JustPerform

“A better karaoke experience”

Team D1
Hugo Martinez, Kiera Cawley, Aleks Watkins

Use Case

The Problem:

Karaoke at-home experience is fragmented. Users must secure equipment, search different platforms to look for songs that may have suboptimal audio, and do not receive feedback.

Our Solution:

JustPerform creates one seamless karaoke experience that removes audio from users' personal music library and analyzes audio and motion to provide feedback for improvement.

Use Case Requirements

Objectives:

Microphone	10ms latency
Speaker	10ms latency
Song accuracy	80% accurate vocal removal 98% accurate lyrics
User feedback system	90% user agreement in testing
Program latency	<10s latency to start system <10s latency for feedback

Design Requirements

Speaker:

- Less than 15 lbs
- Capacity for >= 80 dB
- Filter latency < 10 ms

Microphone:

- Bluetooth functionality for mobility
- Connection latency < 10 ms

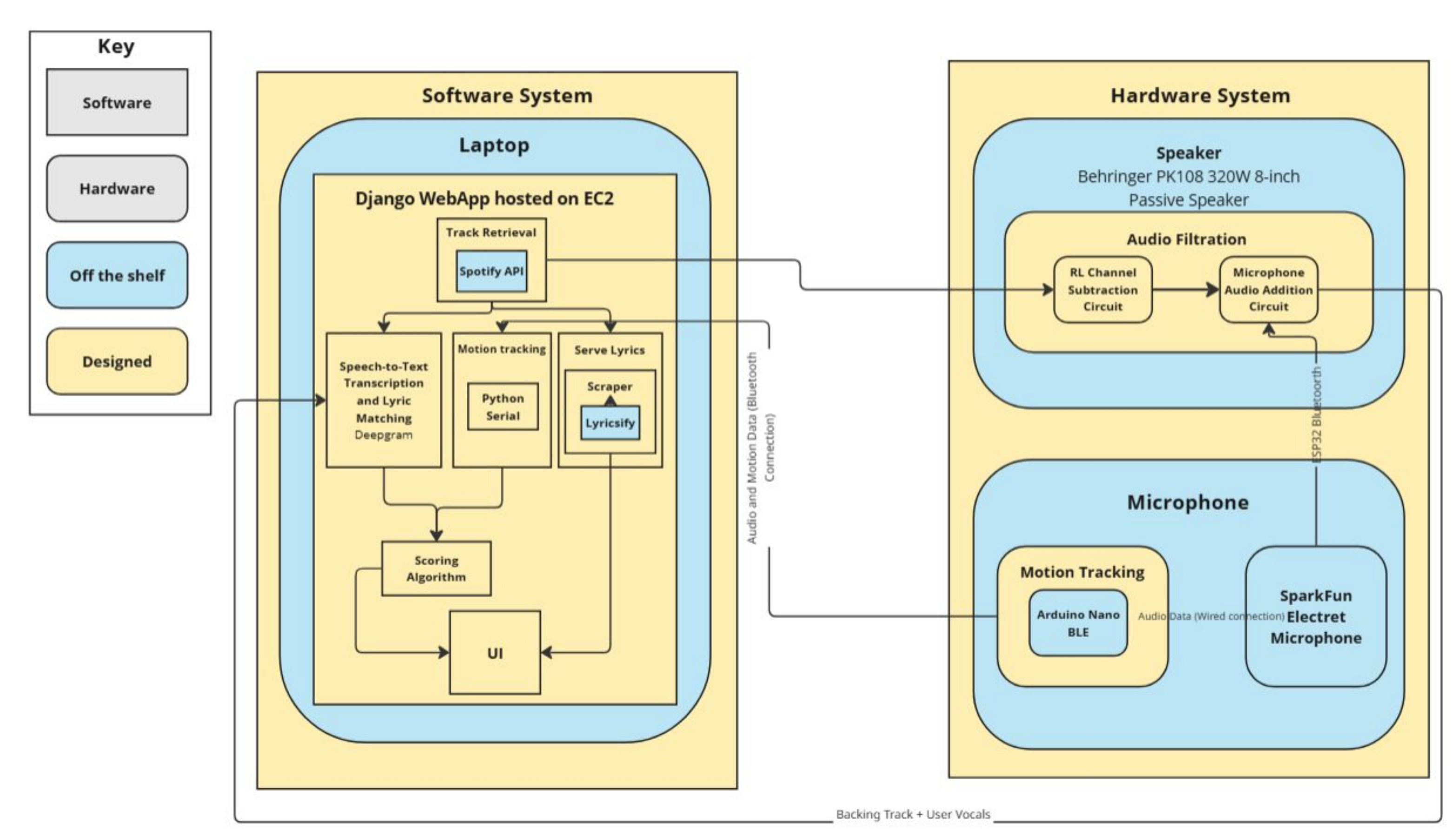
UI:

