



LIVE CODING OF A WEB INTERACTIVE APPLICATION

Luc Jaouen

Matelys, 7 rue des Maraichers, Bat. B., 69120, Vaulx-en-Velin, France.

ABSTRACT

During this presentation, I propose to show how to code, from scratch, a simple interactive web page describing an acoustic concept: the calculation of the resulting noise level, in decibel, from multiple uncorrelated sound sources. For this presentation, a basic HTML + responsive CSS template will be used together with some lines of Javascript to add interactivity. The full code for this interactive web application is available at <https://github.com/ljaouen/FA2023/>

Keywords: *education, interactive, web*

1. INTRODUCTION

Traditional teaching methods, such as lectures and textbooks, can be unengaging, making it difficult for people to retain the information presented to them. However, modern technologies, such as web interactive applications, have made it possible to make the learning process more engaging and interactive. One of the benefits of using web applications is that they are easily accessible. Anyone can access these applications from anywhere, at any time, as long as they had, once, an internet connection to store these applications on their device (a computer, phone, or tablet). This means that learners can work at their own pace and on their own schedule.

In this work, we live code a simple web application to illustrate the calculation of the resulting noise level,

**Corresponding author: luc.jaouen@matelys.com.*

Copyright: ©2023 The author This is an open-access article distributed under the terms of the Creative Commons Attribution 3.0 Unported License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original author and source are credited.

in decibel, from multiple uncorrelated sound sources. A screenshot of the web application is presented in figure 1.

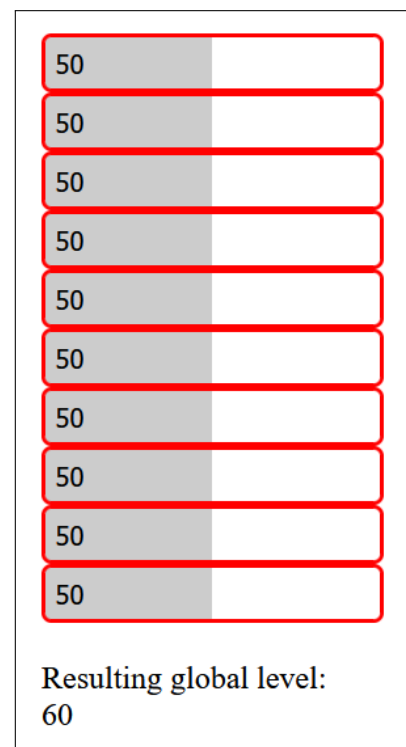


Figure 1. Screenshot of the basic web application that we will create, computing the resulting noise level, in decibel, from multiple uncorrelated sound sources.

Note that the next section of this proceeding presents the HTML+CSS+Javascript template of a web application and not the complete web application. This template can be used to easily create new interactive pages.

2. PREPARING THE CODE

To build the application, we will only need to use a text or programming editor. Two examples are `vim` and `VSCoDe`, with `VSCoDe` being simpler to start with. The application will be created using three different languages or three different steps, like most webpages today.

These languages are:

1. HTML, which is used to define the elements needed for our application,
2. CSS, which is used to style and define how and where the elements of our structure should appear,
3. JavaScript, which is used to add interactions on the elements of the structure and perform calculations.

2.1 Step 1: basic structure in HTML

In this step, we define a basic structure in HTML using the following code (remember that the code can be downloaded at <https://github.com/ljaouen/FA2023/>):

```

1 <!DOCTYPE html>
2 <html lang="en">
3 <head>
4   <meta charset="UTF-8">
5   <meta name="viewport" content="
      width=device-width,
      initial-scale=1.0">
6   <title>Document</title>
7 </head>
8 <body>
9   Our elements will appear here.
10 </body>
11 </html>

```

2.2 Step 2: Add styling with CSS

This step is optional and can be skipped to focus on Step 3 before going back here to set some colors, fonts... using CSS.

An example of CSS, to add in a separate `file.css` or between the tags `<style>` and `</style>` in the HTML file is:

```

1 input {
2   width: 10rem;
3   padding: 5px;
4   border-radius: 5px;
5   border: 2px solid #ff0000;
6   outline: none;

```

```

7   background: linear-gradient(to right
      , #ccc 0%, #ccc 50%, #fff 50%, #fff
      100%);
8 }

```

2.3 Step 3: Adding interactivity with Javascript

While some interactivity can be added using CSS, Javascript offers more possibilities for interactivity in addition to computational capacities. Javascript will be used to execute the computation when a field is changed. Below is an example of code we will use to convert a pressure value from Pascal to deciBel and vice-versa.

```

1 // Convert pressure from Pascal to
  deciBel
2 function convertPressureFromPaToDB (
  pressure) {
3   return 20 * Math.log10(pressure /
  P0);
4 }
5
6 // Convert pressure from deciBel to
  Pascal
7 function convertPressureFromDBToPa (
  pressure) {
8   return Math.pow(10, pressure / 20)
  * P0;
9 }

```

These 2 functions can be stored in a file named `controller.js` or added in between the tags `<script>` and `</script>` in the HTML file

3. ADDITIONAL RESOURCES

This proceeding is just setting up some basics. The coding of the whole application will be done during the presentation at the conference.

Additional examples of what can be done with interactive web applications are available at https://apmr.matelys.com/BasicsAcoustics/1A_EN/ (and in general at <https://apmr.matelys.com>). Some of these applications are discussed in L. Jaouen and O. Robin [1] (see <https://doi.org/10.1121/10.0013011>). In particular, a presentation of the standing wave ratio for the stationary waves inside an impedance tube and its relation to the sound absorption coefficient can be found at https://apmr.matelys.com/2020_Lyon/ (video)

recording available at <https://www.youtube.com/watch?v=T6FgXPTpNQA&t=111s>. An interactive explanation of the coincidence and critical frequencies is presented at <https://apmr.matelys.com/BasicsAcoustics/CoincidenceFrequency/>.

Finally, below is a far-from-being-exhaustive list of other content creators in the fields of Animation & Acoustics:

- Matthew Wright, @MCMW_Acoustics on Twitter,
- Steve Hambrick, <https://www.hambricacoustics.com>
- Dan Russel, @drussellpsu on Twitter, <https://www.acs.psu.edu/drussell/demos.html>
- Paul Fasltad, <https://falstad.com/mathphysics.html>

4. REFERENCES

- [1] L. Jaouen and O. Robin, “Explaining and teaching acoustics through comics, interactive web pages, and video games,” *J Acoust Soc Am.*, vol. 152, no. 2, pp. 745–752, 2022.