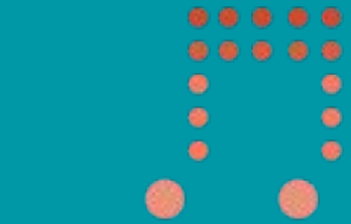


Pitch Perfect

Your on-the-go virtual vocal coach

Team: Ensemble Methods - Funmbi Jaiyeola, Sai Korivi, Carlos Taveras

Use Case

Pitch Perfect aims to address the the high cost and exclusivity of singing lessons taught by an instructor by democratizing high-quality singing lessons to users across the world, from the comfort of their home.

Areas:

- Signal Processing
- Software Systems

Requirements

- Audio:
  - 90% pitch detection accuracy
  - 95% timing accuracy
- Posture:
  - 95% alignment of perfect posture (some motion is acceptable)
- User Interface:
  - Max lag time of 200 milliseconds between singing and UI display for real-time feedback
  - Lyrics for user to follow along with
  - Access to storage space that contains all recorded files of practices/exercises
  - Songs and exercises that differentiate good and bad singing
  - Visual data to show progress over time
  - Intuitive user interaction

User Interface

- Real-Time Visual Feedback:
  - Pitch accuracy
  - Good Posture
  - On-beat Tempo

- Representation of Analytics and Progress of User
  - Data visualization of improvement
  - Patterns

Audio Analysis

- Latency
  - Our product must provide users with real-time pitch and timing feedback, thus we are constrained in time allowed for voice processing

- Metric Choices
  - Our metrics must prove to be reliable indicators of good singing performance

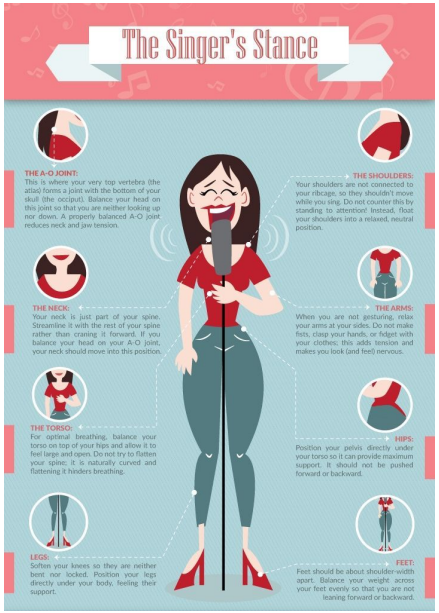
- Metric Extraction and Comparison
  - Our product must be able to extract metrics from a song from which we will use as a template to evaluate users' performances

Posture Analysis

- A person can not stay still without some motion
  - Makes posture detection slightly difficult
  - Latency between video capture becomes an issue

- Accounting for trivial motion such as:
  - Bobbing of the head
  - Swaying of hips

- Accounting for the presence of microphone
  - Microphone stand
  - Holding microphone