- Lyric sourcing < 3 s
- Lyric timing < 200 ms of delay
- Audio scoring < 3s delay upon signal
- Motion scoring < 3s delay upon signal

Solution approach

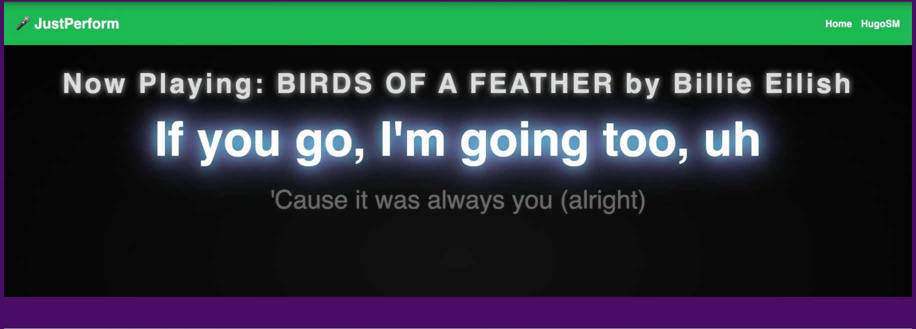
Feature	Motivation	Considerations
Connect to user's personal music library Auto populate and display lyrics Remove vocal track in real time	Unite the tedious parts of karaoke experience into one location	Appropriate sizing for those with visual disabilities Non-resource intensive solutions
Process audio for singing timing and volume	Focus on typical karaoke experience and gamification aspect; pivoted from improvement feedback	Accessible to those with different accents
Process motion for activity levels and beat matching	Promote performance and gamification experience beyond just vocals	Accessible to those with mobility issues, as long as they can hold the microphone

Solution Approach


Key
Software
Hardware
Off the shelf
Designed
Software System
Laptop
Django WebApp hosted on EC2
Track Retrieval
Spotify API
Speech-to-Text Transcription and Lyric Matching
Deepgram
Motion tracking
Python Serial
Serve Lyrics
Scraper
Lyricsify
Scoring Algorithm
UI
Audio and Motion Data (Bluetooth Connection)
Hardware System
Speaker
Behringer PK108 320W 8-inch Passive Speaker
Audio Filtration
RL Channel Subtraction Circuit
Microphone Audio Addition Circuit
ESP32 Bluetoorth
Microphone
Motion Tracking
Arduino Nano BLE
Audio Data (Wired connection)
SparkFun Electret Microphone
Backing Track + User Vocals

Complete Solution

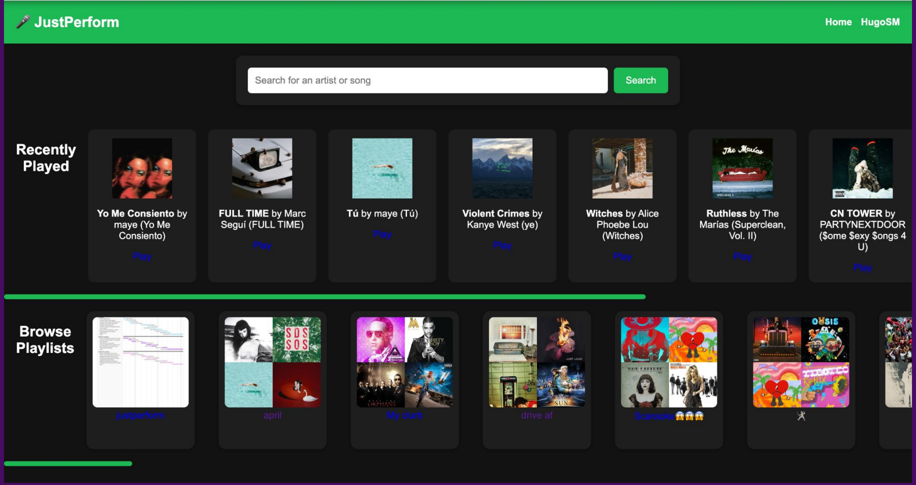
- Log into Spotify Web Application deployed to URL
- Select from recently played, playlists, or search tracks
- Lyrics and a beat are provided
- Left-Right Subtraction removes vocals in real time
- Handheld bluetooth mic tracks audio and motion
- Real time feedback based on motion and audio analysis



