

User Study: Functional Prototype Evaluation

Revised Requirements

Playlist Management: As a user, I want advanced playlist customization options to rearrange and edit my playlists in real-time.

Music Browsing and Playback:

- **As a user,** I want an intuitive interface to easily navigate the app and access my playlists without confusion.
- **As a frequent traveler,** I want seamless offline listening capabilities to enjoy music without Wi-Fi or mobile data.
- **As a user,** I want the app to automatically scan and display all audio files from my device storage, so I don't have to manually add songs.
- **As a user,** I want the app to be able to update the metadata of my music files which may have some details such as artists missing.

No Autoplay: As a user, I want the app not to autoplay music when I open it or connect to a Bluetooth device, so that I have full control over when my music starts playing and is not interrupted unexpectedly.

Revised Personas

PERSONA 1 - Alice Smith 24 years old female, UIC graduate student who enjoys a huge variety of unexplored music usually not found in online streaming applications.	PERSONA 2 - Sam Williams, 35 years old Asian male, graphic designer who prefers an offline music player for focused, uninterrupted listening and values simplicity and organization in his music library.												
BIO <i>Alice is a commuter student who loves collecting rare music from various sources. She downloads tracks from niche websites and archives but often loses track of where they are saved on her device. This makes it frustrating to find and play her music, especially during commutes. Alice values simplicity and needs a tool to organize her music efficiently, allowing her to enjoy her favorite songs without the hassle of searching through cluttered files.</i>	BIO <i>Mark is a freelance graphic designer who often works in cafes and co-working spaces. He relies on an offline music player to help him focus during work sessions, preferring curated playlists or full albums without interruptions. His music library is stored on his phone, as he dislikes streaming services due to data usage and inconsistent internet connections.</i>												
FEATURE PREFERENCES <table border="1"><thead><tr><th>Music Exploration</th><th>Favorites</th></tr></thead><tbody><tr><td>Collaborative</td><td>Solo</td></tr><tr><td>Organisation</td><td>Flexibility</td></tr></tbody></table>	Music Exploration	Favorites	Collaborative	Solo	Organisation	Flexibility	FEATURE PREFERENCES <table border="1"><thead><tr><th>Music Exploration</th><th>Favorites</th></tr></thead><tbody><tr><td>Collaborative</td><td>Solo</td></tr><tr><td>Organization</td><td>Flexibility</td></tr></tbody></table>	Music Exploration	Favorites	Collaborative	Solo	Organization	Flexibility
Music Exploration	Favorites												
Collaborative	Solo												
Organisation	Flexibility												
Music Exploration	Favorites												
Collaborative	Solo												
Organization	Flexibility												

Revised Use Case / Scenarios

Scenario 1

Alice, a commuter student, often finds herself struggling to locate her downloaded music files on her phone. On this particular day, as she boards the bus to school, she decides to listen to some music to make the journey more enjoyable.

She opens Tapify on her phone and is immediately greeted by a clean, organized interface that displays her music library of pop songs. She sees her Taylor Swift and Ed Sheeran playlist that she plays every morning, instead of sifting through folders or searching for her tracks. Alice has downloaded Taylor Swift's new album and when she taps on the "Recently Downloaded" category in Tapify. Instantly, she sees a list of her latest music downloads, organized by date.

She spots her track - "Yellow by Coldplay" and taps play. Without any delays or frustration, her music begins playing through her headphones as she relaxes into her commute.

Alice smiles, appreciating how simple and hassle-free the process was, allowing her to enjoy her music exactly when she needed it.

Scenario 2

Mark, a freelance graphic designer, is working at a local café on a tight deadline. The noise around him is distracting, so he decides to play his favorite instrumental music to help him concentrate.

He opens his offline music player and taps on his "Focus" playlist, but notices the new track - "Country Roads" by the local band Ritviz is misspelled as "County Road". He clicks on the edit option available at the right side of the phone screen and a screen with all the song's metadata pops up. Mark quickly renames the track, making it easier for him to find the track in the future.

Once the name is updated, he taps play, and the instrumental tracks begin to flow seamlessly. With the music in the background, Mark feels more focused and productive as he works. When his work session ends, Mark easily closes the app, satisfied with both the customization and how smoothly his music experience supported his workflow.

Implications for Design

Clear and Step-by-Step Navigation

Do: Create a simple navigation flow where users can access features like “Search,” “Playlists,” and “Recently Added” with minimal steps.

To Meet: Alice’s need to quickly find and play music while commuting without searching through files.

Editable Song Metadata

Do: Add an option to edit song names, artists, and album details within the app.

To Meet: Mark’s need to organize music files and ensure naming consistency across tracks.

Custom Playlist Creation

Do: Allow users to create and save personalized playlists directly from the app.

To Meet: Both Alice’s and Mark’s need for customized music experiences based on their moods or activities.

Offline Playback Optimization

Do: Ensure music loads instantly and functions seamlessly in offline mode, regardless of file size.

To Meet: Alice’s need for uninterrupted music during her commute and Mark’s need for a reliable offline player at work.

Powerful Search with Filtering

Do: Implement a robust search bar with filters like “By Artist,” “By Album,” and “Recently Played.”

To Meet: Alice’s need to quickly locate tracks in a large, disorganized music library

Specification: Information Design

Music Library (Collection of Songs):

- **Facets:** Title, Artist, Album, Genre, File Location, Date Added, Duration.
- **Relation to Design Model:** A searchable and sortable music library mirrors the user's mental model of managing a physical music collection. It ensures quick retrieval of tracks.
- **Presentation:** Songs are displayed in a list format with prominent facets like title and artist. Sorting and filtering options (e.g., by genre or date added) make navigation straightforward.

