholder-image.png: Placeholder image from Figma.
- rating-elements.svg: SVG sprite for rating icons.
- rating-score.svg: SVG sprite for score icons.
- ytthumbnail.jpg: Placeholder for video thumbnails.

/js

- main.js: Contains JavaScript functions for interactivity.
- bootstrap.bundle.min.js: Minified Bootstrap bundle.
- bootstrap.bundle.min.js.map

/scss

- _breadcrumb.scss: Styles for the breadcrumb section.
- _content.scss: Styles for the header section.
- _header.scss: Styles for the header section.
- _header-nav-tabs.scss: Styles for tab navigation.
- _links-buttons.scss: Styles for clickable elements.
- _table.scss: Styles for tables.
- _table-ratings-icons.scss: Styles for table rating icons.
- styles.scss: Variables and base styles.

index.html: First section from Figma design.

section-2.html: Second section from Figma design.

favicon.ico: Favicon for the website.

package.json: Node.js package file.

package-lock.json: Lockfile for dependency management.

README.md: Developer instructions.

3. Setup and Usage

How to Open the Page

Open `index.html` or `section-2.html` in any browser to view the sections.

Dependencies

- **Bootstrap 5.3.3**

To use with Node.js:

1. Run `npm install` to install dependencies.
 2. Update npm if necessary.
 3. Start the project with `npm start`.
-

4. Features and Functionality

- **Overview of Key Features:**

- Table filtering system, is a static filtering meant to be used just as a showcase for the HTML elements. It will display the selected filter naming in a badge, according to instruction got with the Figma design.
 - Popups: Some filters are displayed in popups, these scripts can be found in the main.js files.
 - Active States for tabs are using JavaScript to apply and remove the active states, these can be found in the main.js file.
 - The table has rating badges, rating scores and icons, which can be customized within the HTML.
 - The links and buttons have hover states, which were represented in the Figma design. A similar hover state was applied to links and buttons that didn't had hover state in the Figma design.
-

5. Code Details

- **HTML:**
 - The page uses semantic HTML elements like `<section>`, `<form>`, `<aside>`, `<article>` and `<nav>` beside the usual HTML tags for structure. This can help to identify elements and help the page SEO when it will be online.
 - The tables use classic `<table>` tags to generate the tables, which is better for SEO and development
 - **CSS:**
 - The names of CSS classes refer to the element that are used, in the pages.
Example: `.table-header-filter-button` refers to the table header, not the content, not the row of the table or other elements.
 - Combination of classes are used to style some element like styled checkboxes which can be found in the side filtering section of the second page and in the custom popup that appears when the first table subjected to filtering.
 - **JavaScript:**
 - In the main.js JavaScript file contains all the functions used in the two pages. They are separated by comment which describe the element for what is used for.
Example: `/* script for the table filtering dropdown BBT starts here */` this is in front of a script and at the end of the script you can find `/* script for the table filtering dropdown BBT ends here */`
 - Also pair the script with the HTML element using the id's in the HTML files and `main.js` file.
-

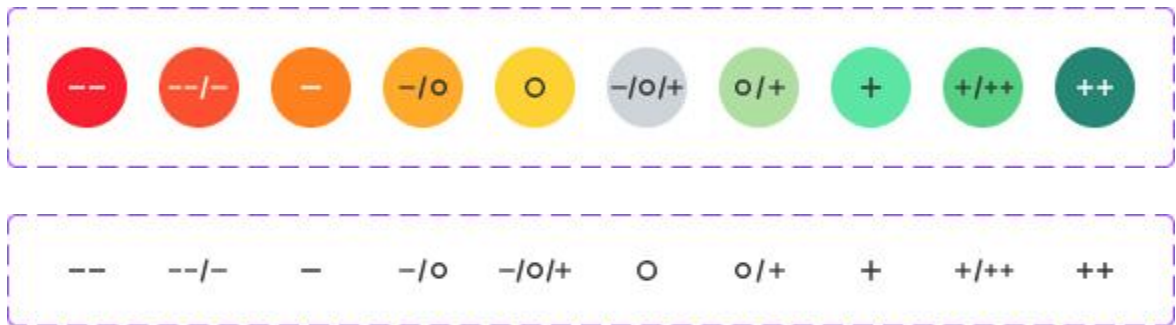
5. Icon Management

The tables are using tree types of icons beside standard text. These are

`<div class="rating-icon ra-6"></div>` - for the circle type icon.

`<div class="rating-score rs-4-alt"></div>` - for icon just with the mathematical symbols

Another variation of the icons if they also contain a number which refers to the footnote of the table found in the first section of the page. `<div class="rating-score rs-6">³</div>` . This number is placed and style in a `<sup>` HTML tag.



The rating score and rating number elements use a sprite image in SVG format to display the needed icon. It is an SVG file to maintain the image quality for different screen resolution. If the quality of the SVG is not satisfying, please edit and export a better quality svg from the Figma design.

The two sprite images can be found in the image folder: rating-elements.svg for rating elements and rating-score.svg for the rating score icons.

Example of usage of the icons:

`<div class="rating-icon ra-6"></div>` - the first class will determine the type of icon "rating score" or "rating icon".

`<div class="rating-icon ra-6"></div>` - the second class will determine the type of rating score or icon.

These are the namings for rating score: .rs-0, .rs-1, .rs-2, .rs-3, .rs-4, .rs-4-alt, .rs-5, .rs-6, .rs-7 and .rs-8 the "rs" in the naming refers to the "Rating Score" and the number represents the number of the icon as it's placed in the sprite image. The sprite image is a direct export of the image from Figma.

These are the namings for rating icons: .ra-0, .ra-1, .ra-2, .ra-3, .ra-4, .ra-4-alt, .ra-5, .ra-6, .ra-7 and .ra-8 the "ra" in the naming refers to the "Rating" and the number represents the number of the icon as it's placed in the sprite image. The sprite image is a direct export of the image from Figma.

The naming numbers were used in the Figma file, this was the best solution to maintain the logic behind the naming. The class is shortened, for page speed and rendering of the html, once there are multiple rows of table, long class names can be an issue.


There are icons that represent flags. `<div class="flag-icon eu">EU</div>` - The class name of the icon is "flag-icon" and the second part of the class name "eu" represents the flag. The flag is a background image used in css `background-image: url("../images/europe-flag-icon.png");` - to extend the number of flag list new css rules has to be added in the css file and new images ha to be added in the image directory.



Arrow icons in links and buttons are either background images or background masks to support the hovering effect.

Overzicht

▼ Beschrijving van PFAS-verwerkende sectoren

 Chemische industrie (productie van PFAS)

➤ Voornaamste PFAS gebruik in de productieprocessen van

6. Forms and Bootstrap Integration

- For the checkboxes and input, the standard bootstrap 5.3.3 checkbox and input is used with customization.
-

7. Testing and Debugging

- The website was tested locally, as static HTML through all the page screens. There are some elements that can be optimized for certain scenarios.
- The design was tested using A/B visual testing to ensure pixel perfect layout.