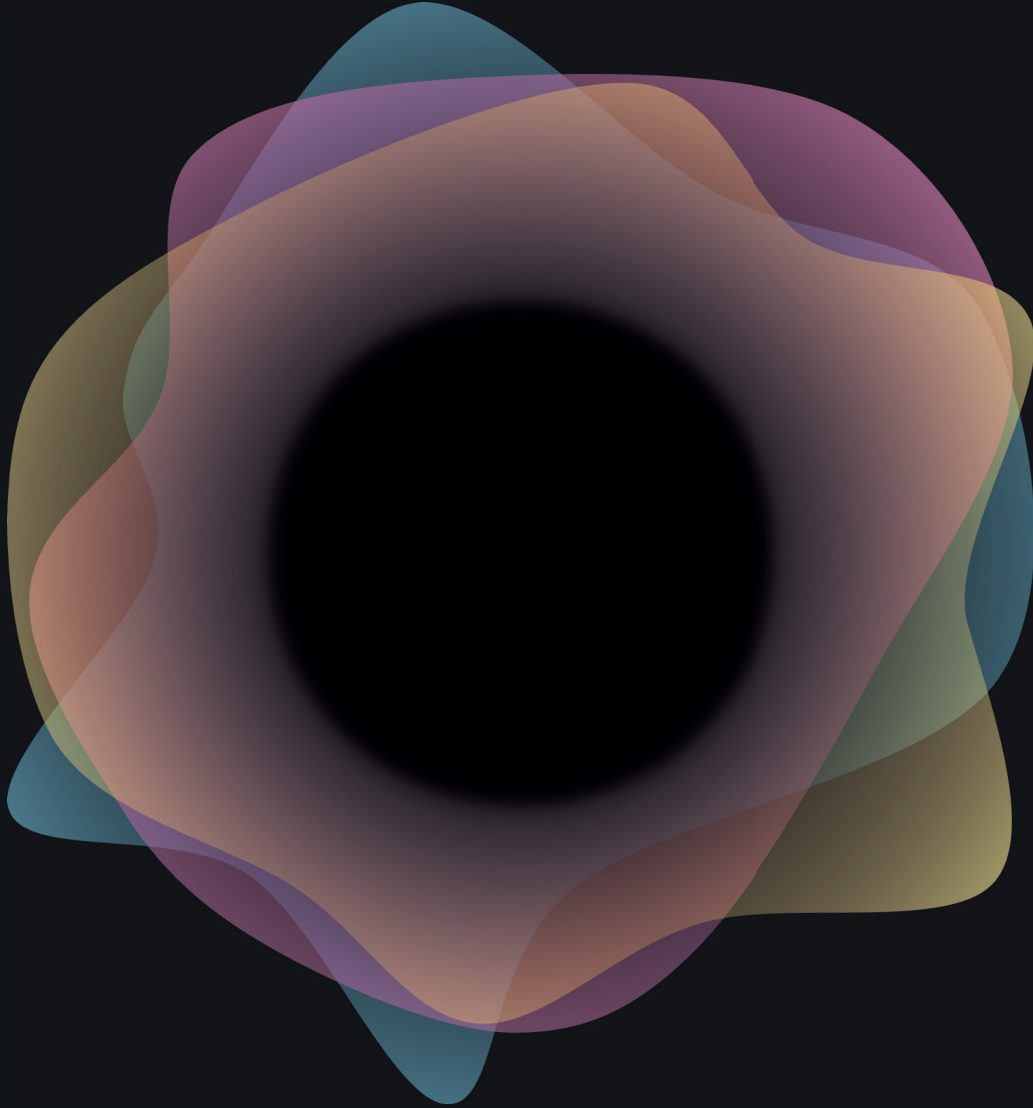


Spütify



Konzept-Dokumentation

Studio Web & Mobile Design & Engineering HS24 Dept. I
I.BA_220_STUW1.H2401

Herbstsemester 2024

Marin, Lukas, Lisa, Laura

Fragestellung

Wie kann die Visualisierung des Musikgeschmacks in Form einer interaktiven Zeitreise das persönliche Musikerlebnis vertiefen?

Idee

Im Fokus steht die „Music Journey“ als Timeline, die durch die letzten Jahre des Musikgeschmacks führt. Die Reise beginnt in der Gegenwart und ermöglicht es den Nutzern, weiter in die Vergangenheit zu scrollen, um frühere Phasen ihres Musikgeschmacks zu erkunden. Diese Musikreise ist grafisch als Kreis dargestellt, der sich je nach Abschnitt der Musikgeschichte des Nutzers verändert. Es werden die letzten 4 Wochen, die letzten 6 Monate, das letzte Jahr sowie jedes Jahr von 2023 bis 2016 (basierend auf den Wrapped-Playlists) abgebildet.

Innerhalb des Kreises wird der Top-Song des jeweiligen Abschnitts abgespielt, wobei der Kreis gleichzeitig die vorherrschenden Genres, Stimmungen und das Tempo des Songs visualisiert. Nutzer können genauere Details und Auflistungen der Musikdaten abrufen.