Playlists:

- **Facets:** Playlist Name, Tracks, Creation Date, Duration.
- **Relation to Design Model:** Playlists are user-curated collections of songs, providing a personalized experience. They enhance engagement by enabling users to create mood-based or event-specific sets.
- **Presentation:** Playlists are shown as cards or in a dedicated tab, with customizable options like renaming and reordering tracks.

Editable Song Metadata:

- **Facets:** Title, Artist, Album, Genre, Year.
- **Example Values:**
 - *Before:* "Track_01.mp3"
 - *After:* "Thunderstruck" (Title), "AC/DC" (Artist).
- **Relation to Design Model:** Aligns with Mark's need to organize music files and Alice's desire for consistency in her library.
- **Presentation:** Users can edit metadata through a straightforward dialog box or overlay on the song details page.

Search and Filters:

- **Facets:** Search Query (Artist, Album, Song Name).
- **Relation to Design Model:** Provides a focused and powerful way to narrow down extensive libraries, supporting Alice's need to find tracks quickly.
- **Presentation:** A search bar with dropdown filters for facets, displaying live results as the user types.

Revised Functional Prototype

Current figma design - (Based question route using the improved figma prototype - due to issues faced in development)

Figma prototype link - [Figma Prototype](#)

Our previous prototype included - 3 main features

- Extracting songs from all storage media within device
- Music playback
- Song browsing

Link to github repo with functional prototype 1 - [Tapify V1.0](#) (will require mp3 files to run locally)

Link to video recording of first version: [Tapify V1.0 Demo](#)

Our current prototype updated the look and feel of the project,

- All files are being scanned and shown in the listview.
- A global search for artists and song names has been added
- The user can update the metadata (song name, artist, album) of the song files within the app for better understanding
- Music playback and volume control
- Shuffle option for music tracks

Link to github repo with current prototype - [Tapify](#) (will require mp3 files to run locally)

Link to video recording of current version: [Tapify V2.0 Demo](#)

Methods - Participants

	Age	Gender	Race
<i>Participant 1</i>	24	Male	White
<i>Participant 2</i>	28	Male	Asian
<i>Participant 3</i>	22	Female	Asian
<i>Participant 4</i>	27	Male	White
<i>Participant 5</i>	25	Male	White
<i>Participant 6</i>	25	Female	Asian
<i>Participant 7</i>	35	Male	White
<i>Participant 8</i>	38	Female	White

<i>Participant 9</i>	25	Female	Asian
Participant 10	32	Female	Asian
Participant 11	22	Female	Asian
Participant 12	24	Male	Asian

Method - Consent Form

Google form link to the consent form used by our group - [Consent Form for Music Application Usage Study](#)

Methods - Question Route

- How did you decide which song to play first?
- What steps did you take to locate a specific artist or track?
- How did the app help you keep track of your music collection?
- What was your approach to editing metadata for songs you wanted to update?
- If you needed to quickly create a playlist for a specific mood or event, how would you approach that using this app?
- When exploring shuffle mode, how did it impact your listening experience during a long session?
- If you needed to fix a mistake—like an incorrect metadata update—how did you go about it?

Findings - Qualitative Data analysis: coding

Focused Enhancements and Features:

1. User Interface and Experience:
 - Phenomena Addressed: Affordances, visibility, and feedback.
 - Key Features:
 - Intuitive navigation for seamless interaction.
 - Enhanced visual hierarchy for easier content discovery.
 - Observation: Cleaner UI encourages engagement and helps users make sense of available options.
2. ListView for Scanned Files:
 - Phenomena Addressed: Visibility and sensemaking.
 - Key Features:
 - Organized display for all media files.
 - Simplifies browsing through the library.
 - Observation: Users easily locate and access desired files.

3. Global Search for Artists and Song Names:
 - Phenomena Addressed: User needs and conceptual models.
 - Key Features:
 - Advanced search functionality for precision.
 - Observation: Addresses a common user need to quickly locate content.
4. Metadata Editing:
 - Phenomena Addressed: Annotation and user desires.
 - Key Features:
 - In-app editing of song information like title, artist, and album.
 - Observation: Empowers users to personalize their library and maintain organization.

Enhanced Playback Features:

5. Playback with Volume Control:
 - Phenomena Addressed: Feedback and mapping.
 - Key Features:
 - Precise volume adjustments.
 - Observation: Users have better control over audio output, enhancing satisfaction.
6. Shuffle Option:
 - Phenomena Addressed: Game mechanics and user desires.
 - Key Features:
 - Randomized playback for variety.
 - Observation: Provides an element of surprise, keeping users engaged.

User Behavior Insights:

- Common Behaviors:
 - Browsing via ListView for organized access.
 - Using search to locate specific tracks or artists.
 - Editing metadata to maintain a clean library.
- Rare Behaviors:
 - Experimenting with shuffle mode.
 - Adjusting playback volume for nuanced listening environments.

New Implications for Design

Enhance Metadata Customization Options

The ability to edit metadata empowers users to personalize their music library. Expanding customization features—such as bulk editing, auto-suggestions for metadata, or integration with online databases for accurate tagging—to allow users to efficiently organize their collections.

Simplify Playlist Creation and Management

Users value the ability to curate and organize their music into playlists for specific moods, events, or personal preferences. The app should provide tools for creating, editing, and organizing playlists, such as drag-and-drop song addition, and the ability to share playlists.

Support Smart and Dynamic Playlists

Introducing features like dynamic playlists (e.g., auto-updating based on criteria such as genre, artist, or user-defined tags) caters to users who prefer automation over manual organization.

Sharing Options

Adding a “Share” option to song-specific menus, accessible through a dropdown in the ListView. Allow users to share songs directly to messaging apps (e.g., WhatsApp, iMessage) and social platforms (e.g., Instagram, Twitter).