WD Diploma - Capstone Project

Project Title: aimBeats
Author: Brandon Ong
Cohort: Web Development 2022 (Part-time)

Project Overview

1.1 Description

A circle clicking rhythm game where users test their dexterity while playing along to their favorite tunes.

1.2 Problem

aimBeats gives the casual music listener an easy to learn and personalized music listening and gaming experience, which puts their dexterity to the test. The game was inspired by both Osu! - a popular online rhythm clicking game - along with Aim Labs - a online game/workshop that assist FPS (first person shooter) gamers with improving their aim. aimBeats is a mix of the two in that users will be able to search and select songs they enjoy listening to and test their hand-eye coordination in clicking the circles that randomly appear to the beat of the music.

1.3 User Profile

The app will be geared towards:

- Casual music listeners as a secondary activity while listening to their favorite music.
- FPS gamers as a warm up exercise.

1.4 Requirements: Use Cases and Features

As a user:

- I can connect my Spotify account to search and play my favorite songs
- I can compete against friends by setting new high scores
- I can pick my favorite songs to play and improve my hand-eye coordination

1.5 Tech Stack and APIs

- Front-End: React
- Back-End: NodeJS and Express
- Libraries & Packages
  - React Router 5.3
  - React Spring

    - Material UI
    - react-particles-js
- Spotify API
- PassportJS
- MySQL & KnexJS

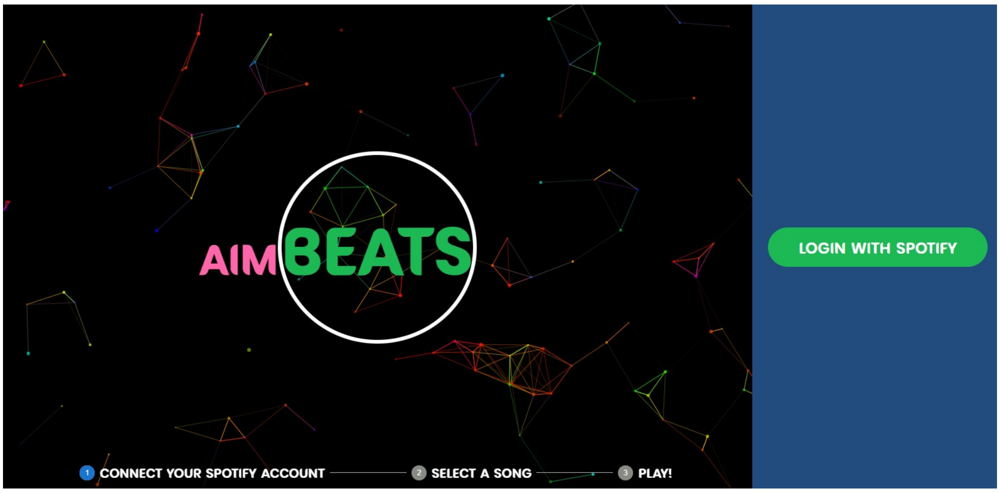
2. Client-Side Implementation

Site Map

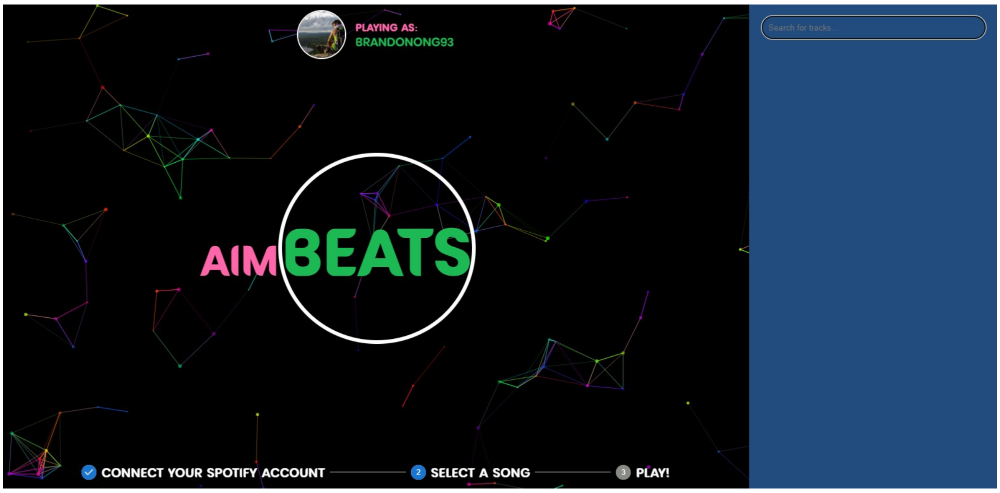
- Home / Login Screen
- Song Search
- Gameplay
- Game High Score Screen