Am Ende der Reise gibt es einen Share-Button, mit dem Nutzer ein Bild ihrer gesamten Musikreise oder eines spezifischen Abschnitts erstellen können, um ihre musikalische Geschichte in sozialen Medien zu teilen. Unsere Zielgruppe sind Premium-Spotify-Nutzer, insbesondere jüngere Generationen, die gerne tiefer in ihre Musikhistorie eintauchen, sowie langjährige Spotify-Nutzerinnen und -Nutzer, die ein Interesse an Datenanalysen und Musikvisualisierungen haben.

Visuelle Referenzen

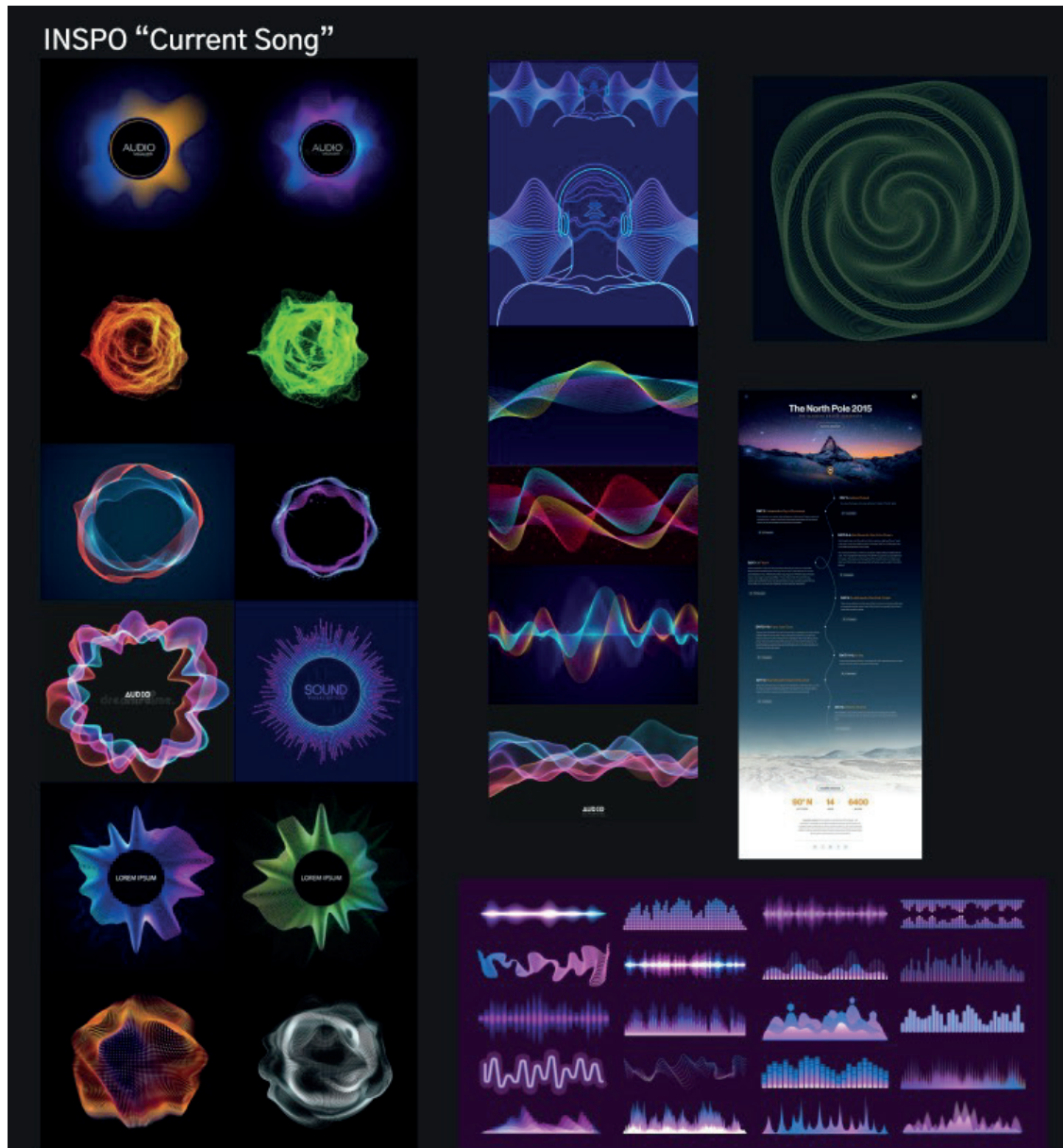


Abbildung 1: Moodboard

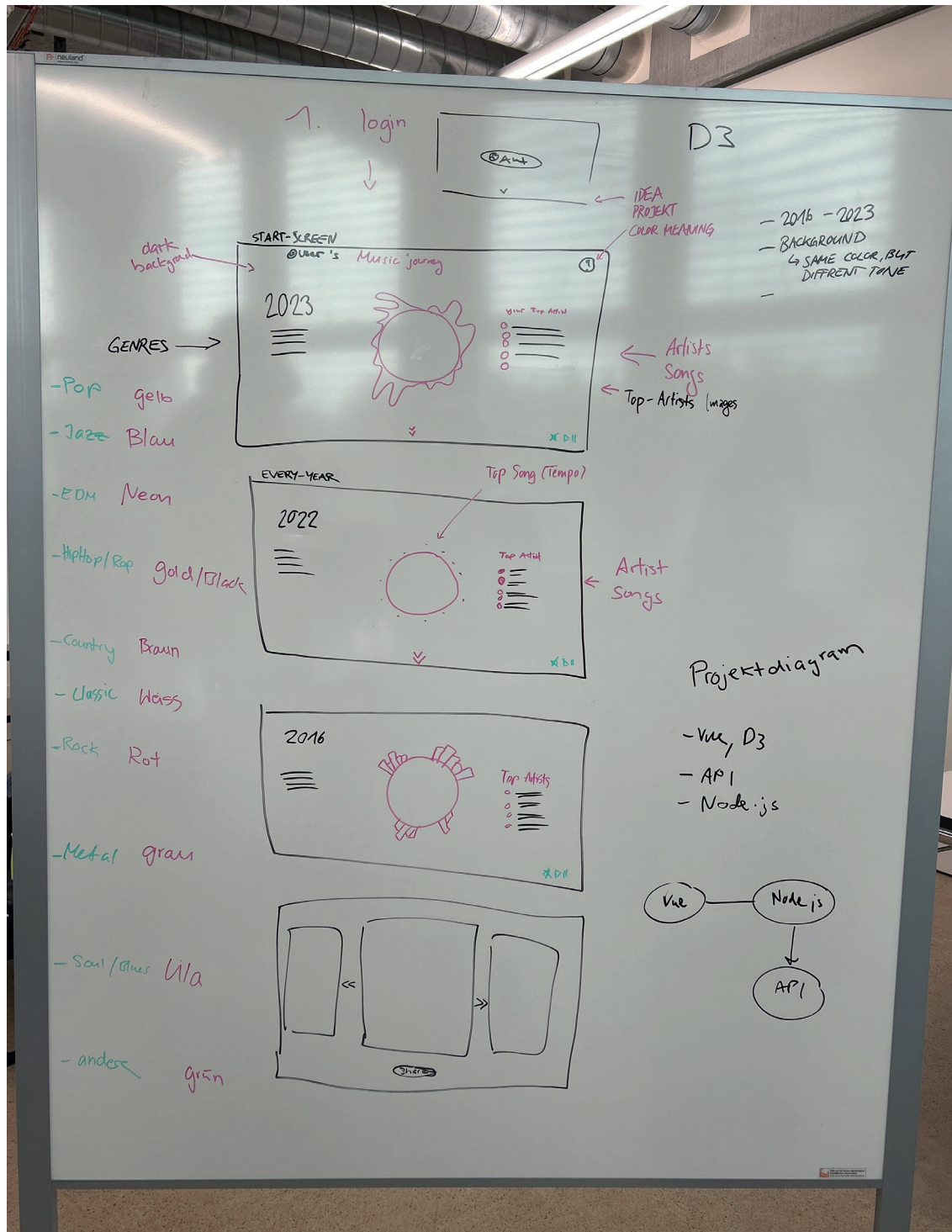


Abbildung 2: Skizzen

