

USER STUDY LOW FIDELITY

Revised Use Case / Scenarios, Personas, Requirements (Specification) - 10 points

Revised Personas

These personas are derived from an amalgamation of user interviews. Through our research, we identified recurring expectations and frustrations among users. Our interviews reached data and thematic saturation, enabling us to create these personas which represent the expectations.

Persona 1 is based on a commuter student who uses local storage music due to connectivity issues during traveling.



FRUSTRATIONS

Playlist organisation is complicated (by artists, unknown, albums)

Playlist modification when playing music is not intuitive enough.

Inconsistent landing page of music player

FEATURE PREFERENCES



PERSONA 1 - Alice Smith 24 years old female, UIC graduate student, uses music player for commute at least 2-3 hours per day

BIO

Alice is a commuter student who travels approximately 3 hours a day. She generally listens to music when travelling. She enjoys a variety of music genres and has a playlist curated for each one. She prefers minimalism – an easy-to-use app that allows her to quickly access her favorite music without distractions. Alice often feels lost when navigating through cluttered menus and features she doesn't use. She prefers curated, mood-based playlists. She uses music player apps over online streaming applications due to network connectivity issues.

GOALS

Playlist organisation is complicated

Playlist modification when playing music is not intuitive enough.

Inconsistent landing page of music player

Persona 2 is based on a user who prefers curating their own music as they have a niche music taste. They generally download music and when sharing it they generally share some part of the metadata like song title and artist/album.



FRUSTRATIONS
Inter playlist navigation is not convenient. Takes you back to home screen before you can go back to the playlist selection
Unable to link a unknown song with artist manually
Sharing features share a lot of unnecessary info in the form of a m3u8 file

GOALS
Allow user to add metadata to their music
Minimize the number of clicks or taps needed to move from one playlist to another.
Sharing songs should be in a text format

FEATURE PREFERENCES

Music Exploration	Favorites
Collaborative	Solo
Organisation	Flexibility

Persona 2 - Sam Williams, 28 year old male, Product designer, uses music apps generally when working, enjoys a huge variety of unexplored music usually not found in online streaming application

BIO

Sam is a freelance product designer who works from his home studio in Vancouver. He's always been passionate about discovering new, underground music, particularly from independent or lesser-known artists. Since much of his favorite music isn't available on mainstream streaming services, Sam has built up a large collection of locally stored files on his laptop and phone. Music is central to his creative process, and he has specific playlists tailored to different projects and moods—ambient tracks for focused work, upbeat indie for brainstorming, and mellow instrumentals for winding down.

His workday flows best when she's in her creative zone, uninterrupted. He often switches between different playlists based on the task at hand, he prefers apps that are easy to navigate and responsive, allowing him to shift gears without losing focus. Same values intuitive, minimalist design in the tools he uses. When apps have complicated navigation or force him to take unnecessary steps—like going back to the home screen to switch playlists—he finds it disruptive to his workflow.

Revised Use Case / Scenarios

Scenario 1

Alice has a long commute every day to college, and she relies on music to make the journey more enjoyable. Her current music app, however, leaves her frustrated. The app's cluttered and complicated interface makes it difficult to find her offline playlists, and every time she opens it, she's greeted with recommendations that don't work without a network connection, adding to her frustration.

One day, Alice decides to try Tapify, an offline music player app she hopes will make her commute smoother. When she opens Tapify, she's greeted by a clean, minimalistic interface with her curated playlists right on the landing page. With one tap, she starts her "Relax" playlist without needing to navigate through irrelevant tabs or network-dependent recommendations.

As Alice enjoys her music, she wants to reorder a few tracks to better match her mood. Tapify's drag-and-drop feature makes it easy for her to adjust her playlist without interrupting playback, giving her full control over her listening experience. Since Tapify is fully offline, Alice doesn't have to worry about network connectivity issues disrupting her music. She can enjoy a seamless, stress-free listening

experience that adapts to her preferences and commute needs. Tapify allows Alice to enjoy her favorite music without distractions or connectivity issues.

Scenario 2

Sam is a freelance product designer working from his home studio in Vancouver. As he dives into a new design project, he likes having the right music to set the mood—usually something ambient for deep focus or indie tracks to get his creative energy flowing. But switching between his playlists is a hassle on his current music app; every time he wants to change the vibe, he has to go back to the home screen, which interrupts his flow.

Looking for something better, Sam decides to try Tapify. When he opens the app, he immediately notices how easy it is to move between playlists. With just a couple of taps, he can jump from his “Focus” playlist to his “Brainstorm” playlist without any unnecessary steps or interruptions. This streamlined navigation fits perfectly with his creative process, letting him switch moods seamlessly.

While listening, Sam remembers he wants to add metadata to some of his lesser-known tracks, like artist info and mood tags. Tapify’s simple tagging feature allows him to do this effortlessly, making it easier to organize and find these tracks later. Sharing his music with colleagues is also a breeze; Tapify lets him send a text-based link to a song or playlist, avoiding the messy file formats he dealt with before.

With Tapify, Sam’s music setup finally matches his workflow. He can shift from one playlist to another easily, add custom metadata to keep his library organized, and share music without hassle. Tapify’s intuitive design keeps his focus on his work, not on navigating the app.

Revised Requirements

Functional Requirements

- **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.

- **Playlist Management:**
 - As a user, I want advanced playlist customization options to rearrange and edit my playlists in real-time.
 - As a playlist enthusiast, I want to be able to customize my playlists with new album covers and personalized themes for different song backgrounds
- **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.

Non-Functional Requirements

1. **Performance:** The app must load music and playlists within 1-2 seconds to maintain a fast user experience. Offline music should be accessible immediately when there is no internet connection.
2. **Usability:** The app should have a minimalist and intuitive interface with no more than three clicks to access any feature (e.g., recently played, playlists, or artist selection). The design should accommodate users across a wide age range.
3. **Scalability:** The backend should handle thousands of users without affecting performance, even with complex operations like playlist customization and shared playlist collaboration.
4. **Reliability:** The app must maintain uptime of 99.9% during music playback and provide fallback options (e.g., switching to offline music) in case of server issues.
5. **Security:** User data, including playlists and music preferences, should be stored securely with encryption.
6. **Platform Compatibility:** The app should be optimized for Android with support for multiple versions and device types.

Project Constraints

- **API Usage Limitations:** APIs for streaming services have a limited number of calls allowed per day, which is a constraint when scaling up for user testing or data retrieval.
- **Offline Access:** Ensuring offline functionality might require managing local storage effectively without violating streaming platform policies.
- **Resource Limitations:** Limited resources (time, budget, workforce) to complete user testing, design iterations, and development phases.
- **Platform Limitations:** Compatibility issues across different Android versions or devices.

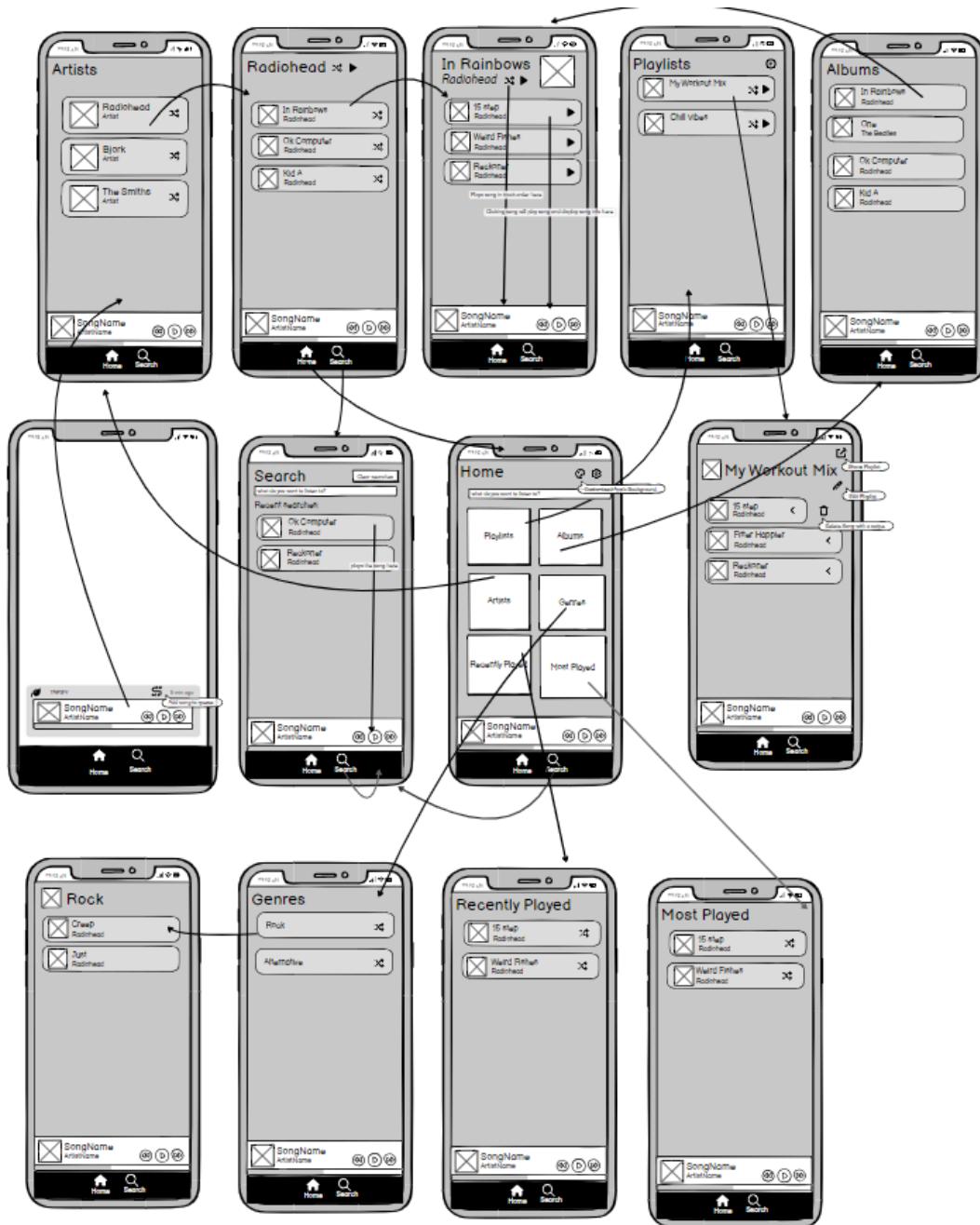
Project Issues

1. API Constraints for Testing: Due to daily API call limitations, it is difficult to test real-world user scenarios with streaming APIs. Proposed Solution: Focus on local storage music for user testing with at least 10 users to bypass API limitations.
2. Offline Playback: Ensuring seamless offline playback with locally stored music without violating licensing or service agreements.
3. User Feedback: Need to gather meaningful feedback from diverse users (commuters, students, fitness enthusiasts, etc.) to optimize the app's various modes.
4. Data Management: Handling shared playlists and recommendations without excessive data usage or storage overload.

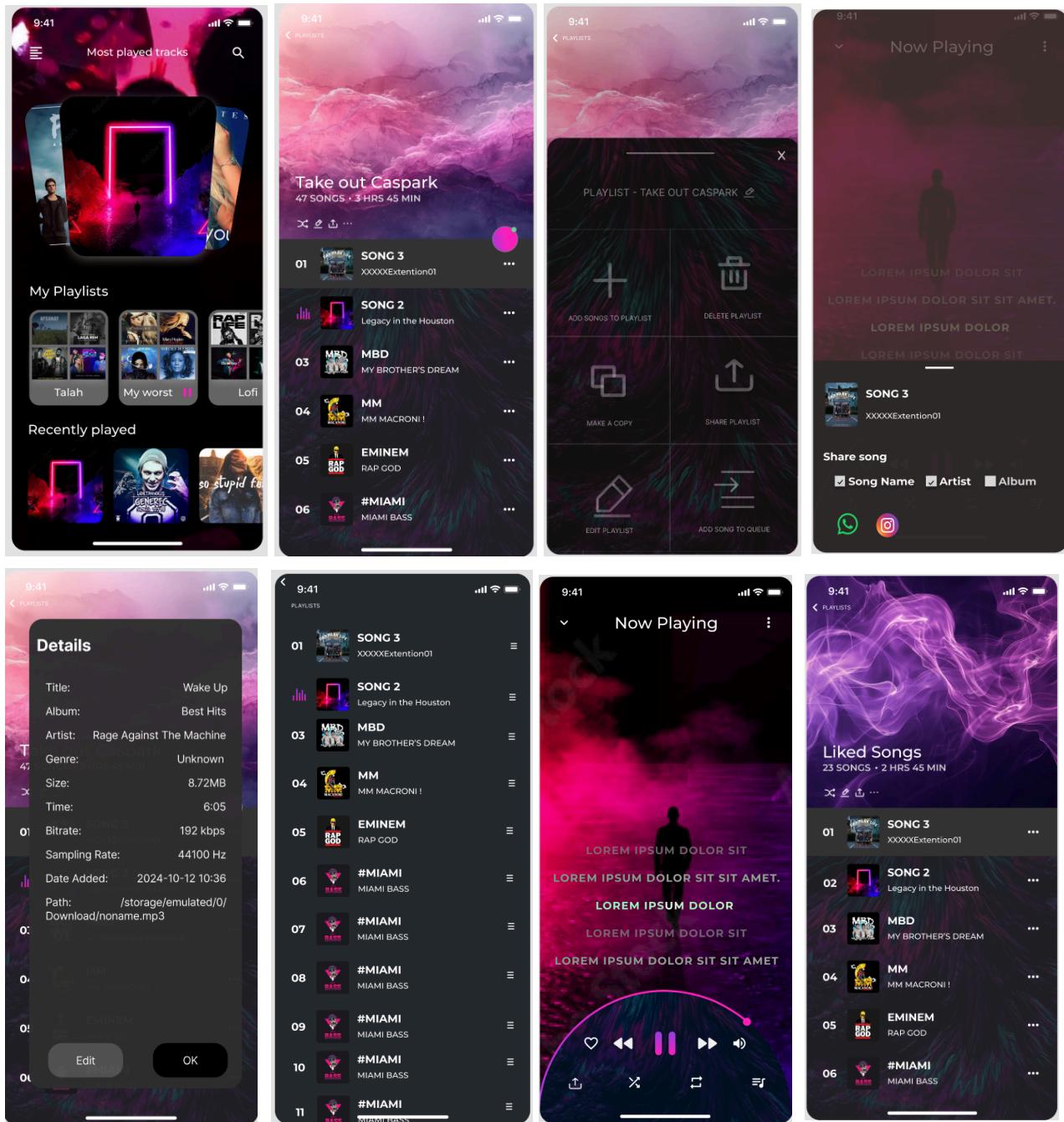
Due to the project constraints, we will focus on music users who have their music locally stored. Based on the derived requirements, our focus will shift towards providing better organization of playlists and prompts based on user's behavior rather than AI-powered recommendations. We have gathered requirements from music users across various platforms and usage types to ensure a comprehensive analysis.

Revised Low-Fidelity Prototype

Wireframes

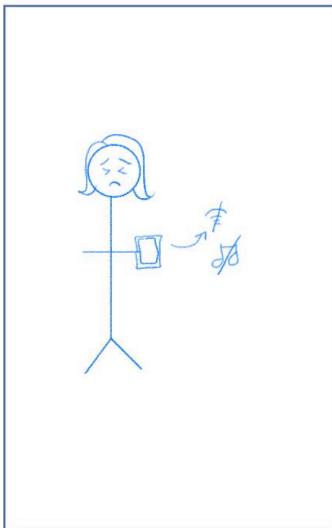


Mockups



Storyboard

Scenario: A user relies on Tapify's offline capabilities and playlist customization to make their commute enjoyable and uninterrupted.



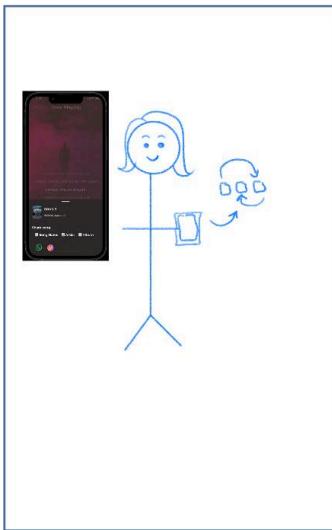
Sam struggles with network dependent features makes it hard to access playlist on the go.



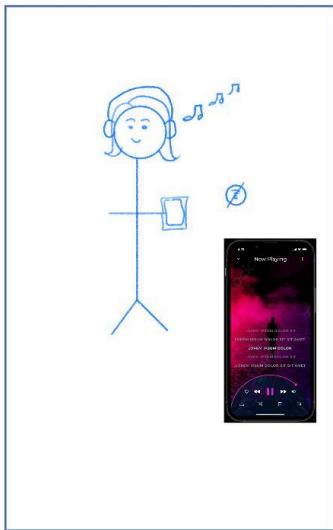
Sam opens Tapify and finds their favorite playlist displayed right on the homescreen, available offline.



With one tap, the user starts playing their playlist seamlessly, even without a network connection.



User easily reorder songs within the playlist, customizing it to match their preferences.

