

Team: Dewbed

Members: Emily and Devika

Project: 1B - Research

Primary Authors: Bettina and William

---

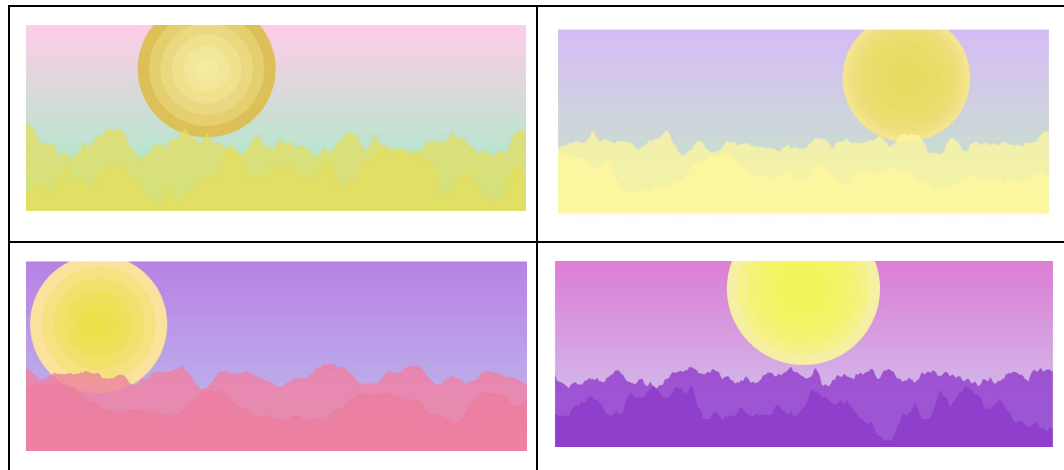
## P1B - Competitive Research and Positioning

**Determining Competitive Products:** We first researched the products that inspired our software (Discord, Spotify, and Zoom) and took note of the features that the software already provided. We also researched the apps highlighted on [Spotify's Developer's Showcase](#) that had significant similarities to our product. Lastly, to find similar products on the Discord side of our product, we researched the most popular Discord Music bots.

### Competitive Products:

- [C-Listening Room](#)
  - **Description:** A website that allows you to listen to Spotify music with friends. All listeners can add to the queue and the highest “voted” song will play next. When the queue runs out, songs in similar genres will play. A robot can also play music with the user to suggest songs as well.
  - **Key Strengths:** Core functionality worked very well, seamless integration between the web app and Spotify.
  - **Key Weaknesses:** No option to log out/revoke access, the UI is basic and unintuitive.
  - **What they can improve on:** Needs better privacy (You can see the name of all the users who are online, so if a user is connected to Facebook through Spotify, you can see their full name.)
  - **Key Differences (from our plan):** Does not have a discord integration nor does it analyze music taste. There is no text on how the robot suggests music and it seems a bit random. No analysis feature that converts music analysis into the playlist cover art.
  - **Key Implementation Takeaways:** Uses Spotify Web API, [Github code](#), hosted on Heroku, built primarily using JavaScript.
  - **Payment Type:** No payment needed!
  - **User Feedback:** Not public/available
- [Musicape](#)

- **Description:** Generates a landscape image based on recently played songs in Spotify. Provides an explanation of the image based on music theory concepts.
- **Key Strengths:** Good analysis of music key, beat, and general emotion. Creates an aesthetic illustration.
- **Key Weaknesses:** is a simple, standalone feature.
- **What they can improve on:** Make the image square so it can be used as a playlist cover. Could easily be a feature in a more comprehensive music analysis application.
- **Key Differences (from our plan):** No feature to allow users to listen to music together. Does not suggest new songs or new playlists. No analysis feature that converts music analysis into the playlist cover art.
- **Key Implementation Takeaways:** Uses Spotify Web API, [Github code](#), hosted on Heroku, built using a mix of HTML, CSS, and JavaScript.
- **Payment Type:** No payment needed!
- **User Feedback:** Not public/available



The background of your musicScape is **pink** because you've recently listened to music that relates to **happy or cheerful** emotions.

It's **daytime** in your musicScape because your recent music is mostly in **major** mode.

The mountains are more jagged because you've listened to **more energetic** songs.

You've been a **less active listener** during the past 24 hours, so your musicScape has one pair of mountains.

The mountains are tones of **bright yellow** because your songs are mostly in the **key C#/D b**.

- [JQBX](#)

- **Description:** An IOS, Android, and Web App where you can share and listen to music live time with friends. You can DJ the music or just listen along. Users can also rate songs and ask to skip.
- **Key Strengths:** You can select different DJs who can easily select songs and create a listening session. Users can thumbs up/down the music to give feedback to the DJ. Android, IOS, and Web App.

