

SoundWeave: The *iMovie* for your audio tracks

Introduction

The project's goal was to create an intuitive and dynamic audio pill player, akin to a simplified version of iMovie but for audio. This document outlines the approach I took, the design decisions made, and the challenges encountered during the development process.

You can access the web app here: <https://parthskansara.github.io/SoundWeave>

Here's a link to the GitHub repository: <https://www.github.com/parthskansara/SoundWeave>

Approach

Technology Stack

- **Framework:** React
- **State Management:** React Context for global state management of the tracklist
- **UI Components:** Tailwind CSS for the user interface components
- **Icons:** Material UI for conventionally recognizable icons
- **Audio Handling:** Crunker, an efficient wrapper for the Web Audio API, for managing audio playback and manipulation
- **Drag & Drop:** HTML5's native attributes and drag events for implementing drag-and-drop features
- **Audio Player:** React H5 Audio Player, for providing a UI to interact with the audio

Features Implemented

1. **Audio Track Upload:** Users can upload audio files which are then visualized as individual "audio pills" on a timeline.
 - a. **Drag and Drop:** Users can drag audio files onto the workspace. Appropriate validation is performed to ensure only .mp3 files can be uploaded.
 - b. **Upload button:** Users can also upload their local files using the Upload button. This is also useful for mobile users.
 - c. **Delete button:** To remove a track from the workspace, users can simply drag it over the Delete button at the bottom of the workspace.
2. **Dynamic Timeline:** Audio pills can be shifted around on the timeline to change the order of playback. For reordering, we have used a marker at both ends of the audio pill. If a pill is dragged and dropped on a marker, the track order is changed. The marker is highlighted in yellow if a pill is dragged onto it, indicating a potential reordering, to make the UX more intuitive.

3. **Playback Control:** Users can play, pause, and skip through audio tracks, with real-time updates on the timeline.
4. **Download Tracks:** Once the user has decided on the order of the tracks, they can download the concatenated track using the Download button at the top right.

Design Decisions

UI/UX Design

- **Minimalist Interface:** A clean and distraction-free UI was implemented to keep the focus on functionality.
- **Color Palette:** To infuse a retro look, a Cappuccino color palette was used, characterized by warm shades of brown, cream, and beige, reminiscent of vintage aesthetics.
- **Interactive Timeline:** The timeline is the central element, designed to be interactively manipulated by dragging and dropping audio pills.
- **Responsive Design:** Ensuring that the application is usable on various screen sizes and devices.

Audio Management

- **Client-Side Processing:** To minimize latency, audio processing is done in the browser, leveraging Crunker, a wrapper for the Web Audio API.
- **Non-Destructive Editing:** Changes in the timeline do not alter the original audio files, allowing for flexible editing.

Challenges Encountered

Intuitive UI for reordering

Creating an intuitive user interface that allows for easy reordering of audio pills presented a unique set of challenges. The goal was to enable users to drag and drop audio segments seamlessly within the timeline, requiring a delicate balance between responsive drag-and-drop functionality and maintaining an accurate representation of the tracklist order. This was achieved through the implementation of marker elements at the end of each audio pill, along with custom drag handlers and state updates that not only ensure a smooth user experience but also prevent any disruptions in the playback or editing process. Iterative design improvements were key in refining this feature to meet our standards of intuitiveness and efficiency.

Syncing Audio and UI

Ensuring that the UI reflects the actual playback state, especially when multiple tracks are shifted around quickly, was challenging. This was addressed by implementing a robust state management system that tightly couples the UI with the audio playback engine.

Concatenation of Audio Tracks

Developing a reliable and efficient method for concatenating multiple audio tracks into a single file, while preserving audio quality, proved to be technically challenging. Utilizing the Crunker library to handle the intricacies of audio encoding and ensuring seamless transitions between tracks required extensive testing and fine-tuning.

Future Enhancements

- **Multi-Track Support:** Allowing users to add multiple parallel tracks for layered playback.
- **Audio Trimming and Splitting:** Implementing a set of features for allowing users to trim and split individual audio pills.
- **Detailed UI for Timeline:** Implementing a more detailed UI for the Timeline, including timestamps and proportionate lengths of the song cards

Conclusion

This project was an excellent opportunity to explore dynamic content manipulation and complex user interactions in a web application. The challenges encountered were significant learning experiences, and overcoming them has provided a robust and user-friendly audio pill player. Additionally, considerable attention was dedicated to creating a visually appealing design, employing tools like contrast checkers to ensure the accessibility needs were met, underscoring my commitment to delivering not only functional but also accessible and aesthetically pleasing web solutions. I am confident that this project aptly demonstrates my ability to handle the front-end development requirements for dynamic and interactive web applications.