Playlist-Builder

Authors: Holzer Tobias, Werlen Nevio

Teacher: Vianin Jeremy

Class: Inf4 App

Inhalt

[1. **Frontend**: 3](#_Toc210053865)

[**1.1 Ziel:** Das Frontend ermöglicht das Suchen, Auswählen und Exportieren von Songs in einer Playlist. Es dient als visuelles Interface für ein Musik-Playlist-Management-System. 3](#_Toc210053866)

[**1.2** **Verwendete Technologien:** 3](#_Toc210053867)

[1.3 **Designprinzipien**: 3](#_Toc210053868)

[**2** **Struktur des Frontends** 3](#_Toc210053869)

[**2.1** **Aufbau der Seite:** 3](#_Toc210053870)

[**2.3** **Footer oder Abschlussbereich** 3](#_Toc210053871)

[**3** **Technische Umsetzung** 4](#_Toc210053872)

[**2.4** **3.1 HTML-Struktur** 4](#_Toc210053873)

[**2.5** **CSS-Design** 4](#_Toc210053874)

[**2.6** **Interaktive Elemente:** 4](#_Toc210053875)

[**4** **Funktionale Aspekte** 4](#_Toc210053876)

[**5** **Erweiterungsmöglichkeiten** 5](#_Toc210053877)

[**2.7** **Technisch könnte das Frontend noch erweitert werden durch:** 5](#_Toc210053878)

[**6** **Zusammenfassung:** 5](#_Toc210053879)

1. **Frontend**:

**1.1 Ziel:** Das Frontend ermöglicht das Suchen, Auswählen und Exportieren von Songs in einer Playlist. Es dient als visuelles Interface für ein Musik-Playlist-Management-System.

* 1. **Verwendete Technologien:**
     + HTML5: Strukturierung der Inhalte
     + CSS3 / Flexbox: Styling, Layout und responsive Gestaltung
     + Google Fonts (Inter): Typografie
     + FontAwesome: Icons (z. B. Papierkorb für Löschen)
  2. **Designprinzipien**:
     + Dunkles Theme (Hintergrund #191414)
     + Grüne Akzente (#1db954) für Interaktivität
     + Abgerundete Container und Schatten für Tiefenwirkung

1. **Struktur des Frontends**

## **Aufbau der Seite:**

* + - Die Seite ist in drei Hauptbereiche gegliedert:
    - Header
    - Enthält den Projektnamen als Überschrift und einen Untertitel.
    - Dient zur Orientierung und visuellen Einführung in die Anwendung.
    - Farblich hervorgehoben durch grüne Akzente.
  1. **Hauptbereich**
     + Suchbereich: Eingabefeld für Songs oder Künstler, begleitet von einem Suchbutton.
     + Ergebnisliste / Playlist: Darstellung der ausgewählten Songs mit Titel, Künstlername und Löschoption.
     + Aktionsbuttons: Funktionen zum Hinzufügen von Songs zur Playlist und zum Export der Playlist.

## **Footer oder Abschlussbereich**

* + - In dieser Version nicht explizit vorhanden, potenziell für zukünftige Funktionen wie Navigation oder Zusatzinformationen nutzbar.

1. **Technische Umsetzung**

## **HTML-Struktur**

* + - Semantische Trennung der Inhalte über Header, Main und Sections.
    - Nutzung von Container-Elementen, um einzelne Funktionsblöcke logisch zusammenzufassen.
    - Eingabefelder und Buttons sind klar voneinander getrennt und mit Klassen für gezielte Styling- und JavaScript-Anbindung versehen.
    - Songs werden als wiederholbare Elemente innerhalb der Playlist dargestellt, was dynamisches Hinzufügen und Entfernen erleichtert.

## **CSS-Design**

* + - Farbschema: Dunkles Theme (Hintergrund dunkelgrau/schwarz) mit grünen Akzenten zur Hervorhebung von Interaktionselementen.
    - Schriftarten: Nutzung der Google-Font „Inter“ für moderne, gut lesbare Typografie.
    - Layout: Flexbox wird für vertikale Ausrichtung und Abstände zwischen Elementen verwendet.
    - Container: Hauptcontainer ist zentriert, mit maximaler Breite von 480 Pixeln, abgerundeten Ecken, Schatteneffekten und leicht transparentem Hintergrund.

## **Interaktive Elemente:**

* + - Buttons und Eingabefelder besitzen visuelle Feedbackmechanismen (Hover- und Fokus-Effekte).
    - Löschen-Buttons für Songs sind rund und mit Icons versehen.
    - Responsivität: Die Anwendung ist flexibel und passt sich an verschiedene Bildschirmgrößen an, insbesondere im mobilen Bereich.

1. **Funktionale Aspekte**

* **Suchfeld**: Ermöglicht die Eingabe von Songtiteln oder Künstlernamen. Die visuelle Rückmeldung bei Fokus unterstützt die Benutzerfreundlichkeit.
* **Playlist**: Darstellung der Songs in einer tabellarischen Form mit klarer Trennung zwischen Titel, Künstler und Aktionselementen.
* **Buttons**: Einheitliches Design für alle Interaktionspunkte, farblich konsistent und durch Hover-Effekte hervorhebbar.
* **Löschfunktion**: Jede Playlist-Zeile besitzt einen dedizierten Button zum Entfernen des Songs aus der Liste.
* **Exportfunktion**: Möglichkeit, die zusammengestellte Playlist zu speichern oder herunterzuladen.

1. **Erweiterungsmöglichkeiten**

## **Technisch könnte das Frontend noch erweitert werden durch:**

* + - JavaScript-Integration: Für dynamische Suchergebnisse, das Hinzufügen und Entfernen von Songs in Echtzeit.
    - Verbesserte Responsivität: Optimierung für Tablets und kleine Smartphones.
    - Animations- und Übergangseffekte: Für flüssige Interaktionen beim Hinzufügen oder Löschen von Songs.

1. **Zusammenfassung:**

Das Frontend des Playlist-Builders ist modular, semantisch korrekt und responsiv aufgebaut. Es kombiniert moderne Web-Technologien mit einem klaren Designkonzept, das sowohl visuell ansprechend als auch funktional ist. Durch die klare Trennung von Struktur, Layout und Interaktion ist es leicht erweiterbar für dynamische Features oder API-Integration.

Backend:

Jenkins: