

# Team 8 Project Backlog: **Kadence**

Jackson Rosenberg, Avery Schaefer, Nathan Simon, Colston Streit, Raymond Xie  
rosenbe7@purdue.edu, schae35@purdue.edu, simon70@purdue.edu,  
cstreit@purdue.edu, xie328@purdue.edu

## **Problem Statement:**

Picking music is hard. Whether you're trying to decide on a soundtrack to your workout, discover new bands in your hometown, or just pick something that fits your current mood, finding the perfect playlist is incredibly difficult with just Spotify or Apple Music. Kadence is a playlist creation app designed to bridge that gap, allowing for users to create the perfect soundtrack for whatever they're doing. Kadence will feature multiple revolutionary modes, including a fitness mode that selects songs based on the user's heart rate data from a paired smartwatch, a mood mode that generates a playlist based on how the user is feeling, and a local artist mode that finds new and undiscovered bands right in the user's backyard.

## **Background:**

### *Domain and Target Audience*

Spotify and Apple music have over 500 million users combined. Our target audience is anyone in that population who wants to elevate their music listening experience or anyone that generally enjoys discovering new music. Users who are bored of listening to the same music over and over will come to Kadence to refresh their ears. Another target audience of ours is the fitness community. Users will be able to tailor their playlist to their workout by connecting a device such as a smart watch and find music that enhances their experience in the gym. Kadence will provide a service that is supplemental to the main two streaming services in a variety of contexts that range from focused work environments to high energy cardio exercise routines. Initially our target audience will include someone with a premium subscription to one of the two music platforms, and the app will be primarily advertised to those owning a Fitbit or Garmin tracking device (as this will be our main selling-point feature).

### *Similar Systems and Their Limitations*

The only similar system that we found is called Groovifi. Groovifi interacts solely with Spotify to generate playlists based off of a song that you pick. Because of the requirement of choosing a seed song rather than general music taste / preferences, users can sometimes fall victim to the paradox of choice. Groovifi then allows you to adjust criteria such as genre and key, which is similar to our product, but is a bit more limited in total scope. One of Groovifi's most serious limitations, which we are hoping to solve, is that it does not integrate with Apple Music. We also plan on primarily focusing

on a feature that will link our app to a user's fitness watch and use live fitness data to automatically generate playlists and / or add new music to a current listening queue.

Spotify and Apple Music also have their own version of personalized playlist curation including mixes based on favorite artists and music discovery. However, the only factor changing these is user listening history, and our playlist generation app will focus on very customizable input in order to tailor to specific requests rather than curation focused on general information or habits about the user.

## **Functional Requirements:**

1. As a user, I would like to create an account.
2. As a user, I would like to login and logout of my account.
3. As a user, I would like to be able to delete my account.
4. As a user, I would like to be able to reset my password (using email for recovery if time allows).
5. As a user, I would like to view my profile.
6. As a user, I would like to have and edit a username.
7. As a user, I would like to have and display a profile picture.
8. As a user, I would like to update my profile.
9. As a user, I would like to display a short bio about myself and my music taste.
10. As a user, I would like to link a music platform.
11. As a user, I would like to be able to update my music platform.
12. As a user, I would like to be able to remove a music platform.
13. As a user, I would like to see what music platforms are currently linked.
14. As a user, I would like to create default preferences on genres, artists, etc. that affect playlist generation.
15. As a user, I would like to be able to update my default preferences.
16. As a user, I would like to connect to my fitness device.
17. As a user, I would like to be able to remove my connected device.
18. As a user, I would like to view my connected devices.
19. As a user, I would like to be able to choose a default device.
20. As a user, I would like to be able to allow / disallow the app to access my fitness data.
21. As a user, I would like to be able to rename a connected device.
22. As a user, I would like to select the genre of the playlist.
23. As a user, I would like to blacklist songs and artists from my playlists.
24. As a user, I would like to be able to create a preference between lyrical and instrumental songs.
25. As a user, I would like to choose whether explicit songs are added to the playlist or not.

