



# How have songs changed in the past 22 years?

Welcome 🖐️

We want to welcome you to our process book. Just in case, you can find our website [here](#) and our screencast [here](#). In some words, how the book is organised. First, we will time travel through our work, and then we will go into details of each part of our website. In the end, we will give an overview of our peer assessment.

## A time travel through our work

First, we had to find a **topic** that fit us all. Since we three have different backgrounds, we found a connecting piece of music. In the beginning, we wanted to do something related to music. In general, we tried to find out if we could link changes in music trends to social changes, but we had to discard this idea and only focused on how music has evolved over the past 22 years. To get started, we had to find out what data was available, and during this, we also had to **learn** how music is broken down data and how its success was assessed. This digging with data helped us get the first idea of how we wanted to present and organise our project: first the chart data in general, then the data of the artists, the genres, lyrics and in the end, the songs of the properties.

After collecting the data, we had to clean it and get the first “feeling” for it with an exploratory data analysis. Once we understood the data better, we started creating the first **designs and sketches**. We made the first ideas for the five different parts. Also, we came up with a first skeleton for the overall website, which we discarded later in favour of a different design to merge all the components.

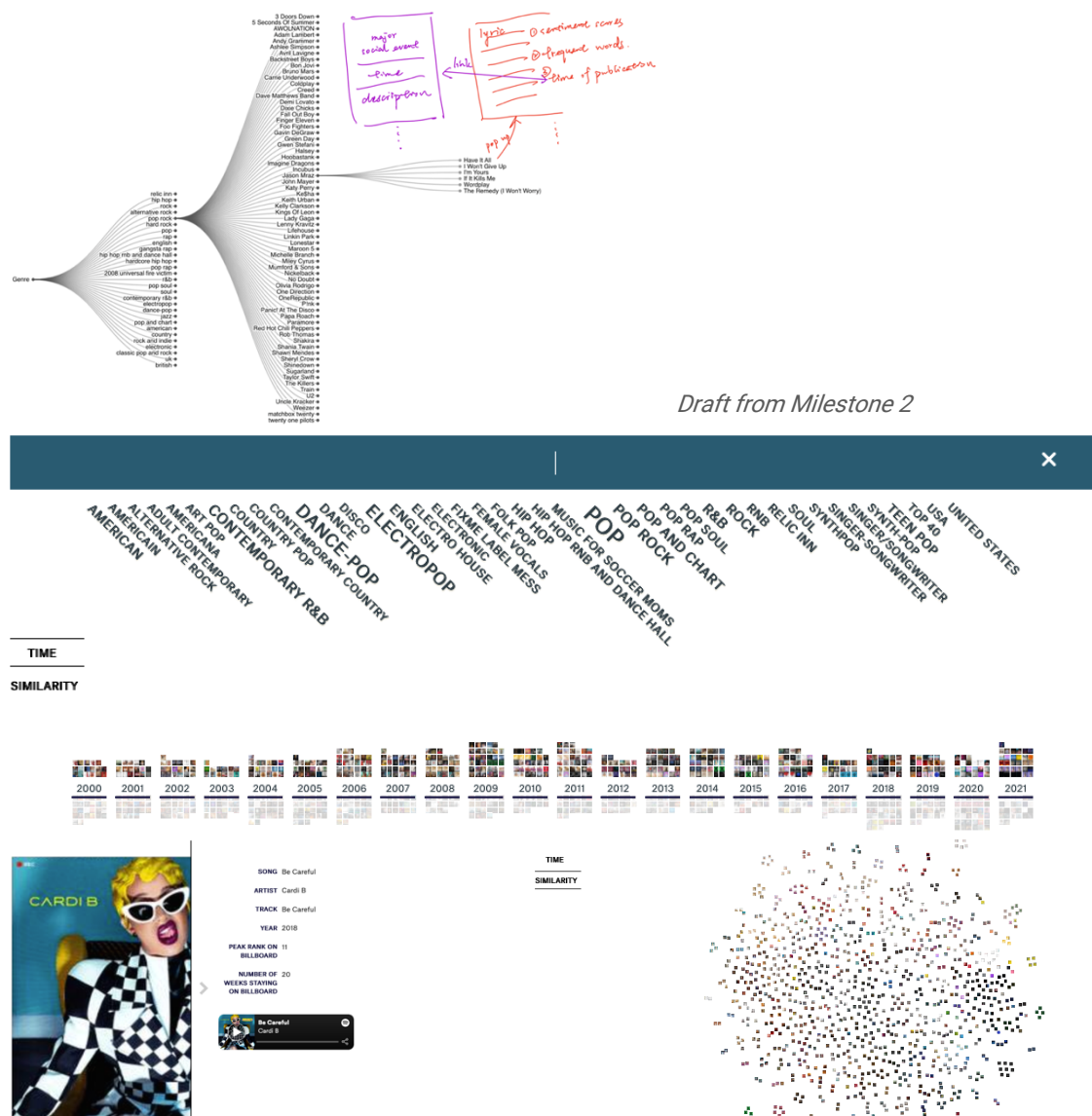
Once we had the sketches, we started with the **coding** part of our project, where we applied our knowledge learned in the course about HTML, CSS and d3.js. We decided to not stick only to d3.js since our group was partially familiar with different frameworks. In the beginning, every visualisation was an isolated prototype, which allowed us to split the work and not run into conflicts with each other. After we finished our prototypes, we merged everything into one website.