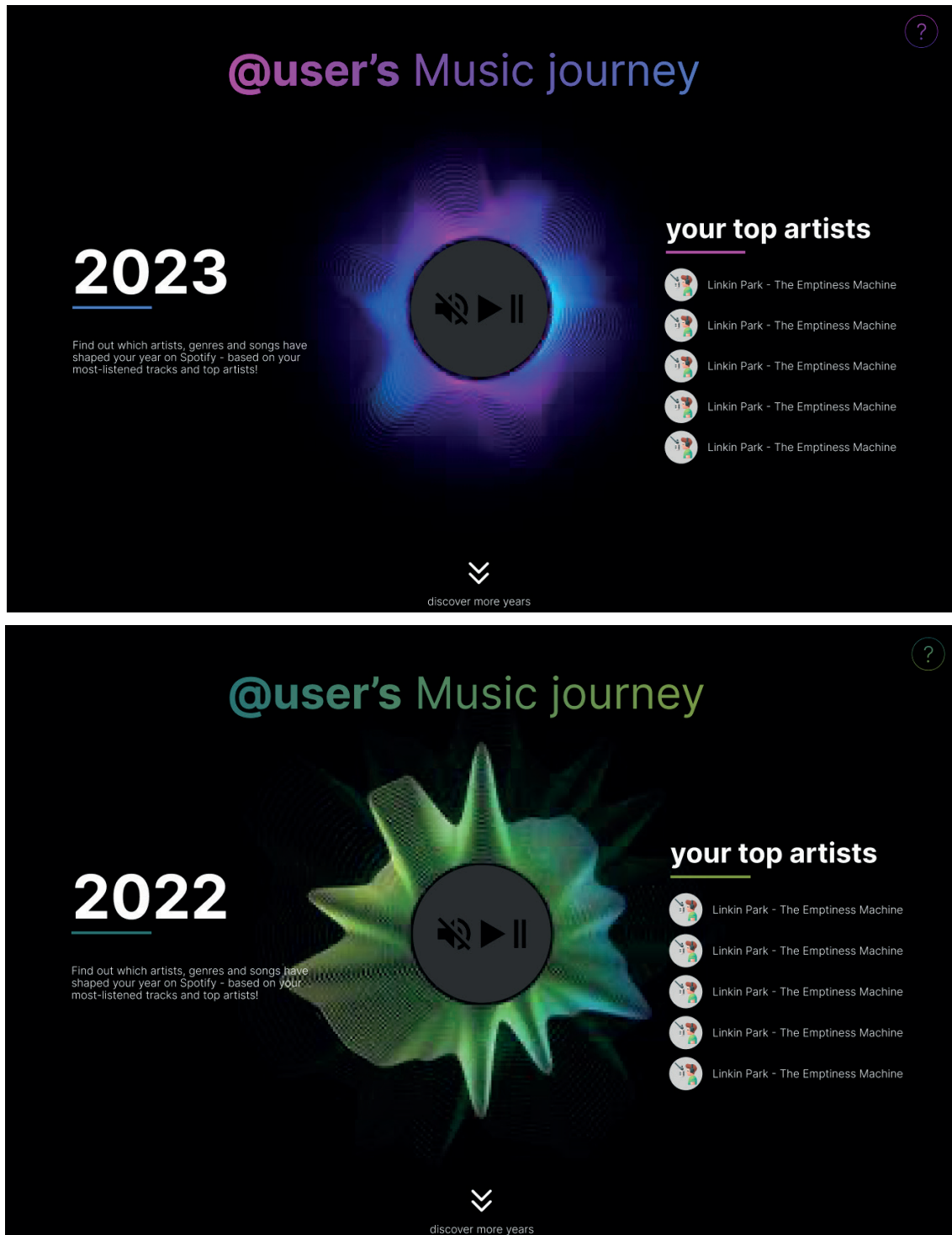


Abbildung 3: Mock-up Variante 1

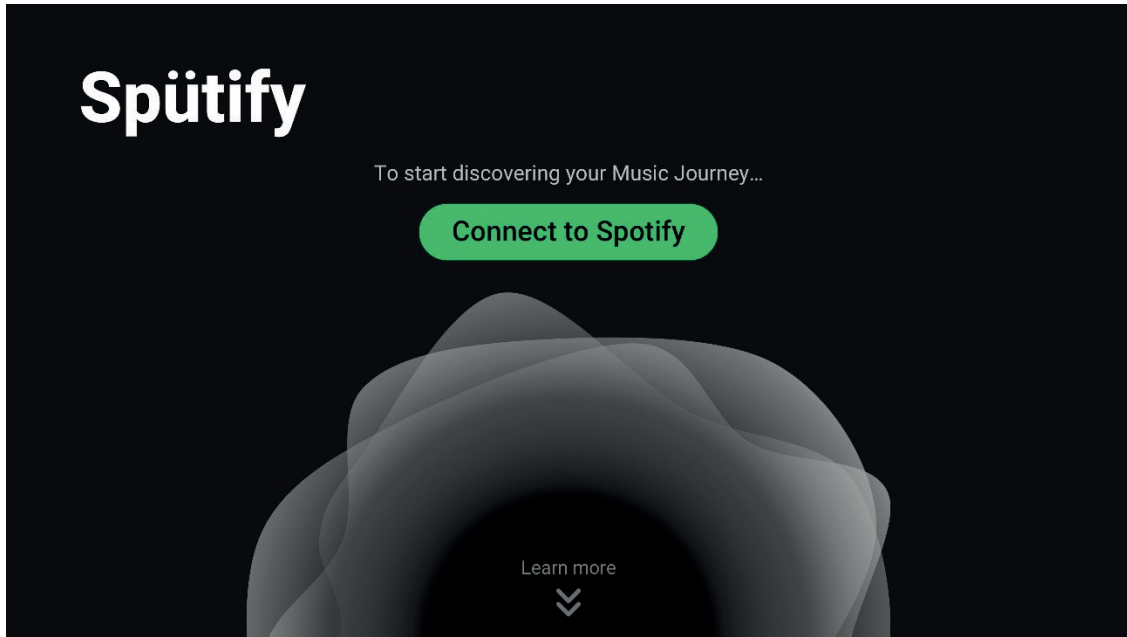


Abbildung 4: Mock-up Variante 2 Log-in Page

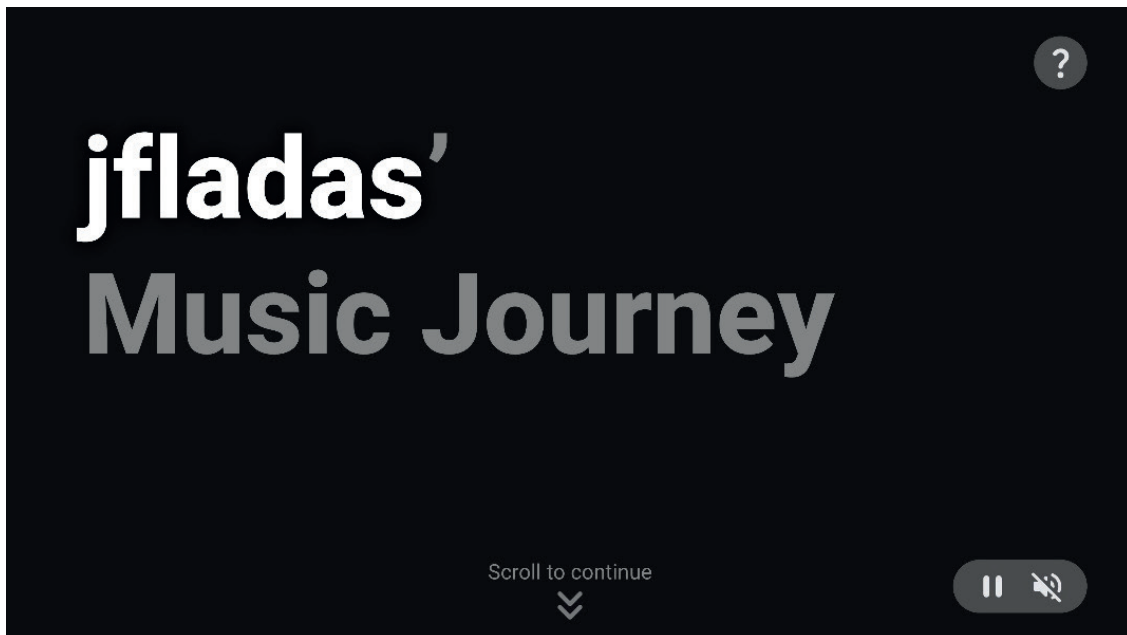


Abbildung 5: Mock-up Variante 2 Intro Page

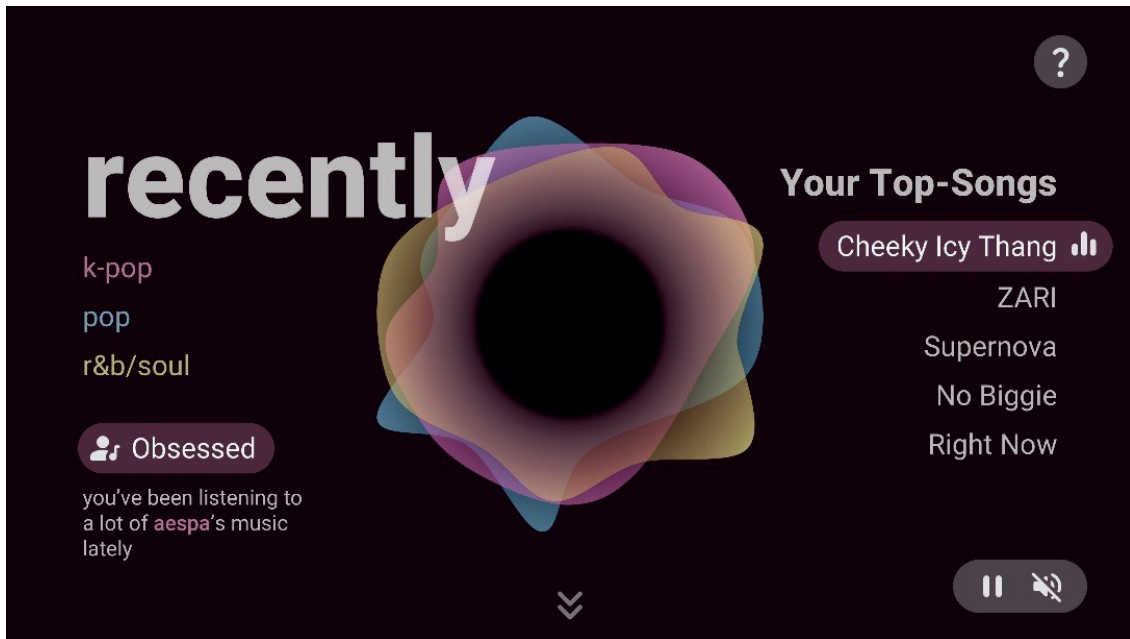


Abbildung 6: Mock-up Variante 2 Aktueller Song

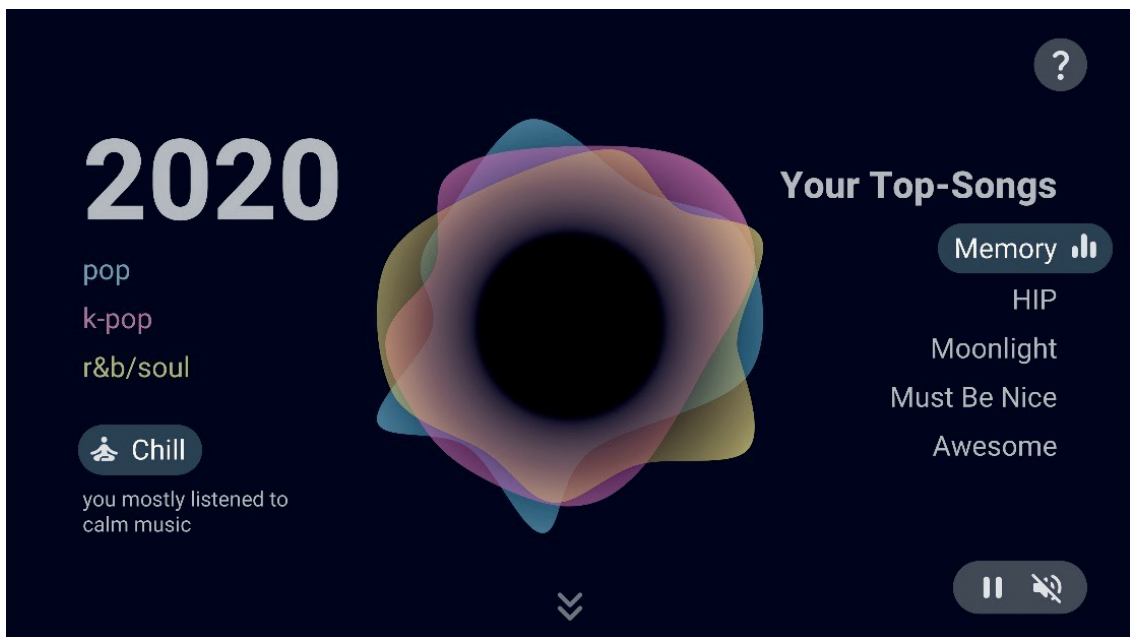


Abbildung 7: Mock-up Variante 2 Jahres Page

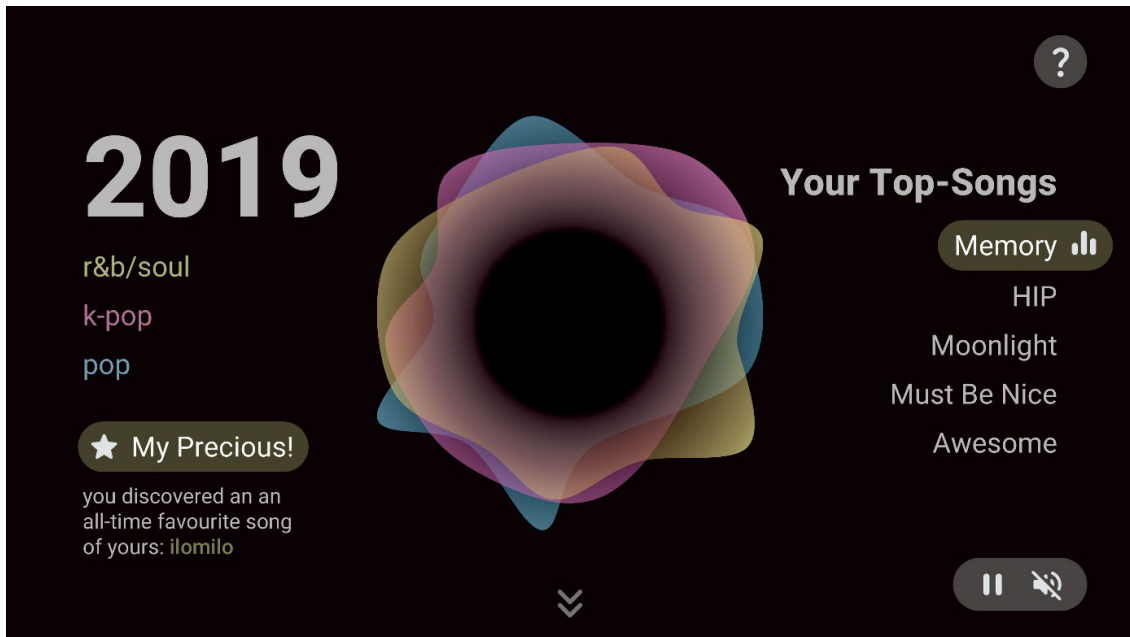


Abbildung 8: Mock-up Variante 2 Jahres Page

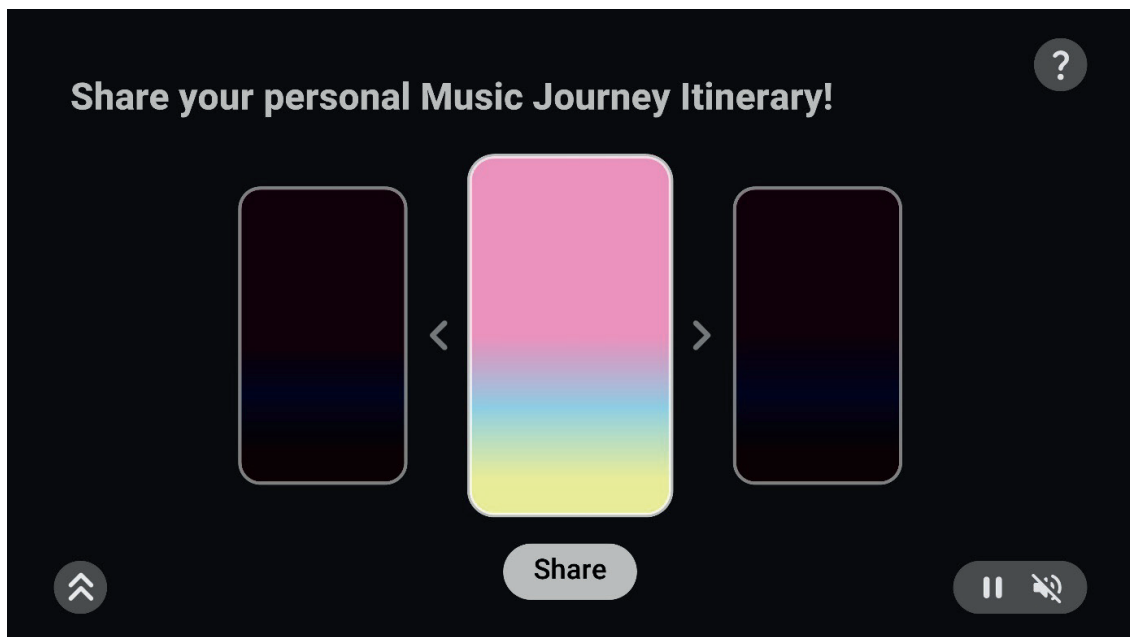


Abbildung 9: Mock-up Variante 2 Share Page

Test mit API's