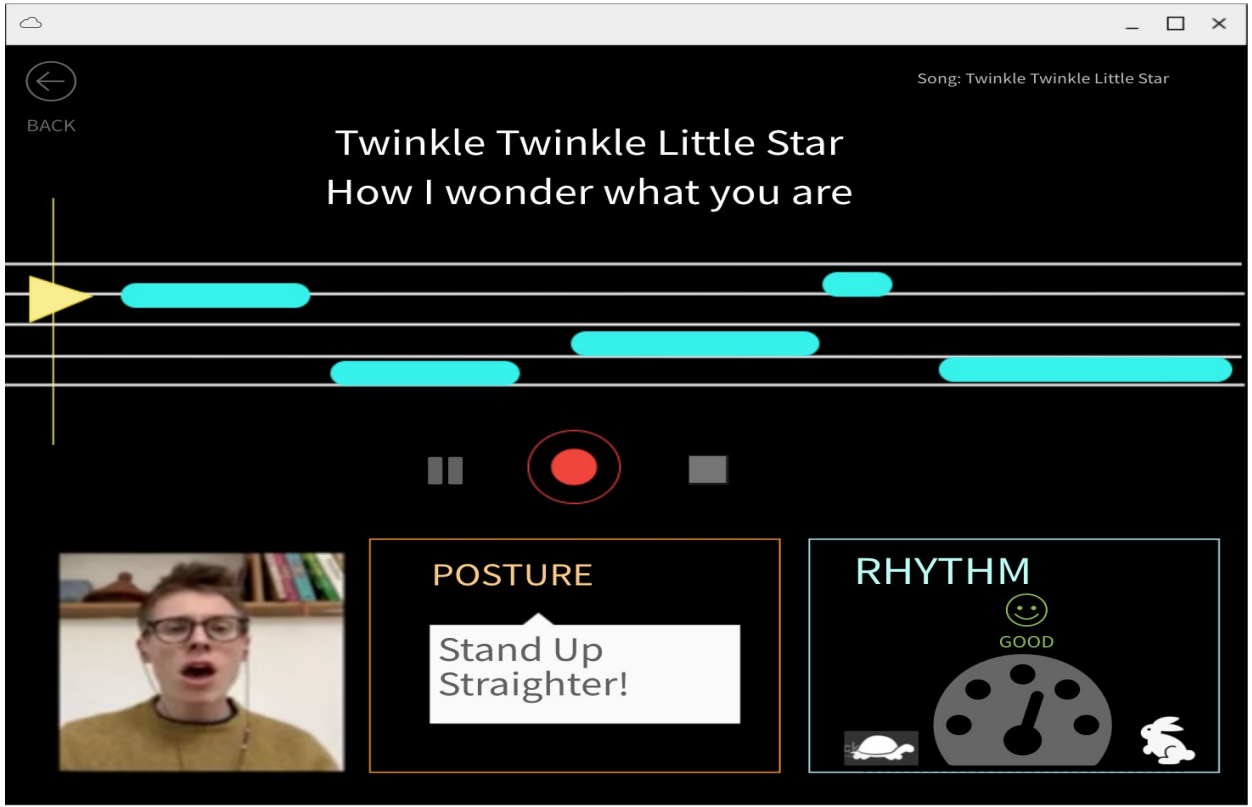
Solution Approach

- Hardware:
  - Microphone
  - External Camera

- Software:
  - Web Application using Django Framework
  - WebAudio API for audio visualizations
  - User data and recordings storage on database
  - Cloud Deployment: AWS
  - OpenCV for posture detection
  - SciPy for voice processing

- Music Theory
  - Monophonic music (no instrumentals)
  - Links to simple breathing exercise

Solution Approach



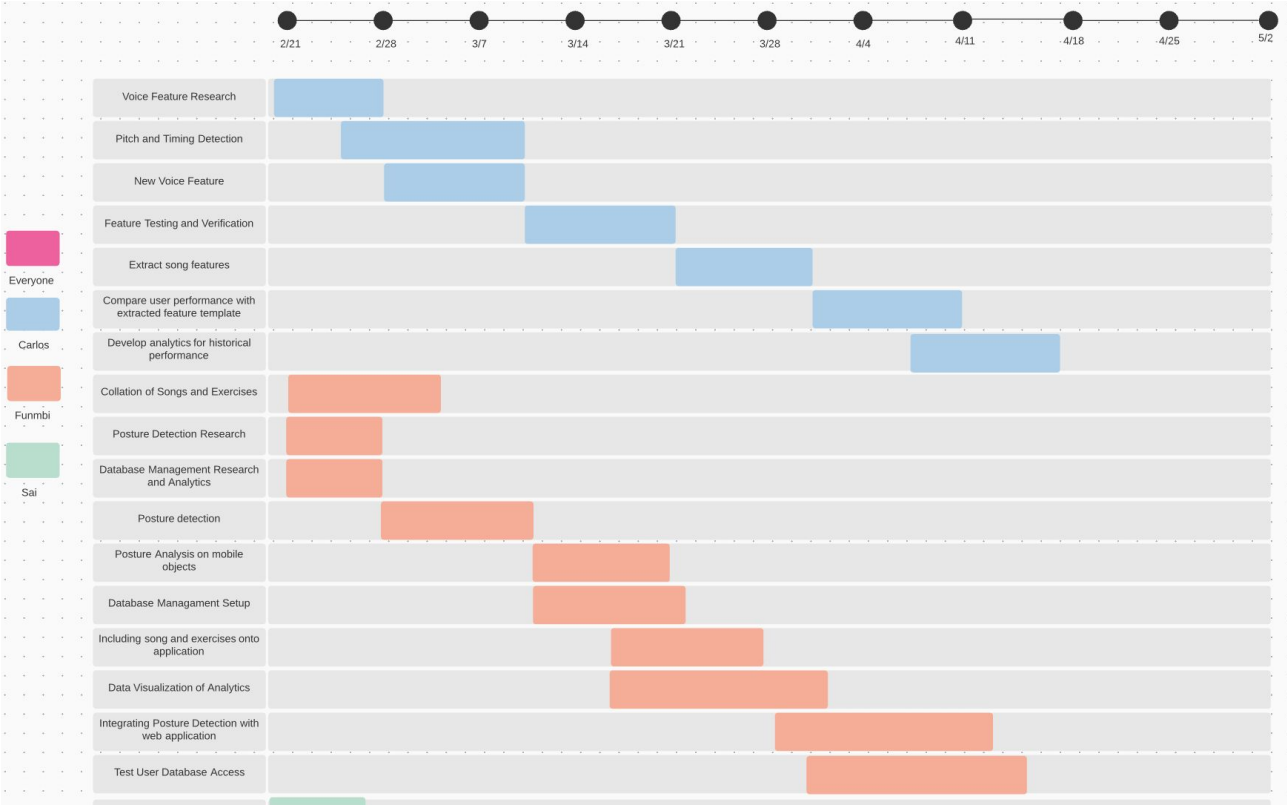
Testing, Verification, and Metrics

- Real-Time Visual Feedback:
  - Human Eye -> Analyze metrics are met from what is done
  - **Metric:** Feedback Delay < 200 ms
- Pitch and Timing Accuracy:
  - Test our pitch and timing detection algorithms against known pure tones
- Posture Detection:
  - Use existing photos of good & bad posture
  - Test using videos of members of the group
  - Test on non-human objects
- Storage
  - Location check (on database to user)
  - Accessibility with user credentials

