

Noted

Where story meets soundtrack

Alyssa Frederick | Kabir Jolly | Jasmine Vonk | Caleb Liu

[Link to Med-Fi Prototype:](#)

<https://www.figma.com/file/Am6VPJDIdEcaP3zb4RYTHe/cs147-med-fi?type=design&node-id=0-1&mode=design&t=baUCE8NqateRlyPz-0>

Target Audience

Our target audience was drawn from two significant trends we identified through comprehensive interviews: Firstly, we learned that numerous journalists have the desire to incorporate photos into their entries but lack a simple way to do so. Secondly, we serendipitously discovered that many of our interviewees listened to a lot of music and believed music had a very deep and associative power to capture one's past. Our target audience emerges from the intersection of these insights: *people who enjoy looking at photos and listening to music that desire an organized way to chronicle and reflect on past experiences.*

Design Tools

Figma was our primary design and wireframing tools for the medium fidelity prototype. Our solution is designed for a cross-platform mobile application, but our design dimensions are of an iPhone 14. In the early stages, we used *Notability* to sketch screen layouts, logos, and icons.

Operating Instructions

This section details the screens involved in each task flow and describes the interactions.

Throughout the interface are buttons that lead the user through our three task flows as well as general navigation through the prototype. Clicking on any space in the interface will show a bounding box which indicates that the interaction is valid in the mobile app.

Navigation Bar (bottom of screen):

- Reflection: Tap the leftmost journal icon to open the reflection screen displaying your memories.
- Friends: Tap the friends icon to open the friends screen, showing your friend's photo song pairings.
- Compose: Tap the plus icon to see the compose section, where users can create a photo/song pairing
- Messages: Tap the messages icon to send photo/song pairings to friends.
- Profile: Tap the profile icon to view your own profile which shows your personal timeline of photo/song pairings.

Reflection:

1. Featured Memory:
 - A random photo/song pairing from your past is displayed at the top of the reflection screen. Below the pairing is the date and description.
2. Browse Collections:
 - Noted curates collections of photo/song pairings, grouping them by mood/genre of the music. In this prototype, these collections are hard coded in.
3. Add a Text/Journal Entry:
 - Click on a photo/song pairing to open an expanded view of it. Click the "share more" button to open a text-input box where users can add their description to the pairing.

Friends:

1. Vertical Feed:
 - Friend's photo song pairings are formatted in a vertical scrolling feed.
2. Manage your ensemble (friends):
 - a. Click the top right icon of a friend with a pencil to open up a list of your current ensemble (friends). Click the three dots to the right of a friend's name to adjust how frequently they come up in randomly generated photos (in the compose screen). There are two options: "see more" and "see less" - clicking either one will adjust the likelihood of them appearing in randomly generated photos. (We will use facial recognition AI to do this).

Compose:

1. Photo Selection

- Initially prompted with a random photo from your library
 - Optionally regenerate a random new photo from your library
 - Press the top right picture icon to choose your own photo from your library
2. Song Selection
 - [not sure if implemented in fig] Press the search bar and type in a song to search for a song from Spotify's catalog
 - Horizontally scroll through your top songs (provided by Spotify). Click the CD to choose the song. (In the prototype, you must choose the third CD from the left)
 3. Description (optional)
 - a. Tap on the gray text input box to add a caption/description for the pairing.
 - b. Toggle the visibility by pressing the toggle button in between private and public
 - c. Hit the "done" button to share.

Messages:

1. Friend Selection
 - A vertical list of friends is displayed, organized by recency (newest at the top). Press a friend row to enter a direct message screen with that friend.
2. Send a pairing
 - Press the CD Icon to the left of the text input to display the user's photo/song pairings. Press a single photo/song pairing to place it in the message box. Press the send icon on the right of the text input to send the message.

Profile:

Limitations

Here, we describe the limitations for our prototype.

There is no finality for a search bar in the song selection screen in the compose section because it would require implementation of Spotify's API which cannot be done in Figma.

When there are multiple clickable components of the same type (ex: CDs of songs), we only support clickable functionality for one of them. Because the interactive behavior/linking would be similar to these components, we believe

that as long as there is one interactive element, the user would be able to get a realistic user experience for our app.

Users may not input text in the prototype because Figma does not support typed user input and rendering of it. To mitigate this limitation, we place filler text when a text-input is pressed to indicate that pressing a text input leads to text being inputted. (see more info in wizard of oz section)

Furthermore, users cannot play music when pressing on the CD within the photo-song pairing Figma component. This is because we are not able to connect the Spotify API like we plan to do for the app, so there is no audible component to the medium-fidelity prototype.

Wizard of Oz Techniques

Here, we describe the Wizard of Oz techniques we simulated during prototyping and our rationale.

The user receives a list of their top tracks based on their listening history, which has been hard coded in.

- Rationale: It would be difficult to conditionally render a list of personalized information. Not to mention, there is no Spotify API integration with Figma.

We fill in descriptions with default when a user clicks a text input attempting to add a description to a photo/song pairing.

- Rationale: As mentioned before, this is because the prototype cannot conditionally render based on user-inputted tasks.

Hard-Coded Items

Here, we describe the static data used in the prototype and our rationale.

Our Wizard of Oz features from above are hard-coded. .

- Rationale: Due to the limited functionality of Figma and lack of a functioning database our prototype cannot conditionally render based on user data, text input, Spotify data, etc.

The photo/song pairings displayed in the profile page and feed are hard-coded. In high-fidelity, these pairings would be user-created.

- Rationale: Figma cannot make calls to a database that stores user information

Profile information for users and friend profiles are hard coded

- Rationale: Figma cannot make calls to a database that stores user information