Now Playing: BIRDS OF A FEATHER by Billie Eilish

If you go, I'm going too, uh

'Cause it was always you (alright)



Recently Played

Yo Me Consiento by maye (Yo Me Consiento)
Play

FULL TIME by Marc Seguí (FULL TIME)
Play

Tú by maye (Tú)
Play

Violent Crimes by Kanye West (ye)
Play

Witches by Alice Phoebe Lou (Witches)
Play

Ruthless by The Marías (Superclean, Vol. II)
Play

CN TOWER by PARTYNEXTDOOR ($ome $exy $ongs 4 U)
Play

Browse Playlists

justperform

april

My clurb

drive af

Scaraoke😱😱😱

🕺

Testing Verification and Metrics

Specification	Method	Requirement	Result
Accuracy of acceleration values	Moving the mic in a series of distinct motions	Measured values should match expected +/- 10%	Avg Deviation: 5%
Accuracy of change of direction algorithm	Move the mic in a series of sharp direction changes, record time of movements	Measured values should match expected +/- 10%	Avg Deviation: 7%
Accuracy of scoring algorithm	Move the mic along to the beat high accuracy and with internationally erratic movements, compare scoring	Achieve 90% test user agreeability	95% agreeability (10 users)
Accuracy of microphone input	Have users sing into the mic, compare detected words with lyrics	Detected words should match 90% of lyrics	92% accuracy (10 users)

Testing Verification and Metrics

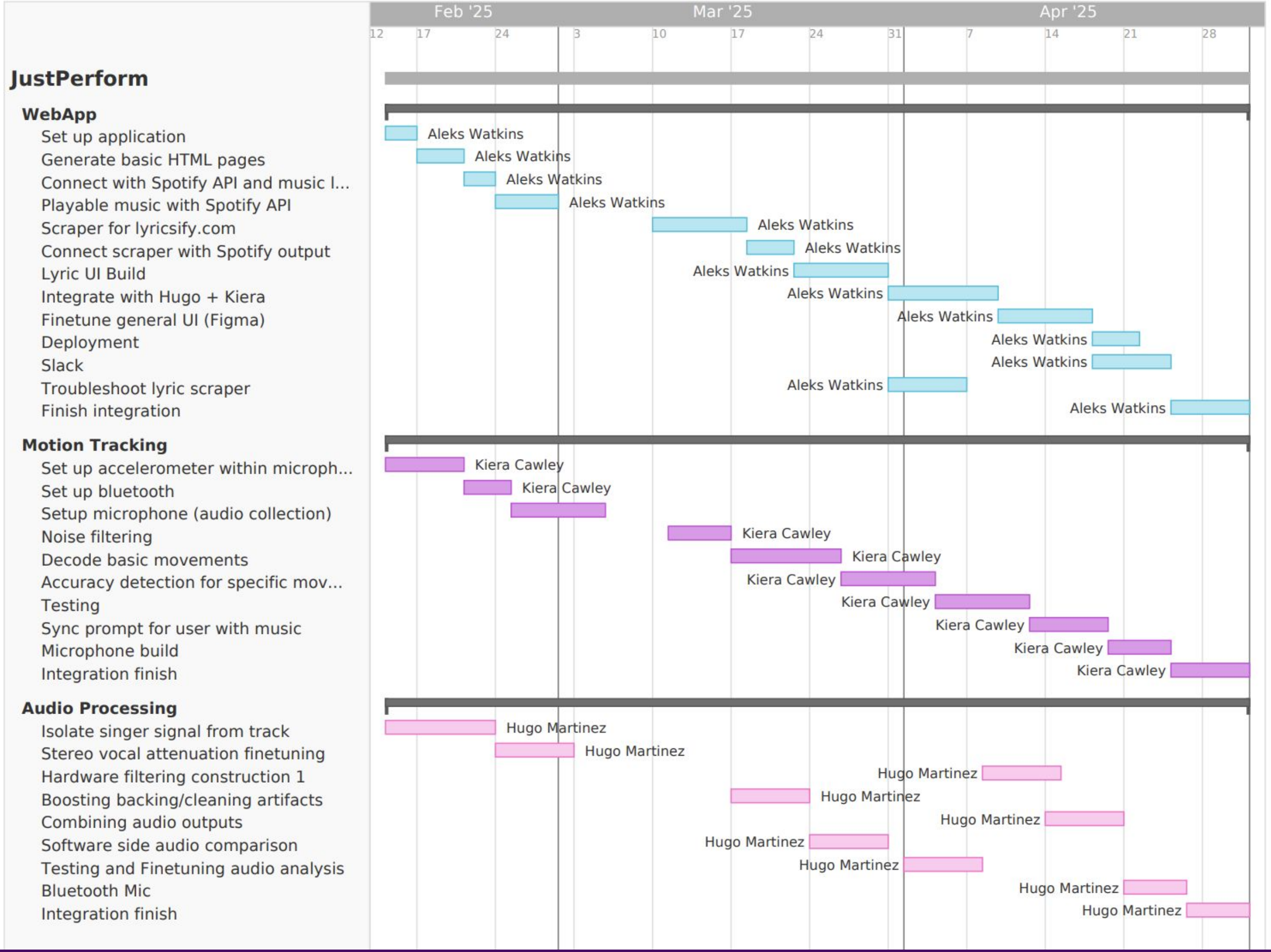
Specification	Method	Requirement	Result
System boot latency (load & player)	API calls through Spotify and rendering via Django	< 10 seconds	< 1 second
Lyric retrieval latency	Using requests/selenium to pull website and BeautifulSoup to scrape	< 3 seconds	< 2 seconds*
Lyric display latency	Cached lyrics on start up with dynamic JS refresh every 100ms	< 200 ms	< 100 ms
Audio and motion feedback latency	Backend updates shared state and frontend polls for updates	< 3 seconds	Pending integration