In milestone two, we got the feedback that we should ensure that the visualisations fit well together. So changed the prototypes to create a similar look and feel throughout the page and visualisations. We wanted to create a little data story to connect everything with less text. Since we wanted to “show and not tell”. We shared our final draft with our peers, friends, and family and implemented their feedback to collect feedback. Finally, we wanted to create an interactive and enjoyable use of the website. For this, we included features like a music player and lyrics generator. Even if it is not a visualisation, it supports the journey through our data story and makes it enjoyable.

# Part 1: Billboard Songs

## Chartview

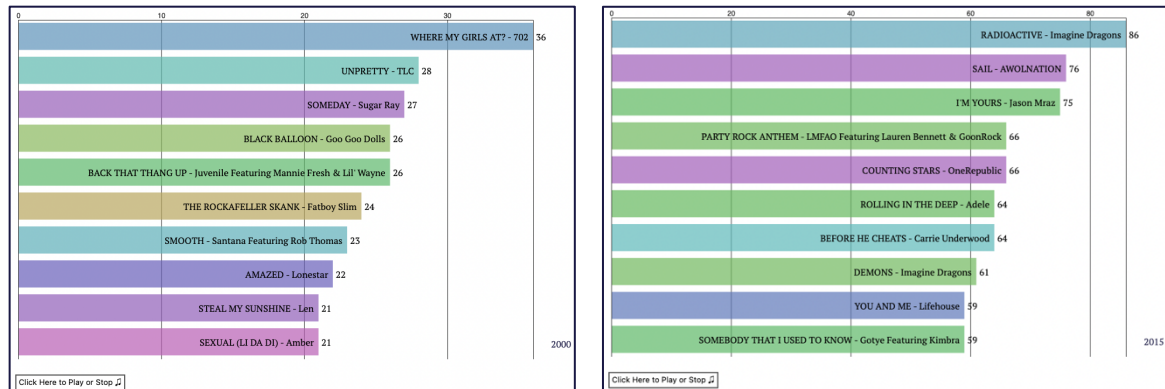
With our Billboard, Spotify and genre datasets, we would like to create a tasteful website that accompanies you with melodious music. Since there are still too many choices, we spent a week figuring out the type of the main visualisation we would like to have and managed to create this first interactive “Big Picture” of music. The fundamental skeleton of this visualisation is from the Vikus Viewer, which we had to modify heavily to create the required visualization. Each entry is zoomable and then the details of the cover will be shown, we additionally added the Spotify music clip for each song such that you could easily try out a new song. The similarity tab on the left will give a nice clustering result that groups the similar covers together.



