**Week 12 documentation**

Following a consultation with Wi En, I identified a key shortcoming of my website: while it presented charts and data in a clear and concise manner, it lacked the necessary context to truly engage readers. To address this issue, I decided to enhance the narrative of my website by incorporating a range of visual and interactive elements, including images, written content, and interactive features. By doing so, I aim to create a more immersive and engaging experience for my audience, helping them to better understand the data and the significance of waste crisis in Singapore.

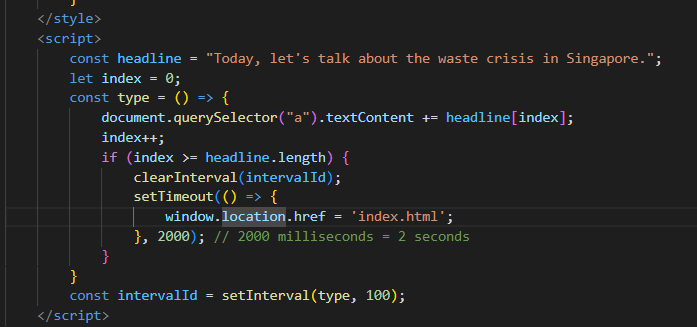
**Landing Page**

To create a dynamic and engaging landing page, I decided to incorporate an auto-typing effect that would capture the attention of visitors right from the start.

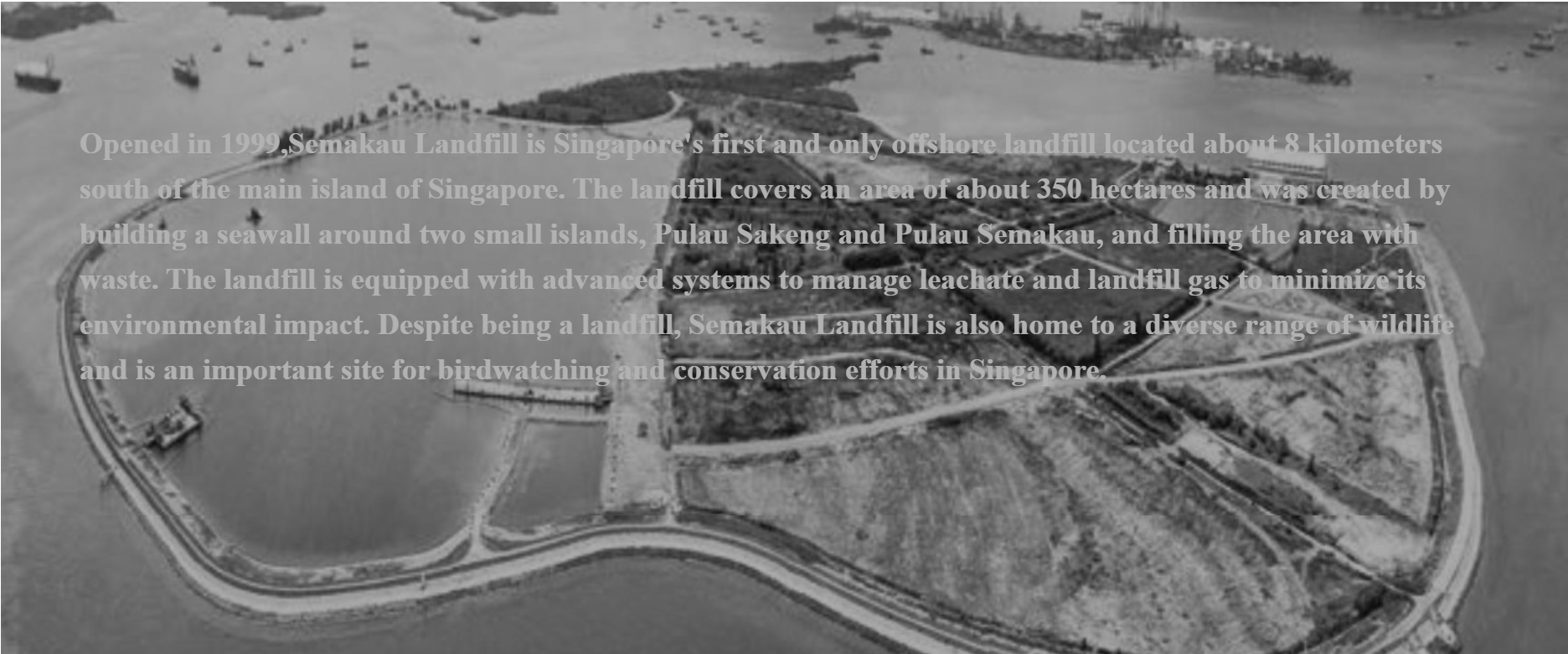
Graphical user interface, text, application, email

