

The Great Saibling Death Match

Process Book

Damian & Jakob
Milestone 3

Idea

Project Idea

We wanted to highlight the domestic fishing industry in an engaging, interactive, and gamified way. Our motivation came from two main reasons:

- Raising awareness of the Swiss ecosystem and our relationship with lakes
- A personal challenge: making something interesting out of a seemingly mundane topic like fishing

The Dataset

We explored two main datasets showing the amount of fish caught (in kg) in the most popular lakes in Switzerland. They provide yearly data on catch amounts per fish species and per lake. One dataset covers professional fishing, the other shows hobby fishing. Each lake's surface area (in km²) is also included.

The datasets, sourced from OpenSwissData, were already quite clean and usable out of the box. A few rare species had missing values. Some cleaning was still needed, like aligning the time scale, since not all species had data for the same years. We merged everything into one JSON file from the original multi-sheet Excel file.

The Data Story

The first step was defining the story: What questions do we want to raise? What should users take away from all these numbers?

We settled on these **core questions**:

- Where does the fish we eat come from?
- Which fish are commercially fished in our lakes?
- Which species are fished the most?
- How has fishing changed over the past 30 years?

Communication Concept

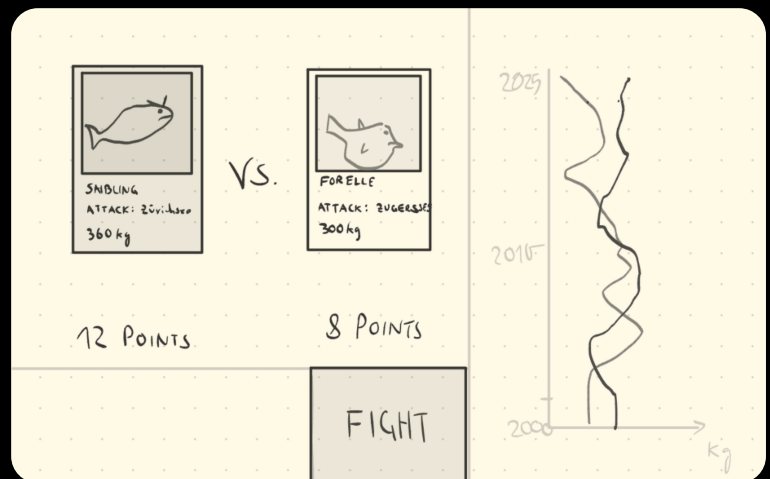
The next challenge was how to communicate these questions without being boring. Our initial idea, a battle of lakes based on output, was scrapped. We shifted focus to the fish themselves.

We gamified the interaction, turning the project into a Pokémon-inspired card battle game. Hence the title: The Great Saibling Deathmatch. Early hand-drawn sketches show this evolution.

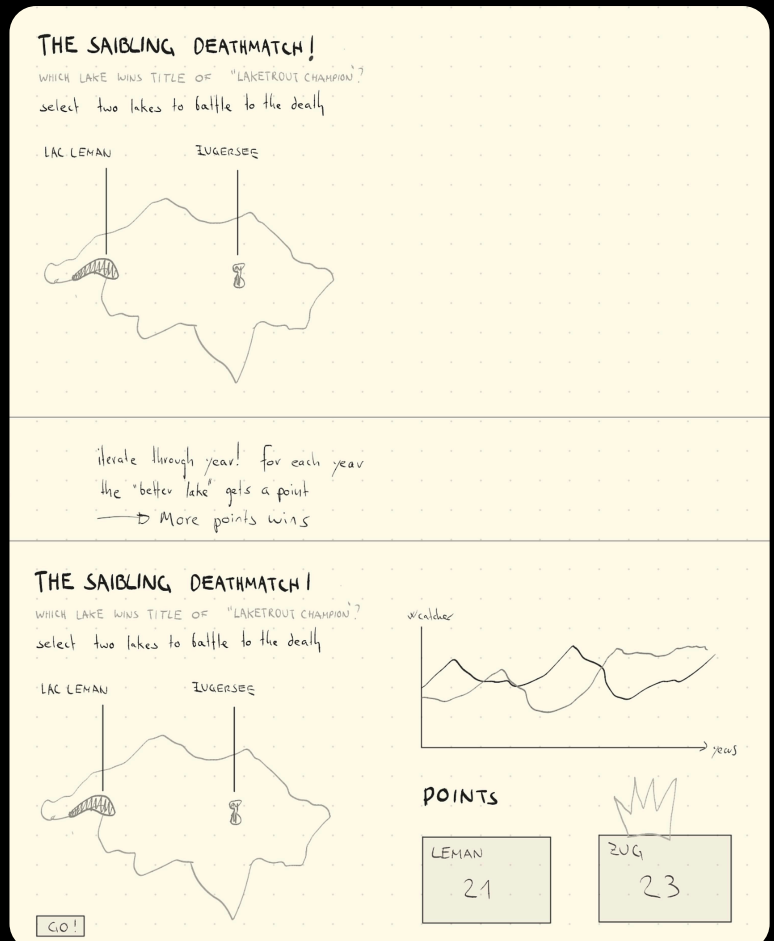
Before coding, we designed a wireframe and aesthetic reference in Figma. This helped us plan and stay focused. We mapped out the game flow, aiming to balance gameplay with learning. The solution: fish species became champions, lakes became attack options. Battles used real data, comparing fishing trends over 30 years. This made data visualizations an integral part of the game, not just an educational add-on.

Brainstorm

Early concept illustrations during brainstorming session, which was chosen.



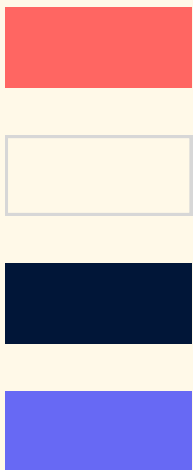
Early concept illustrations during brainstorming session, which was not chosen.