Used Framework: [Vikus Viewer d3.js](#)

## Timeless Billboard Music across 22 years

Some songs are popular once and then forgotten; others become timeless classics and will be well-remembered by history. Hence, we would like to create an animation to play the most long-lasting music with the highest record staying on Billboard over time. Our animation always shows the 10 songs staying on Billboard for the longest time. With the ranking changing slowly over the year, the top 1 music will be automatically played.

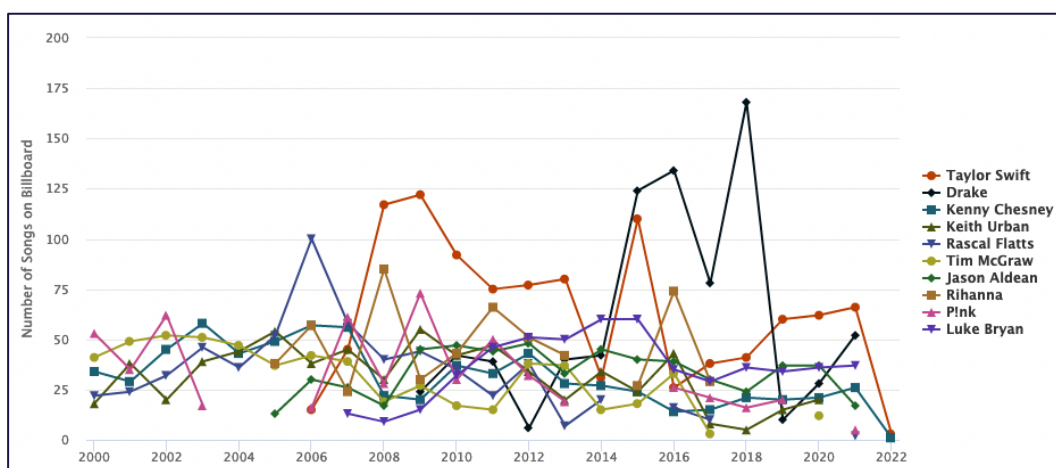


Used Framework: [d3.js](#)

## Part 2: Artists

### Artists with the most number of songs on Billboard

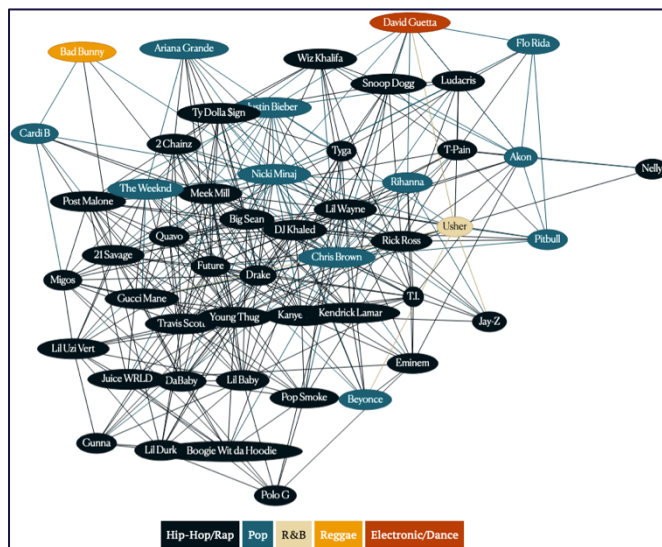
This visualisation gives a clear view of the most famous artists over time. Taylor Swift and Drake have the greatest number of songs on Billboard. However, we can see how their popularity differs from the others.