26. As a user, I would like to set a lyrical language preference.
27. As a user, I would like to set a minimum and maximum song length.
28. As a user, I would like to be able to start playlist generation.
29. As a user, I would like to specify whether or not I want to save the playlist right away or allow Kadence to build a queue of songs to listen to first.
30. As a user, I would like to be able to provide feedback about the generated playlist.
31. As a user, I would like to delete a playlist after initial generation and automatically regenerate / reshuffle.
32. As a user, I would like to be able to select different playlist creation modes.
33. As a user, I would like to set / edit the short and long interval times in interval mode.
34. As a user, I would like to select ramp up or ramp down settings in the fitness mode.
35. As a user, I would like to change my default location to explore different local music scenes.
36. As a user, I would like to skip a song in a generated playlist / queue if I do not enjoy it.
37. As a user, I would like to blacklist additional songs if I strongly dislike them during playback.
38. As a user, I would like to save a playlist generated to either Spotify or Apple Music profiles after listening to it.
39. As a user, I would like to customize the name of the playlist before I save it
40. As a user, I would like to have the ability to alter the playlist (remove songs) after it has been generated (if time permits).
41. As a user, I would like to set minimum and maximum time limits on the playlist being generated.
42. As a user, I would like to view other people's profile pages.
43. As a user, I would like to hide my own profile page from other users by declaring my profile private.
44. As a user, I would like to ensure that all of my sensitive fitness information remains private from all users, including friends.
45. As a user, I would like other people to find and view my profile by declaring my profile public.
46. As a user, I would like to search for other users.
47. As a user, I would like to send friend requests to view private profiles.
48. As a user, I would like to view and listen to playlists that public users or friends generate.

49. As a user, I would like to view what song that a public user or friend is listening to in real-time (if time permits).
50. As a user, I would like to see an activity log of all of my own actions (if time permits).
51. As a user, I would like to see an activity log of new playlists my friends are generating and saving to their profiles (if time permits).
52. As a user, I would like to see statistics regarding my listening history in each mode and about my playlists (if time permits).

## **Non-Functional Requirements:**

1. As a user, I would like the app to be cross platform (iOS and Android).
2. As a developer, I would like user profile data to be safely and securely stored.
3. As a developer, I would like user health / fitness data to be safely and securely stored.
4. As a developer, I would like informative error messages to be shown both on the frontend of the app and in the developer console.
5. As a developer, I would like the app to be properly documented with an informative README.
6. As a user, I would like the app to have an intuitive and consistent user interface.
7. As a developer, I would like the repository to have a CI/CD pipeline.

### ***Usability / Performance***

Our app should have a clean and responsive UI on both iOS and Android devices to ensure that users can easily understand what our app does and how to navigate it. This includes making it clear visually whenever the app is waiting on a response from the backend so that the user doesn't become discouraged by a lack of feedback. Using React Native for cross-platform component building and reuse should make these goals easier to achieve. Finally, these response times should not be so long that the user becomes frustrated - we aim to keep them under 3 seconds wherever possible.

### ***Scalability***

Our app should at least be able to handle all of CS 40700 (students and staff) at one time, as this will allow us to feasibly test our product with a reasonably large audience if need be. The low-cost tier of Azure should definitely allow for this and should also fit within our \$200 credit. Additionally, we shouldn't need to worry about any specific number of API calls since users of our app will also be users of their chosen music or fitness platforms and we will be using their accounts to get songs or health data. Finally, we plan to write our code in a clean, sustainable way that allows for further extension (if desired) after the conclusion of the semester.

## *Security*

Given that much of our app involves using user fitness or location data as an input to our playlist generation algorithm (depending on which mode is set), it is essential that we keep this data secure. We will do this by ensuring that anytime we make a call to a third-party API, no personal data is sent without first being anonymized. Our backend server will also implement HTTPS to ensure traffic is secure from unwanted eavesdroppers. Additionally, some of the external APIs we will be using include a state parameter in requests, protecting against CSRF attacks. Internally, user profile passwords will be stored in a hashed form to prevent anybody other than the user from seeing their password, including us.

## *Hosting / Deployment*

We plan on using Azure to host our backend server and will likely use Docker or some other compartmentalization tool to simplify the process of downloading dependencies. Additionally, we plan to eventually have a working CI/CD pipeline where our code is only deployed if appropriate tests are passed. The front end of our app will be deployed as an app for iOS and Android. For now, we will not be including a web version.