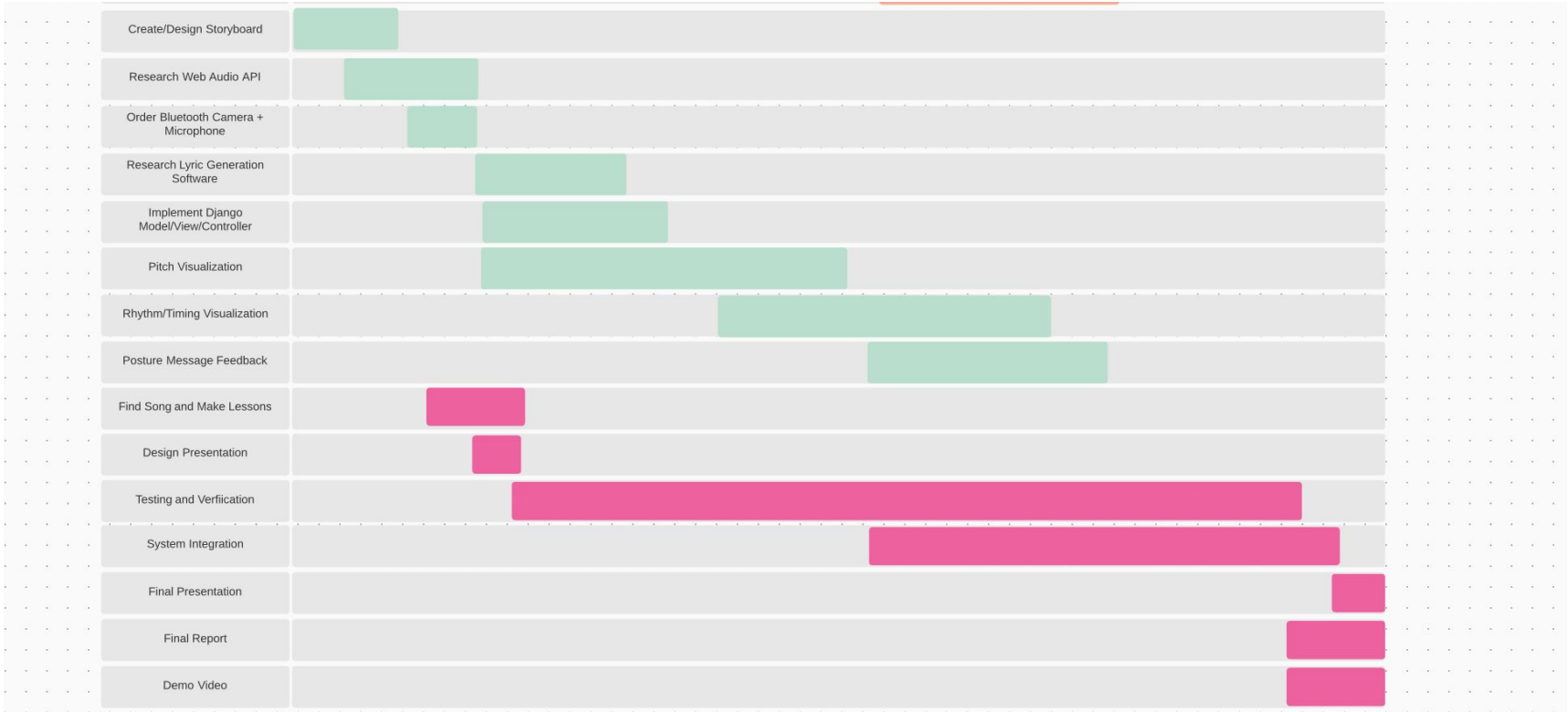
Tasks & Division of Labor

Carlos	Funmbi	Sai
Pitch and Rhythm Detection	Posture Detection	Real-time Visual Feedback
Song Feature Extraction	Cloud Deployment	Audio Visualization
Performance Data Analytics	Database Management	Django Views/Models/URLS

Schedule


2/21
2/28
3/7
3/14
3/21
3/28
4/4
4/11
4/18
4/25
5/2
Voice Feature Research
Pitch and Timing Detection
New Voice Feature
Feature Testing and Verification
Extract song features
Compare user performance with extracted feature template
Develop analytics for historical performance
Collation of Songs and Exercises
Posture Detection Research
Database Management Research and Analytics
Posture detection
Posture Analysis on mobile objects
Database Managament Setup
Including song and exercises onto application
Data Visualization of Analytics
Integrating Posture Detection with web application
Test User Database Access
Everyone
Carlos
Funmbi
Sai

Schedule


Create/Design Storyboard
Research Web Audio API
Order Bluetooth Camera + Microphone
Research Lyric Generation Software
Implement Django Model/View/Controller
Pitch Visualization
Rhythm/Timing Visualization
Posture Message Feedback
Find Song and Make Lessons
Design Presentation
Testing and Verfiication
System Integration
Final Presentation
Final Report
Demo Video