Used Framework: [Highcharts.js](#)

## Top 50 Most Collaborative Artists Network

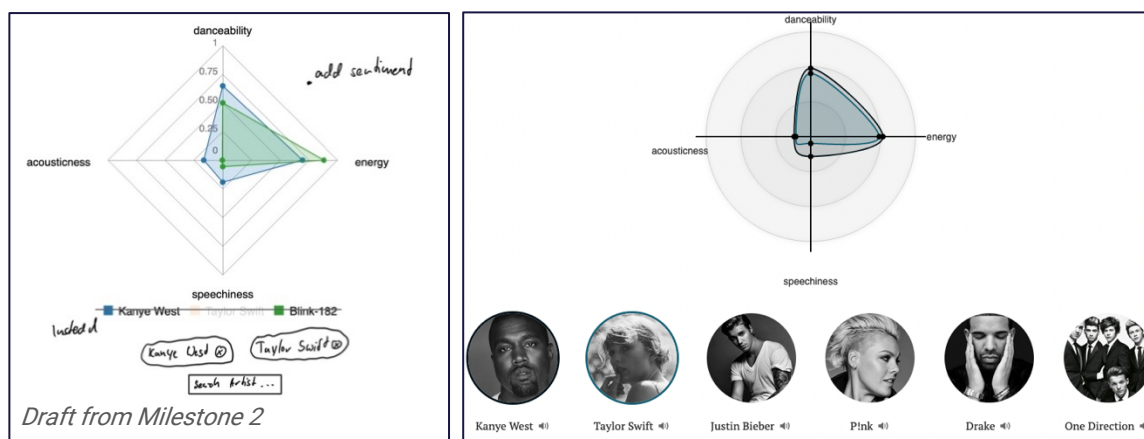
Music artists collaborate; nothing surprising there. But how can we see in a simple and accessible way how an artist is connected with some others, meaning they have worked together once to produce a track? From this need, we decided to use a network graph found on vis.js to show the most collaborative artists and their connections. However, the prototype produced in milestone 2 was flat and dull. So we added colours on the artist's nodes based on their primary genre to visualise at the same time which are the most collaborative genres and how they connect on a singular basis.



Used Framework: [vis.js](https://visjs.org/)

## Artist vs Artist

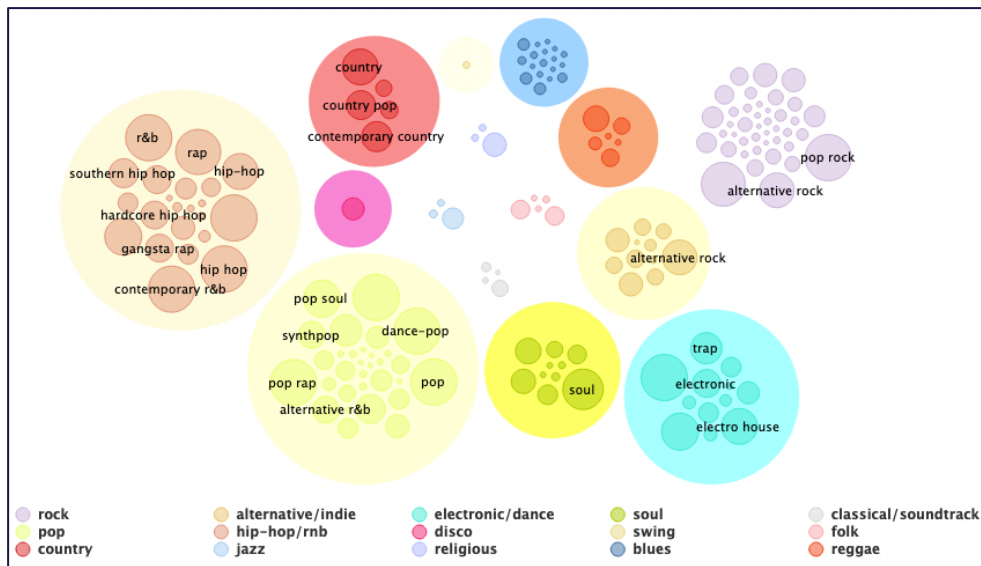
We promised in our second milestone that we would allow comparing different artists against each other; for this, we chose the relevant artists of the last years, and they are comparable in the radar chart. We calculated the mean over all songs in the dataset to get the scores. In the first visual, you can see we choose a different style. During the final steps, we discarded to have the second one, which fits better into the website and also allows us to use the audio player more effectively.