Screen Details

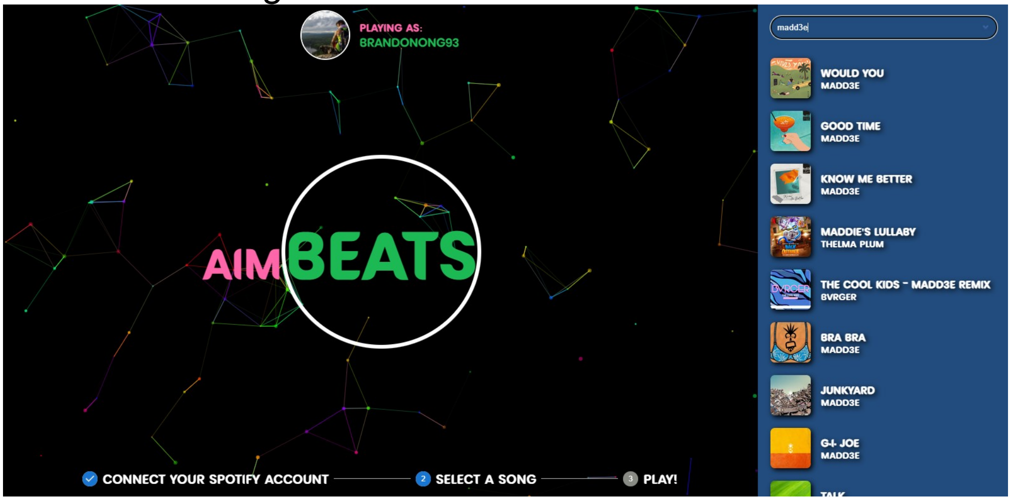
Login



Search Songs



Search/Select Songs



madd3e

WOULD YOU
MADD3E

GOOD TIME
MADD3E

KNOW ME BETTER
MADD3E

MADDIE'S LULLABY
THELMA PLUM

THE COOL KIDS - MADD3E REMIX
BVRGER

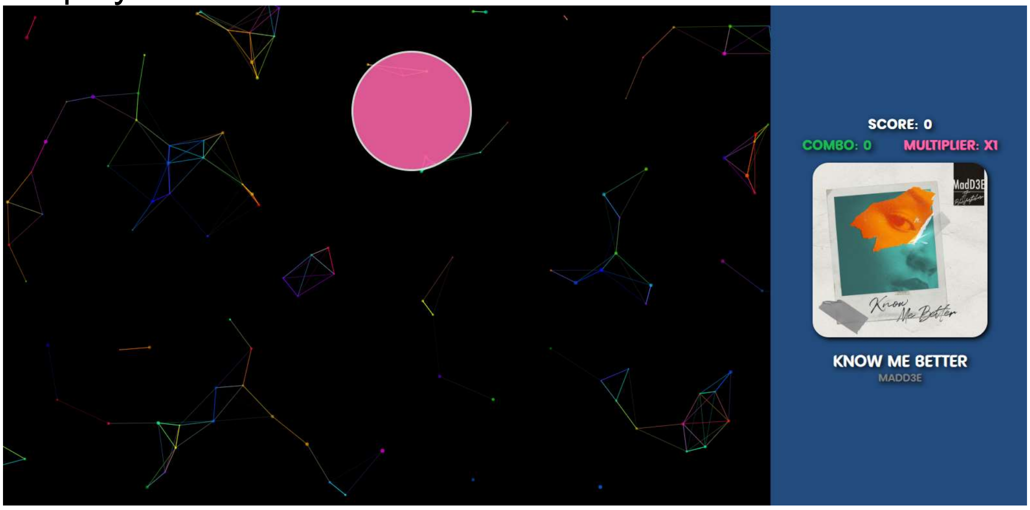
BRA BRA
MADD3E

JUNKYARD
MADD3E

G.I. JOE
MADD3E

TALK

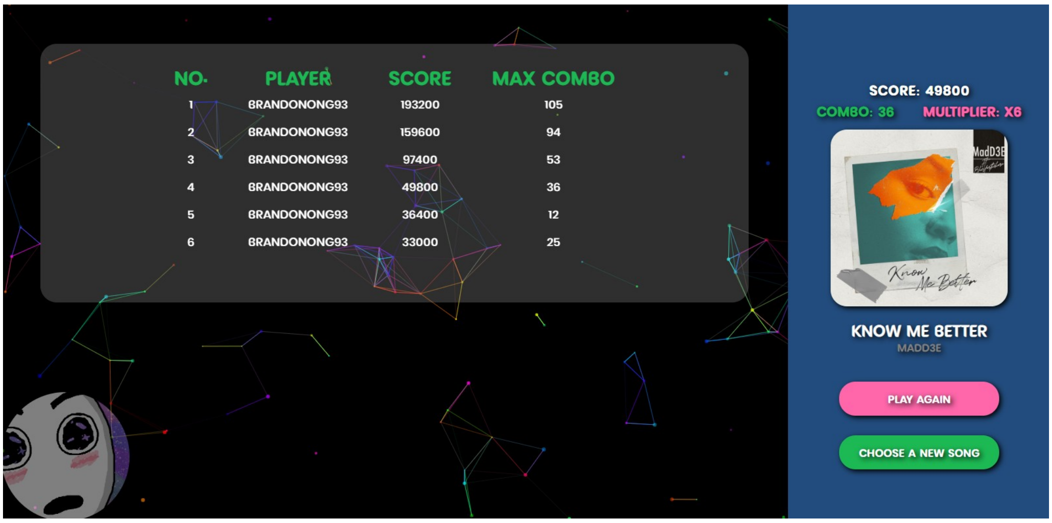
Gameplay



SCORE: 0
COMBO: 0 MULTIPLIER: X1

KNOW ME BETTER
MADD3E

High Score



NO.	PLAYER	SCORE	MAX COMBO
1	BRANDONONG93	193200	105
2	BRANDONONG93	159600	94
3	BRANDONONG93	97400	53
4	BRANDONONG93	49800	36
5	BRANDONONG93	36400	12
6	BRANDONONG93	33000	25

SCORE: 49800

COMBO: 36 MULTIPLIER: X6

KNOW ME BETTER

MADD3E

3. Server-Side Implementation

3.1 End-Point Descriptions

End-point	Response Format
GET "/auth/spotify"	Authenticates user using Passport JS
GET "/auth/spotify/callback"	Redirects to client URL

GET "/auth/profile"	{
	spotify_id: string
	photo: string
	username: string
	display_name: string
	country: string
	access_token: string
	refresh_token: string
	expires_in: integer
	updated_at: date/string
	}

GET "/auth/refresh"	{
	access_token: string
	expires_in: integer
	}

GET "/auth/logout"	Redirects to client URL

POST "/scores"	{
	player_id: string
	song: string
	artist: string
	score: integer
	max_combo: integer
	}

3.2 External APIs that will be consumed

This app will utilize Spotify's Web API to pull in information about the user's music

Endpoints used:

- GET /search
    - Allow user to search for their music and populate search results with song titles, artists, and album images
- GET /audio-features/{id}