- **Key Weaknesses:** Slight Lags, creates two playlists on a user's Spotify without notifying them, which can be quite confusing.
- **What they can improve on:** A more approachable user guide to help new users acclimate to the new environment. Does not suggest new songs.
- **Key Differences (from our plan):** No analysis of the music and playlists created. More importantly, it's more of a listening feature rather than connecting through a chat service and listening. No analysis feature that converts music analysis into the playlist cover art.
- **Article about JQBX for reference:** [Tech Hive](#), [JQBX Product Comparisons](#)
- **Key Implementation Takeaways:** Has both a web app and a downloadable application, is compatible with with most platforms (Windows, Mac, Android & IOS),
- **Payment Type:** Free to use (with Spotify Premium)
- **User Feedback:** 4/5 stars on [appgrooves](#), users enjoy the collaborative music streaming and chat room features, some find the user interface to be confusing (they released a user guide to help).
- Zoom
  - **Description:** Video conference software that allows one person to share their computer audio
  - **Key Strengths:** Everyone in the same meeting can listen to the same audio as someone in the call (whoever is sharing their audio)
  - **Key Weaknesses:** Only one person can share their audio and control so that only one person chooses the music. They also only allow a person to share their whole computer audio so any other sound played on the person's computer
  - **What they can improve on:** Allow the user to only share sound from one application, such as Spotify, instead of sharing all computer audio.
  - **Key Differences (from our plan):** No analysis of the music and playlists created. More importantly, it's more of a video conferencing app. You cannot connect your Spotify nor have listener input on the current song other than the chat feature/ speaking up on a call. No analysis feature that converts music analysis into the playlist cover art
  - **Key Implementation Takeaways:** Zoom features both a web app and a downloadable application.
  - **Payment Type:** Zoom requires a premium account for more than 40 minutes on a video call.
  - **User Feedback:** Interface for sharing video/audio is overwhelming. Default for meetings is to not let participants screen share, which is inconvenient.
- Discord Rhythm Bot
  - **Description:** A bot which you can invite to your server and connect to a voice channel to listen to music together.
  - **Key Strengths:** It is easy to search for songs through youtube. You can also easily interact with the queue and ask the bot what is currently playing. No additional hardware or software needed to invite the bot as well.

- **Key Weaknesses:** the songs recommended are based on song titles on Youtube. If the youtube video is incorrect, whatever audio will play. There can be ads or a delay depending on Youtube. The bot goes down often and can glitch. Spotify premium is required for users that wish to integrate spotify with discord.
- **What they can improve on:** Include a user guide as soon as the bot is introduced into the server. More reliable access to YouTube or access to other services rather than YouTube
- **Key Differences (from our plan):** No analysis of the music and playlists created. Only connects through YouTube and plays music through the Youtube API. There is no connection to Spotify nor does it remember your listening sessions as well . No analysis feature that converts music analysis into the playlist cover art.
- **Key Implementation Takeaways:** <https://github.com/Malexion/Rhythm-Bot>, built using typescript, connects to Twitch in addition to Youtube and Soundcloud.
- **Payment Type:** No payment needed!
- **User Feedback:** <https://rythmbot.co/reviews> highlights some of great reviews by Rhythm users: “Rythm is by far the best music bot I've used on Discord and you all continue to add features, listen to feedback, and generally keep moving forward with creating a great bot!” Negative reviews are not featured on their website.
- [Discord Groovy Bot](#)
  - **Description:** Similar to Rhythm bot, Groovy can be invited to your server and connect to a voice channel to listen to music together. Unlike Rhythm, Groovy can support Spotify, Youtube, and Soundcloud. To play from Spotify, Groovy takes in a link input (for example, a playlist link).
  - **Key Strengths:** Allows anyone in the voice channel to pause, skip, or add to the queue. No lag.
  - **Key Weaknesses:** Does not recommend songs. When the queue ends, Groovy does not play anything. Include a user guide as soon as the bot is introduced into the server. You can not search songs from spotify.
  - **What they can improve on:** Lacks features that allow users to find new songs.
  - **Key Differences (from our plan):** No analysis of the music and playlists created. Does not remember your listening sessions as well. No analysis feature that converts music analysis into the playlist cover art.
  - **Key Implementation Takeaways:** <https://github.com/groovy-bot>, can have multiple bots in a server, has a spotify feature.
  - **Payment Type:** Premium is available with additional features (volume control, audio effects, etc.) Premium requires payment. \$3.99 per 1 server. Payment size increases depending on how many Discord servers the Bot is added to.
  - **User Feedback:** “Groovy is a great option due to its uptime and ease of use.” [online-tech-tips](#)
- Discord