Used Framework: [d3.js](https://d3js.org/)

## Part 3: Genres

Music sub-genres are many and varied. Spotify lists more than 1386 (!). For example, in Hip-Hop, we can find *east coast hip hop*, *trap* or *brooklyn drill*. Based on the idea that it would be exciting to visualise and compare their popularity over 22 years. We aggregated all the sub-genres found on the MusicBrainz dataset and merged them with our existing Billboard dataset. Then, we had to assign a score to each subgenre to represent their popularity fairly and correctly.

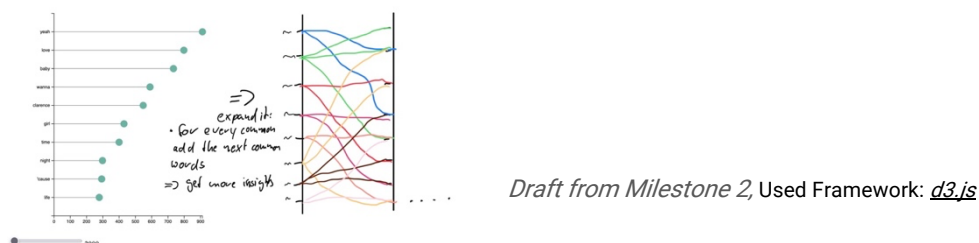


Used Framework: [Highcharts.js](#)

## Part 4: Lyrics

### Let's count

We included this table to show how the lyrics changed over the years. When clicking on a year, the user can select the year he is interested in. We used this format instead of the one in milestone 2 to have a pleasant design that fits the page better. We also realised that the proposed change doesn't give more insights and is more irritating.



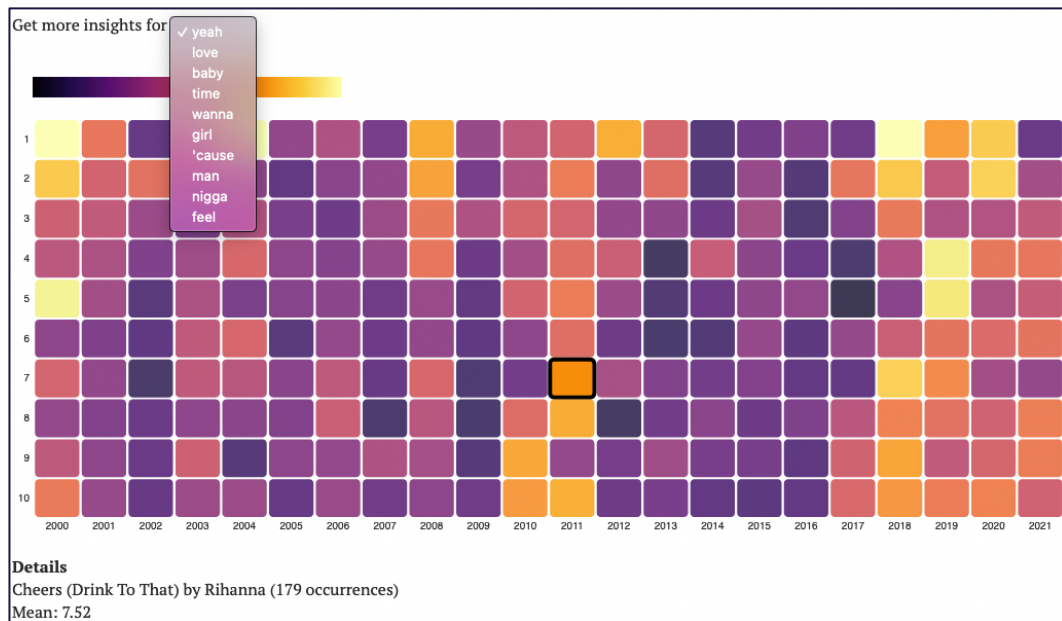
Draft from Milestone 2, Used Framework: [d3.js](#)