    - Get certain data points about a selected track, tempo, time signature, energy, durations (ms), etc...
- GET /audio-analysis/{id}
    - Get more specific data points, beats, sections, segments, tatums, etc...

3.4 Authentication/Authorization and Security

User will need to login to their Spotify account before being able to utilize the application. Will utilize Passport JS to implement this.

4. Project Roadmap

Phase 1

Day	**Goal**
1 (04/10/22)	Create homepage, login screen, and initialize database in MySQL
2 (04/12/22)	Build server boilerplate and define all endpoints in express, connect endpoints to any external web api - Spotify auth
3 (04/16/22)	Create song search and search results component using the Spotify API endpoints to get necessary data
4 (04/21/22)	Build out gamplay screen and actual game mechanics - use Spotify audio analysis data to randomly generate circles according to song's beats/tempo, tapping mechanics, and scoring
5 (04/23/22)	Create a game complete page, showing user's score and options to play again, or choose a different song.
6 (04/24/22)	Create a high score page
7 (04/29/22)	Test/debug end-to-end functionality of app
8 (04/30/22)	CLASS PRESENTATIONS
9 (05/10/22)	Code Clean up, last minute testing/debugging
10 (05/12/22)	DEPLOYMENT
11 (05/14/22)	DEMO DAY

Phase 2

- Create a React Native App version of the game

- Move away from 3x3 grid and have circles randomly appear anywhere within a given play area
- Tweak the mechanism for spawning circles and make user time clicks to the beat

Phase 3

- Add customizable profiles, avatars, etc...
- Multiplayer functionality - split screen vs. opponent