Description automatically generated

This is achieved through a JavaScript function that uses DOM manipulation to select the a element and modify its text content by appending each character of the headline string in sequence. Additionally, the function includes conditional logic that stops the auto-typing effect once the end of the headline string is reached. After a brief delay, the function then redirects the user to the index.html page.



**Section1**



To give readers context about the waste crisis in Singapore, I decided to introduce Semakau Landfill as a starting point. To make this introduction visually engaging, I selected a photo related to the landfill and displayed it as the background of the page. I then adjusted the dimensions of the photo to occupy the entire page with a width of 1440px and a height of 700px. To improve readability and create a contrast between the text and the image, I applied a grayscale filter of 100% and a brightness of 70% to the image. Finally, I added a paragraph of introductory text that floated on top of the image.

Text

Description automatically generated

Currently, there is an issue unsolved. The gray filters applied to the image also affect the text, reducing the brightness contrast between them and making the text harder to read. This problem needs to be addressed before the final submission.­

**Section2**

**Graphical user interface

Description automatically generated**

The initial section of the website presents statistics on the amount of waste generated, recycled, and disposed of in Singapore from 2017 to 2020. To visually demonstrate the consequences of waste accumulation, I wish to add a sequence of photographs portrays the gradual filling up of Semakau Landfill. Alongside these images, there is an attention-grabbing headline informs readers that "Semakau Landfill is projected to reach its maximum capacity by 2035."

Given the many visual elements involved, it is important to establish a layout that ensures both coherence and visual appeal. While it may be tempting to simply use an existing layout and tweak the code to fit the website's needs, I believe that having a graphic design background enables me to pursue a more customized approach. I use Figma to create an aesthetically pleasing design that effectively arranges these elements. Then, I export the CSS code and integrate it into my website. This workflow allows me to have greater control over the design of the website and enhane my understanding of media technology.

A screenshot of a computer

Description automatically generated with medium confidence

This approach has proven to be effective, hence I will continue to do this for my section 3,4 and 5.

**Section 4**

I create a bubble chart in section 4 to visually present the waste management situation of different waste streams in Singapore. The chart utilizes the x-axis to represent the amount of waste generated, the y-axis to represent the amount of waste disposed, and the area of the bubble to represent the disposal rate.

Chart, bubble chart

Description automatically generated

In this code, three arrays of data - WasteStream, WasteGenerated, and WasteDisposed - are used to create input data for the chart using the map function. The map function loops through each element in the WasteStream array and returns a new object that combines data from all three arrays into one object. The new object has x, y, and r values, which are derived from the corresponding indices in WasteGenerated, WasteDisposed, and WasteDisposalRate multiplied by 0.3(for better visual experience). The label property is set to the corresponding value in WasteStream. This mapping of data enables easy extraction and combination of different data sets to create a bubble chart.

Text

Description automatically generated

I gained a wealth of inspiration from a variety of websites, including the following examples:

<https://www.pornweek.nz/>

<https://brightmark.zajno.com/#step-1-plasticsRenewal>

Some learning resources I refer to this week:

1. How to make an auto typing text effect: <https://www.youtube.com/watch?v=42M9esAvAEU>
2. Figma Courses:<https://www.youtube.com/watch?v=jwCmIBJ8Jtc&t=36000s>
3. Figma to Code: <https://www.youtube.com/watch?v=QFz_2s__Vmw>