The image shows two JSON responses from the Spotify API. The top response is for an audio track, and the bottom response is for an artist. In the top response, the 'valence' field is highlighted with a blue dot and a blue arrow points to a text box explaining its meaning. In the bottom response, the 'name' field is highlighted with a blue dot, and the 'popularity' field is highlighted with a red dot.

```
● "danceability": 0.744,
● "energy": 0.747,
  "key": 11,
  "loudness": -4.281,
  "mode": 0,
  "speechiness": 0.0824,
  "acousticness": 0.00691,
  "instrumentalness": 0.00000203,
  "liveness": 0.0739,
● "valence": 0.368,
● "tempo": 120.077,
  "type": "audio_features",
  "id": "4sgMbw19lXGyHcHmtFYGMm",
  "uri": "spotify:track:4sgMbw19lXGyHcHmtFYGMm",
  "track_href": "https://api.spotify.com/v1/tracks/4sgMbw19lXGyHcHmtFYGMm",
  "analysis_url": "https://api.spotify.com/v1/audio-analysis/4sgMbw19lXGyHcHmtFYGMm",
● "duration_ms": 178880,
  "time_signature": 4
}

{
  "external_urls": {
    "spotify": "https://open.spotify.com/artist/6YVMFz59CuY7ngCxTxjpxE"
  },
  "followers": {
    "href": null,
    "total": 6510163
  },
  "genres": [
    "k-pop girl group"
  ],
  "href": "https://api.spotify.com/v1/artists/6YVMFz59CuY7ngCxTxjpxE",