WORD	RANKING IN YEAR 20 00 - 21 ∨
1. yeah	31329
2. love	21692
3. baby	16298
4. time	11755
5. wanna	11734
6. girl	10887
7. 'cause	9364
8. man	8212
9. nigga	8065
10. feel	7459



## ... More Details

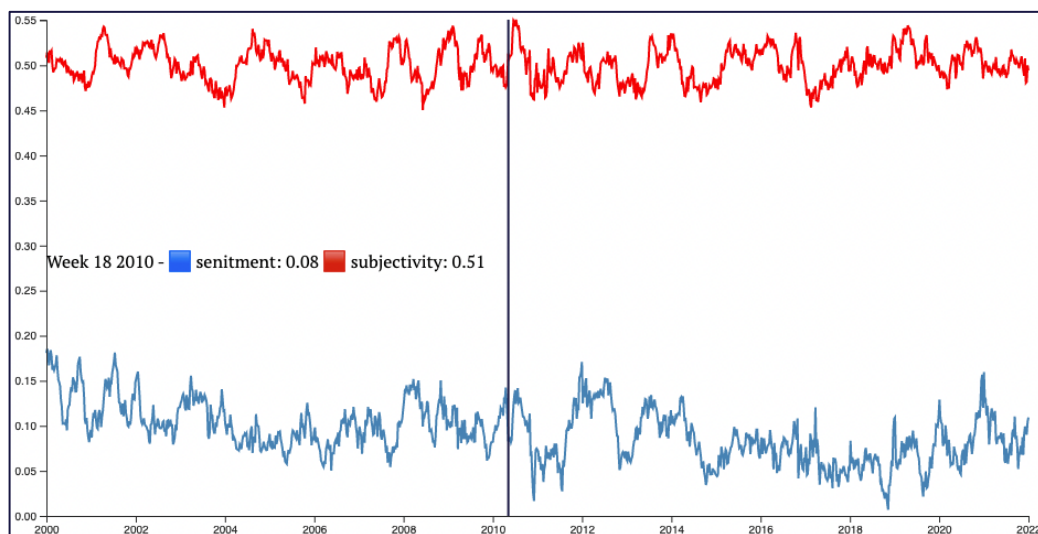
We used the following heatmap to get more insights into the distribution of the words through the years. Here we stuck entirely to the design proposed in milestone 2. We fixed the legend under the heatmap instead of having a moving one.



Used Framework: [d3.js](#)