Testing Verification and Metrics

Specification	Method	Requirement	Result
Left-Right Subtraction Latency	Record time from play on Web App to speaker output	< 10 ms	< 10 ms (unnoticeable)
Left-Right Subtraction Accuracy	Compare audio level of lead vocals before and after attenuation	< ⅕	~ ¼*
Bluetooth Signal Latency	Record time from microphone input to speaker output	< 10 ms	Pending Integration
Speech-To-Text Accuracy	Compared singing input to Deepgram model transcription	> 85%	~ 70%*

Trade Offs

Design Choice	Pros	Cons
Left and Right Channel Subtraction	Allows for real time processing, effective, simple	Versatility reduced to only songs in stereo
Speech to text and lyric comparison	Provides a metric that can be analyzed in real time	Scoring can generally be unreliable, relies on accurate lyrics being provided
Accelerometer System	Lower latency, data provided directly	Adds more hardware Potentially complex wiring
ESP32 Bluetooth Connection	Less wires, allows for more freedom with the microphone	Obfuscates the path from microphone to web app


Feb '25
Mar '25
Apr '25
12
17
24
3
10
17
24
31
7
14
21
28
JustPerform
WebApp
Set up application
Aleks Watkins
Generate basic HTML pages
Aleks Watkins
Connect with Spotify API and music l...
Aleks Watkins
Playable music with Spotify API
Aleks Watkins
Scraper for lyricsify.com
Aleks Watkins
Connect scraper with Spotify output
Aleks Watkins
Lyric UI Build
Aleks Watkins
Integrate with Hugo + Kiera
Aleks Watkins
Finetune general UI (Figma)
Aleks Watkins
Deployment
Aleks Watkins
Slack
Aleks Watkins
Troubleshoot lyric scraper
Aleks Watkins
Finish integration
Aleks Watkins
Motion Tracking
Set up accelerometer within microph...
Kiera Cawley
Set up bluetooth
Kiera Cawley
Setup microphone (audio collection)
Noise filtering
Kiera Cawley
Decode basic movements
Kiera Cawley
Accuracy detection for specific mov...
Kiera Cawley
Testing
Kiera Cawley
Sync prompt for user with music
Kiera Cawley
Microphone build
Kiera Cawley
Integration finish
Kiera Cawley
Audio Processing
Isolate singer signal from track
Hugo Martinez
Stereo vocal attenuation finetuning
Hugo Martinez
Hardware filtering construction 1
Hugo Martinez
Boosting backing/cleaning artifacts
Hugo Martinez
Combining audio outputs
Hugo Martinez
Software side audio comparison
Hugo Martinez
Testing and Finetuning audio analysis
Hugo Martinez
Bluetooth Mic
Hugo Martinez
Integration finish
Hugo Martinez