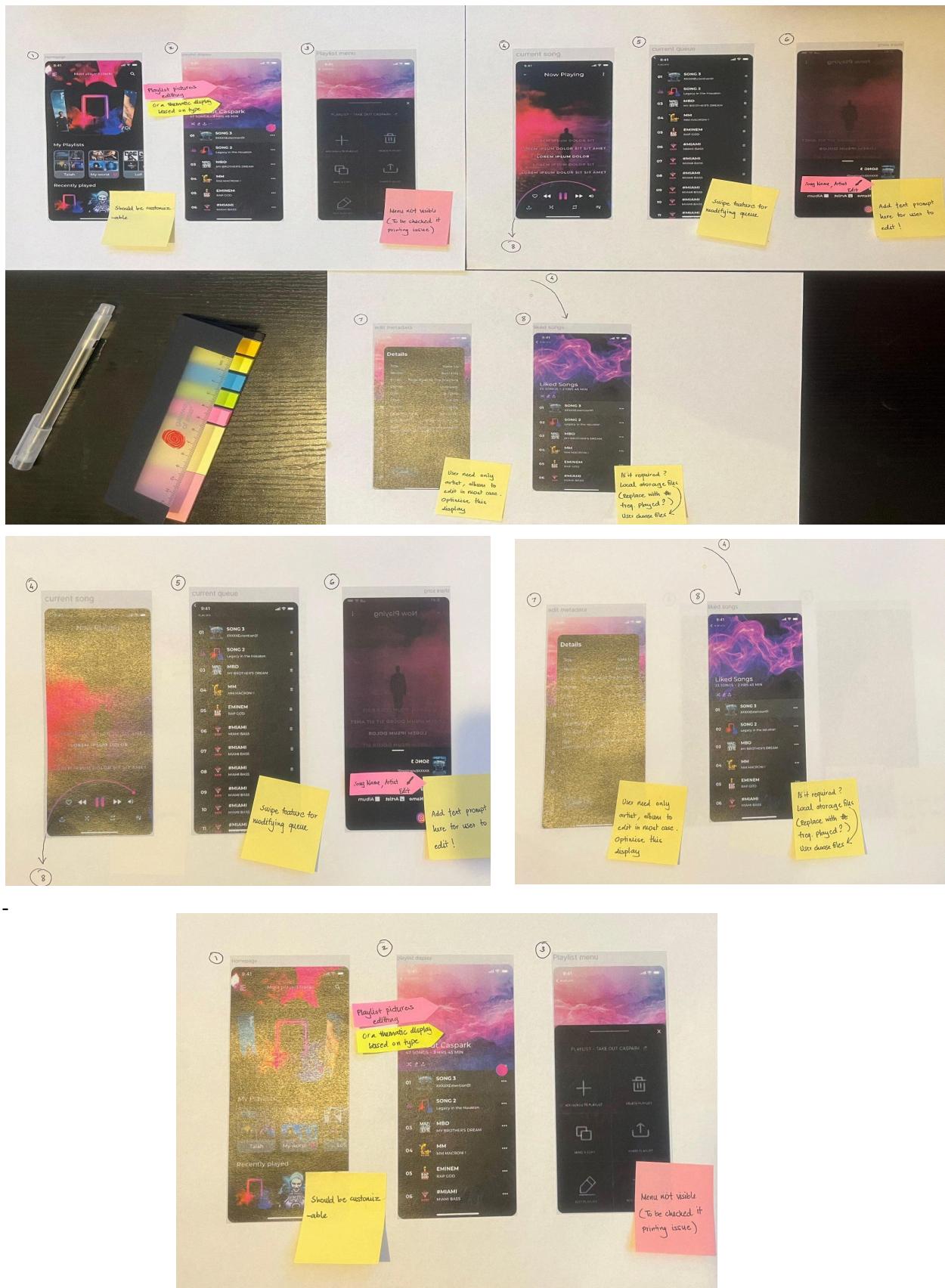


Tapify's offline mode, the user's music remains consistent and smooth regardless of connectivity.



Tapify provides a stress free music exp that meets the user needs.

Low-Fidelity Prototype Kit



Methods - Participants

Interview pool: 12 people

	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

Methods - Consent Form 5 points

Google form link to the consent form used by our group - [Consent Form for Music Application Usage Study](#). All interviewees were asked to fill the form prior to the interview. Based on their preferences regarding our data collection methods, interview analysis was tailored to extract information in the best possible manner.

Methods - Question Route

1. Please describe how you found and played your favorite playlist in Tapify. Is the operation smooth when switching between different playlists? Did you encounter any confusion?

2. When you edit or manage the playlist in Tapify, can you easily adjust it according to your mood or scene? What are your most commonly used functions?

3. When you are offline, can you easily find your stored offline music? Is the offline experience consistent with your expectations?
4. When you listen to music, do you choose a specific playlist according to your mood or activities? Is the "tags" function provided by Tapify convenient for you to edit and manage song information?
5. How do you organize or personalize your playlists in Tapify? Are there any features or settings that make it feel more tailored to your style, or anything missing that would help?
6. When sharing music, do you choose the sharing method in the form of simple text such as song title and artist name? Does this function help you to share music with others better?
7. Was there any action you tried to perform in Tapify that took more effort than expected? What was your experience trying to complete it?

Findings - Data analysis: coding

Code 1 - Ease of Use

The coding reflects the user's positive feedback on the intuitive design and convenient navigation of Tapify interface.

Participant 1: "The bottom navigation bar is very clear, and you can directly find the required functions."

Participant 2: "It is easy to find the recently played music, and the interface design is not complicated."

Users' evaluation of Tapify shows that simple layout and clear navigation can help them quickly locate functions and reduce operational troubles.

The key to user satisfaction lies in intuitive navigation and simple layout. Future designs should continue this style and ensure the consistency of interface and visual feedback when adding new functions.

Code 2 - Customization Options

This code reflects the importance that users attach to the customized options such as playlist management, metadata editing and personalization in Tapify applications. Users want to be able to edit and adjust playlists flexibly and to personalize the display of their music libraries.

Participant 1: "I like being able to quickly switch between different playlists, which saves a lot of time."

Participant 2: "The tag function of the application is very practical, which allows me to add tags related to emotions or activities to songs, so that I can find specific music later."

From the feedback of users, we can see that they attach great importance to the customization options provided by Tapify, especially in editing playlists and adjusting song information (such as artists, labels, etc.). The current customization function can help users better organize and manage their music libraries, thus enhancing the overall experience.

It is found that custom options have a great influence on user experience, so future designs should further enhance this function. For example, the existing metadata editing function can be extended to allow users to control the order, cover and category labels of playlists more finely. It is helpful to improve user satisfaction and application flexibility.