- **Description:** Similar to Slack, a chat like service with servers for gamers. Users can create channels and voice chats to chat and share with friends. [Spotify has an integration within Discord](#) where it shares what a specific user is listening to. Other users can “listen” along on Spotify through their own accounts. Another feature includes the feature to host listening parties in chatting channels.
- **Key Strengths:** Wonderful UI that shows the album, artist, song name, and time in song. Easy to click and listen along, easy to share listening parties. You can also see what friends are listening to and explore those songs as well. Integrated within Discord, no extra software is needed other than logging in to Spotify.
- **Key Weaknesses:** Does not notify a user when someone else decides to listen to their music session, which sometimes does scare users. It also is a few seconds behind the host as well. For listening parties, if the host pauses or stops the session, it ends the session completely. Another issue is the “Listen Along” button can get spammed out easily in general text servers.
- **What they can improve on:** Notify users when someone joins in on their private Discord sessions. Easier access to larger listening parties where the session does not end abruptly due to the host pausing. There are no instructions on how to connect Spotify to your Discord, which leaves new users confused.
- **Key Differences (from our plan):** User who’s spotify is linked is the only user allowed to control the music from the queue, pausing, skipping, etc. Does not use music theory to analyze the playlists created.
- **Key Implementation Takeaways:** Connects with Spotify API to grab user data. There is also an official partnership between the two companies as well so there must be some shared implementation.
- **Payment Type:** All users will need Spotify Premium to either listen along, invite, and show what they are currently listening to.
- **User Feedback:** Read Reddit [User Reviews here](#). Reviewers do not like the in-game status. Online Tech Tips states, “If there’s one thing that Discord does better than every other chatting platform out today, it’s bots... One of the most popular and fun types of Discord bots are music bots.”
- Spotify Radio
  - **Description:** Spotify has an optional feature that generates recommended songs based on a song, playlist, or artist. The user can thumbs up or thumbs down song choices. Spotify will avoid future playing songs that are thumbs down in the future. Songs that are in thumbs up will be saved into a separate “Liked Songs Playlist”. Songs that are recommended tend to be songs that are already saved.
  - **Key Strengths:** Great way to find songs that are from a similar genre or songs from artists that have similar audiences.
  - **Key Weaknesses:** Spotify will play songs based on the playlist title, which is sometimes not relevant to the actual songs in the playlist. Can not save songs without it affecting the radio suggestions.
  - **What they can improve on:** Difficult to find songs from new artists. Spotify doesn’t provide explanations for their song recommendations. Only one person

- **Key Differences (from our plan):** Not collaborative, doesn't show music theory analytics, focuses on the playlist and not the user
- **Key Implementation Takeaways:** Rating songs to provide feedback, producing similar music for the user to enjoy.
- **Payment Type:** Included in the free version of Spotify.
- **User Feedback:** [Article about Spotify Stations](#), [Article comparing it to Pandora](#), users enjoy the lightweight interface, they wish you could see the radio history.
- Spotify Liveshare
  - **Description:** Spotify has a feature that allows users to invite up to 5 people to join a listening party. Participants listen to the same music and can add songs to the collective queue
  - **Key Strengths:** Inviting is through sharing a link and is straightforward. Users can share their listening experience with others
  - **Key Weaknesses:** Feature only on mobile, participants often disconnect unexpectedly. Playback is not necessarily synchronized among different users.
  - **What they can improve on:** integrate different platforms, improve the stability of the build
  - **Key Differences (from our plan):** Does not incorporate any music theory aspects into the shared experience.
  - **Key Implementation Takeaways:** Spotify Liveshare is a fairly new feature that must use implementations from collaborative playlists and Discord integration as well.
  - **Payment Type:** Only on mobile app, included in the free version of Spotify.
  - **User Feedback:** [Article with user reviews as comments](#), users prefer other apps like JQBX currently as the feature is still a work in progress.
- Spotify Collaborative Playlist
  - **Description:** Spotify has a feature that allows users to create playlists that they can share with friends. If they follow the playlist they are able to add songs to the playlist.
  - **Key Strengths:** People can add their own songs to build the playlist and it is easy to use. It also shows who added what songs to the playlist.
  - **Key Weaknesses:** It's only a playlist; it does not allow people to play this song synchronously.
  - **What they can improve on:** Adding songs and inviting people can sometimes be a bit hard. Maybe improving that UI?
  - **Key Differences (from our plan):** Focus is on just the playlist so they don't have a live feature. Does not incorporate music theory analysis of playlists. No analysis feature that converts music analysis into the playlist cover art.
  - **Key Implementation Takeaways:** Must allow all users to add playlist songs but also delete playlist songs, therefore all users are technically "authors" of the playlist.
  - **Payment Type:** Included in the free version of Spotify.

- **User Feedback:** N/A as it has been a part of Spotify since 2008, it is no longer a new feature but rather a foundational, expected to work feature. [Reddit thread regarding how users interact with collaborative playlists](#)