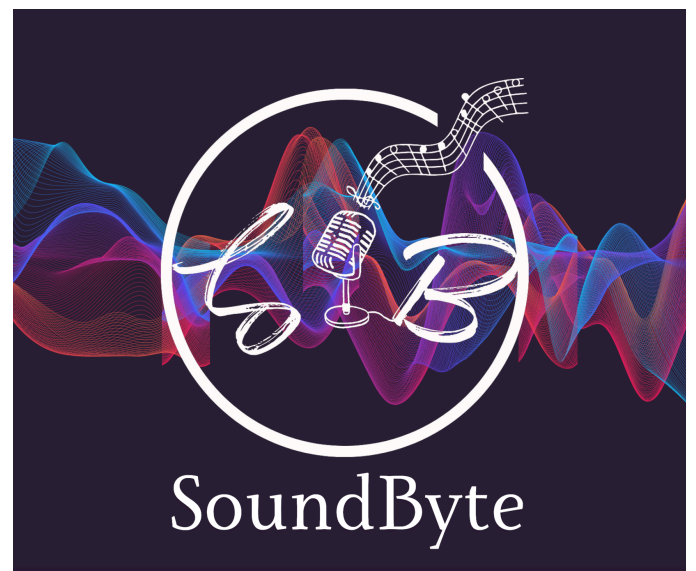


# SoundByte Project



## PROJECT OVERVIEW:

This project aims to expand upon SoundByte's existing pitch visualization app by developing a comprehensive music loading and processing tool tailored for vocalists. While the current app displays real-time pitch accuracy, our goal is to transform it into a robust practice and performance enhancement platform. By allowing users to load sheet music and audio, compare their vocals to the intended melody, and receive targeted feedback, we're building a tool that bridges technology and vocal training.

## KEY FEATURES:

The proposed software will include the following core features:

- **Sheet Music Import & Processing:** Load standard notation files (e.g., MusicXML) and visualize them alongside vocal input.
- **Pitch Matching & Scoring:** Automatically compare the singer's pitch to the song's melody and generate a performance score.
- **Problem Section Detection:** Identify and highlight difficult or off-pitch sections for targeted practice.
- **Backing Track Integration:** Allow users to sing along with instrumental tracks to simulate real performance scenarios.
- **Visual Feedback Enhancements:** Improved UI for note accuracy, pitch stability, and rhythmic timing.

## **TECHNOLOGY:**

To build this software application, we propose utilizing the following technologies:

### **General UI:**

- SwiftUI – The core UI framework used for building the iOS app, offering a seamless and responsive user experience.
- AVFoundation – For handling audio recording and playback within the app.
- AudioKit – A powerful open-source audio processing library that integrates well with Swift for pitch detection, audio effects, and signal analysis.
- MusicKit / CoreMIDI – For working with music files, MIDI input, and possibly sheet music integration.

### **Backend / API:**

- Node.js or Python (Flask/FastAPI)
- Firebase or AWS Amplify (data storage)

### **Sheet Music Rendering:**

- VexFlow (via WebView) – Render dynamic sheet music within the app using a WebView
- MusicXML Parsing – Either server-side or in-app parsing of MusicXML files to extract note data for pitch comparison.

## **PROJECT DETAILS:**

- 4 person team
- Music experience or knowledge not required, but encouraged!
- 1 Mac-book provided