## Sentiment and Subjectivity

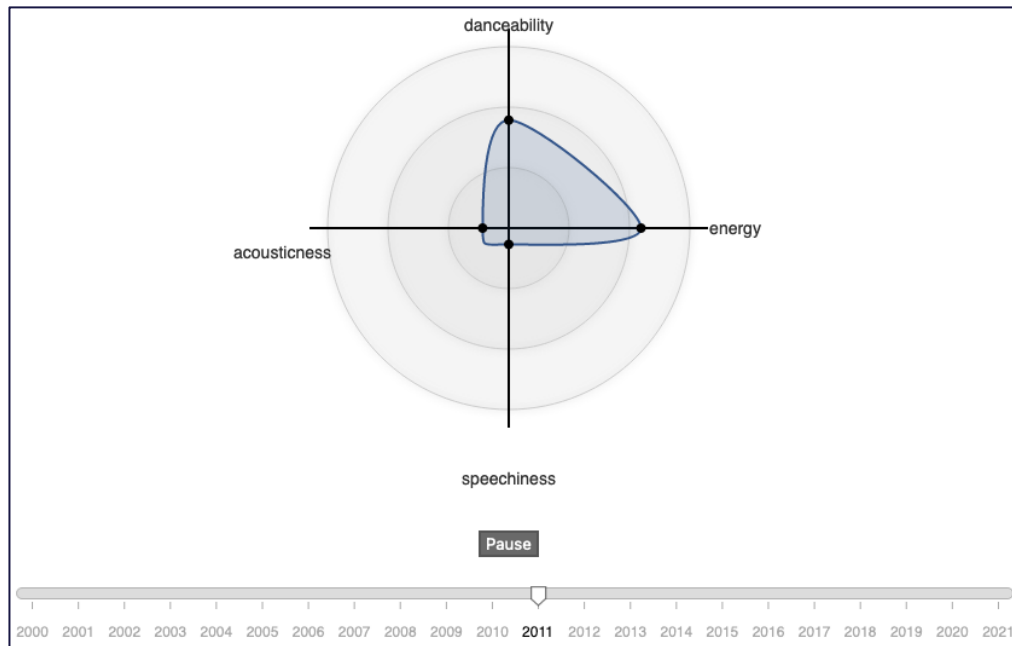
To show how the sentiment and subjectivity changed over time, we choose the line chart since it gives an intuitive way to see the changes over time. To understand the chart better, we included a tooltip, which shows the exact value of the chosen position. The tooltip and the cleaner design are the main changes over the sketch in milestone 2.



Used Framework: [d3.js](#)

## Part 5: Song Change Over Time

To track the change over the time of the songs, we wanted to create an animation that shows the difference and where the user can get more details for each year. We used the sketch of artist vs artist of milestone 2 and appended it with the dynamic component and slider.



Used Framework: [d3.js](https://d3js.org/)

## Challenges

Modifying the original Vikus Viewer to suit our needs was the biggest challenge. We finished the most difficult parts, including cover scraping, datasets preparation and error-free website building with collaboration.

Besides that, another challenge was missing entries in the original datasets. For instance, some songs do not have a valid Spotify API, which results in a missing cover image and no playable music. By selecting the most famous music and discarding the most-recent data in 2022 due to the sparsity of information, our Vikus Viewer now looks much more complete. In addition, the genre data was not initially included in our dataset; by merging an outside genre dataset, the genre data got messy, so we took care of the genre variable by manually cleaning it up.

As for the racing chart of weeks-on-board songs, we looped through the Billboard dataset and kept a sorted list for each month. The difficulty was to find the top 1 music where the clips for several of them are not available from Spotify API. Since there are not many songs, we manually obtained the corresponding music clips and extracted their high notes

## Peer assessments

We did our best to handle our project together as a team. Each of us brought new ideas that helped us take steps in the right direction. We frequently meet to debate our project. In the following, find out a detailed breakdown of our work.

### Aamir

created the skeleton for the website and modified the Vikus Viewer, which was usable for us. He made the Lyrics Generator, all visualisations regarding the Lyrics (Let's Count, 🔍... More Details, Sentiment and Subjectivity), the Artist vs Artist, and the song change over time visualisation and generated the required datasets. As well he implemented the audio features for the website.

### Fatih

scraped the track covers using Spotify API to feed the Vikus Viewer. He also recovered missing Spotify data. We owe him the collaboration network graph, which describes the connections between artists along with their genres, and the genres bubble graph, which shows the popularity of each subgenre/genre. For collaboration network graph, a voting algorithm has been designed to assign each artist to a specific genre instead of a list of subgenres. Details can be found in preprocessing/collabs-network.

### Yiren

processed the original datasets to create a comprehensive music database, including Billboard rankings, Spotify hit predictors, and genre data. She created and configured the datasets for Vikus Viewer, which included the core, a new timeline, and cover similarity data from Vikus Viewer Script. The Timeless Billboard racing chart animation allowed viewers to listen to the No.1 song at any time was designed by her. She also made the artist line chart of the top 10 artists, showing how their popularity has changed over time, and a collapsible genre tree featured in milestone 2 but not on our final website.