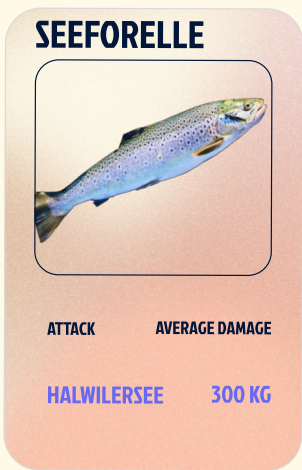
Design

We chose a clean block layout based on **golden ratio** proportions to ensure good visual flow.

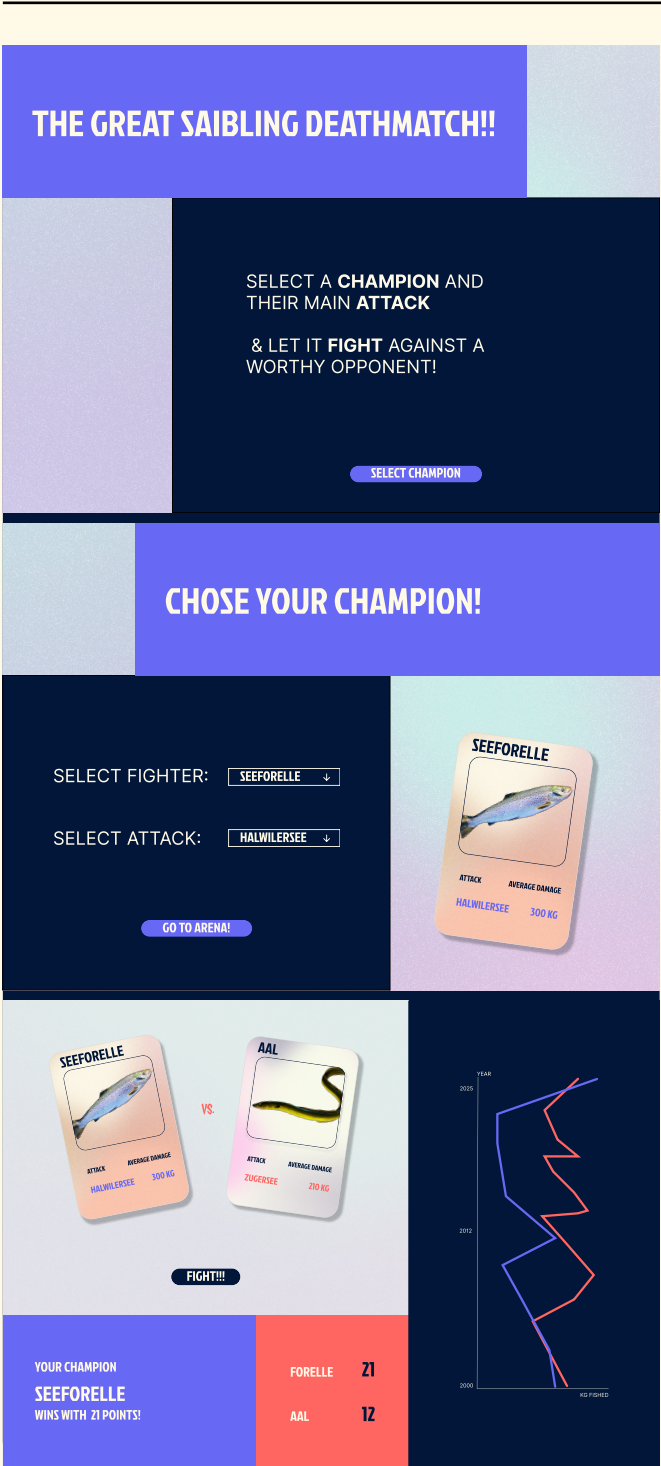
The **color palette** is friendly and **playful**, reminiscent of games like brawl stars and kahoot. It is primarily blue, with red accents to guide the user's attention. **Minimal text** further helps maintain focus and clarity.



Playful color palette



Champion card design



Final website wireframe and design plan made in Figma.

Execution

Next, we implemented the battle system. Year-by-year data was called and compared, points calculated, and plots generated using D3.js (from Weeks 4 - 5).

At this stage, the JavaScript became messy. Since we were both coding, redundancy and poor documentation crept in. We took time to clean up and reorganize the code.

Design-wise, we used assets we created in Figma, royalty-free fish images, and the reference art we made. This helped us quickly build a visually consistent and engaging site with fun champion cards. A mistake we made was prioritizing embellishments, like hover effects and glows, too early. This complicated debugging.

In the end, we managed to merge logic and design into a working site. One late design debate was about the graph orientation. A vertical timeline (with time on the Y-axis) suited the game style, but felt unintuitive for data visualization. We switched to the standard horizontal timeline (time on X-axis). A final issue: the site didn't explain itself well. Since we knew how it worked, we underestimated how confusing it might be for others. We added a better introduction at the end to guide users through the experience.

Game Flow

- The user selects a champion (fish species) and an attack (lake).
- This generates a custom playing card showing the average kg caught per year.
- A random opponent is generated.
- The battle begins: a plot is built year by year, and the fish with the higher catch gets a point for that year.
- The fish with the most points wins.

Implementation

Functionality came first. We converted the data into JSON structures and created the champion selection. Using the DOM API (as practiced in Weeks 3 and 5), we dynamically updated fish names, lake names, and images.

Work- Split

Damian

data search & cleaning
brainstorm & idea development
battle mechanics
code maintenance
general webdev

Jakob

data search & cleaning
brainstorm & idea development
game- and graphic design
card mechanics
general webdev