  "id": "6YVMFz59CuY7ngCxTxjpxE",
  "images": [...],
  "name": "aespa",
● "popularity": 80,
● "type": "artist",
  "uri": "spotify:artist:6YVMFz59CuY7ngCxTxjpxE"
}
```

musical positivity conveyed

Valence, on the other hand, describes the musical positivity conveyed by a piece of music. Songs with high valence sound more positive (e.g. happy, cheerful), while pieces with low valence sound more negative (e.g. sad, angry).

Abbildung 10: Spotify API Testings

Die blau markierten Stellen sind Variablen, die wir sicher einbauen werden, und die roten sind solche, die noch in Frage kommen.

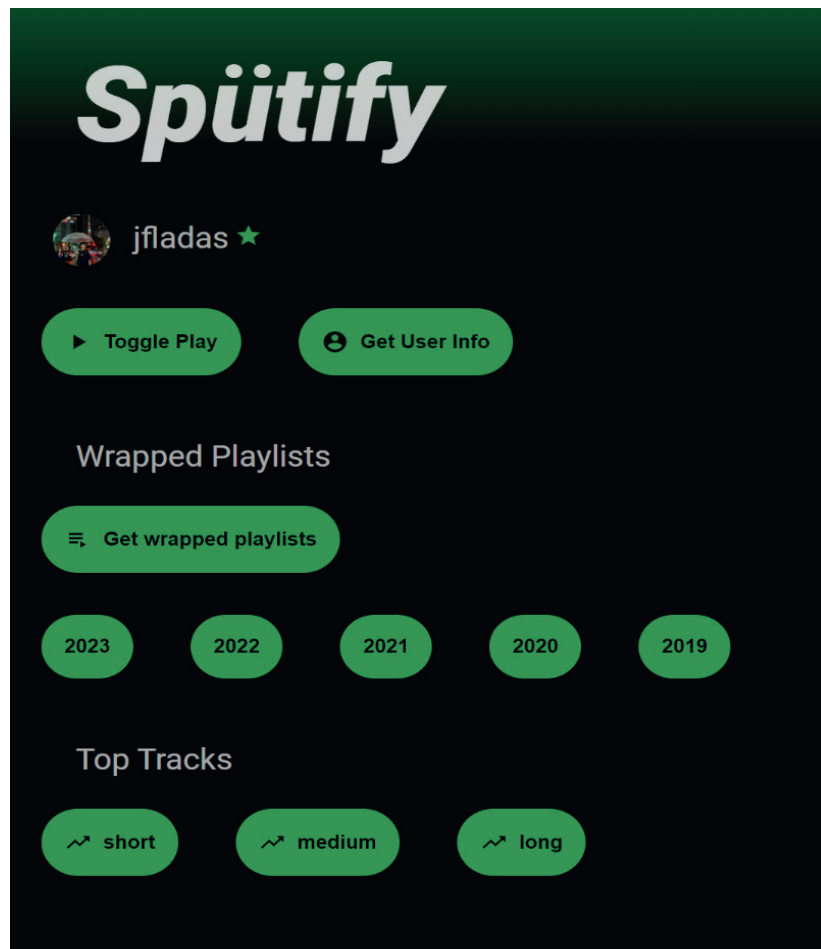


Abbildung 11: weitere Spotify API Testings

Wir haben eine kurze Testseite erstellt, um die Datenabfrage zu testen. Es ist uns gelungen, die Musikwiedergabe zu stoppen und wieder zu starten. Außerdem können wir die Benutzerdaten abrufen und das Profilbild des Nutzers anzeigen. Die Wrapped-Playlists haben wir über die Playlist-ID abgerufen. Besonders die IDs für die Jahre 2020 bis 2016 waren herausfordernd, da sie im Gegensatz zu den späteren Jahren keine universelle ID hatten. Wir mussten die ID durch eine Suchabfrage ermitteln, die nach dem jeweiligen Jahr filtert.

Projektarchitektur mit Diagramm

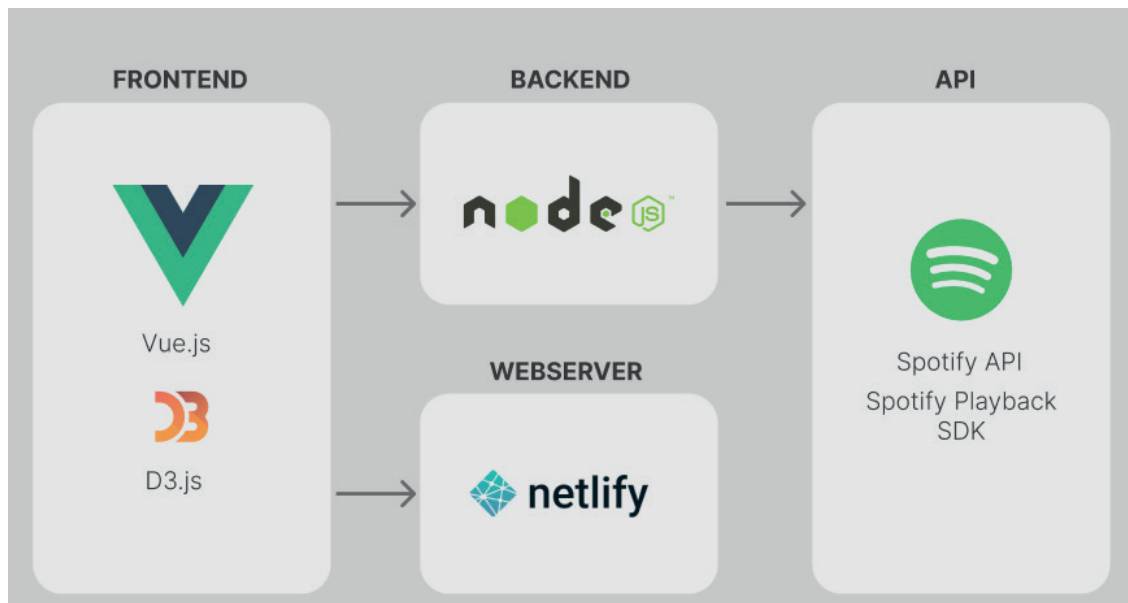


Abbildung 12: Projektarchitektur

Wir entwickeln die Musikvisualisierungen mit der D3.js-Library, um eine interaktive und dynamische Darstellung der Musikreise zu ermöglichen. Das Frontend wird mit Vue.js umgesetzt, was eine reaktive Benutzeroberfläche gewährleistet. Für das Backend nutzen wir Node.js, um die Kommunikation mit der Spotify API zu ermöglichen und Benutzerdaten sicher zu verarbeiten. Die Verbindung zwischen dem Frontend und dem Webserver erfolgt über Netlify, was eine effiziente Bereitstellung der Webanwendung sicherstellt.

Das gesamte Setup sorgt für eine skalierbare, benutzerfreundliche und performante Applikation, die in verschiedenen Browsern und auf verschiedenen Geräten gut funktioniert.

Requirementanalyse (User Stories, FR, NFR)

User Stories

1. Als Nutzer möchte ich meine Musikreise als Timeline sehen, um zu verstehen, wie sich mein Musikgeschmack im Laufe der Zeit entwickelt hat.
2. Als Nutzer möchte ich meine Musikreise als Grafik erstellen und auf sozialen Medien teilen können.
3. Als Nutzer möchte ich den aktuellen Song wiedergeben, pausieren und stumm-schalten können.
4. Als Nutzer möchte ich die Möglichkeit haben, mehr über die Hintergründe der Visualisierungen zu erfahren.

Funktionale Anforderungen

1. Verbindung zur Spotify API zur Datenabfrage
2. Nutzung des Spotify Playback SDK für die Musikwiedergabe im Browser
3. Interaktive Darstellung der Musikreise als Timeline
4. Parameter fließen in die Musikvisualisierungen ein
5. Möglichkeit, die Musikreise als Grafik zu exportieren, um sie in sozialen Medien zu teilen

Nicht-funktionale Anforderungen

1. Die Visualisierungen und die Musikwiedergabe müssen flüssig und performant laufen.
2. Die Webapplikation muss benutzerfreundlich und intuitiv bedienbar sein.
3. Datensicherheit und Schutz der persönlichen Spotify-Daten müssen gewährleistet sein.
4. Unterstützung für verschiedene Browser (Chrome, Firefox, Safari ...).
5. Responsives Design (auch für mobile Geräte).