Code 3 - Social Sharing Features

This code reflects the user's demand and experience for the social sharing function in Tapify. Users want to be able to share their playlists and songs with others conveniently, while retaining music-related information, such as song names, artists and tags.

Participant 1: "I like the sharing option with metadata in the application, which can help me easily share certain music with my friends."

Participant 2: "The function of sharing playlists is very convenient. I can send them directly to my colleagues through links without worrying about the file format."

Social sharing is a feature that users value very much. Users want to share music or playlists with metadata directly, so that others can clearly understand the information of music. At the same time, these functions can reduce the incompatibility of file formats and improve the sharing efficiency.

It is found that the social sharing function has a great influence on the user experience. Allow users to choose what information to include when sharing according to their needs. Consider adding the function of collaborative playlist to enhance the interaction between users and the sense of community participation.

Code 4 - Offline Accessibility

This code reflects the user's demand and experience of Tapify offline access function, and emphasizes the importance of this function to improve the practicality and convenience of application.

Participant 1: "When there is no internet, I can still play my offline music easily, and the whole process is smooth."

Participant 2: "I like this application very much. It will automatically scan my device and find out all the audio files without me having to add them manually."

User feedback shows that Tapify's offline music access function is particularly important for users with unstable network or no network connection. Users want to access and play local music smoothly without Wi-Fi or mobile data, and reduce the complexity of manual operation.

Future designs should further optimize this function to ensure that users can easily find and play local music. Consider providing users with more offline music management options, such as offline music classification and label management functions, to enhance the user experience.

Implications for Design

What is Working Well

- **User Interface Clarity:** The low-fidelity prototype effectively communicated the core functionality of the app, such as the simple navigation and easy-to-use playlist organization. Users quickly grasped how to switch between different features and manage their music without confusion.
- **Positive Initial Feedback:** Participants were enthusiastic about the app's potential, particularly appreciating its minimalistic design and smooth navigation.

What Can Be Improved

- **Enhance Customization Options:** While the initial prototype included some customization, expanding these features will likely increase user satisfaction.
- **Integrate Social Sharing Features:** Respond to users' needs for social interaction by realizing simple playlist sharing, collaborative playlist creation and options for viewing friends' activities.
- **Focus on Performance:** Ensure that subsequent iterations prioritize speed and reliability, particularly for users with extensive music libraries.

Actionable Principles for Tapify Design

1. **Prioritize Intuitive Interaction Design:**
 - Ensure simple navigation and reduce cognitive burden by using clear visual cues, such as making interactive elements such as buttons obvious and clickable.
 - Implement consistent **mapping** between user actions and app responses to facilitate user understanding.
2. **Enhance Customization and Personalization:**
 - Incorporate options for users to personalize themes, layout, and playlist organization. Customization should be **visible** and accessible, allowing users to tailor the experience to their preferences.

- Use user feedback to guide the development of customization features, ensuring they resonate with user desires.
3. **Integrate Social Features Thoughtfully:**
- Design social sharing options that allow users to effortlessly share playlists or see friends' activities. Implement **feedback mechanisms** to notify users when their shared content has been viewed or engaged with.
 - Explore the potential of adding collaborative playlists to encourage user interaction and enhance community participation.
4. **Focus on Performance and Reliability:**
- Conduct rigorous testing to ensure the app performs well under various conditions, particularly when handling large libraries. Address any identified **gulfs of execution**, where users encounter obstacles in achieving their goals.
 - Improve the loading speed and response speed in the development process to provide a smooth and efficient user experience.

Addressing Information, Interaction, and Experience Design Issues

- **Information Design:** Users need a clear understanding of how to access their music and customize their experience. Organize information hierarchically to aid **sensemaking**, making it easy for users to locate their content quickly. Use clear labels and annotations to guide users.
- **Interaction Design:** Provide **visible feedback** for interactions to ensure users feel confident in their actions. For example, when a playlist is shared, confirm the action through a notification or animation.
- **Experience Design:** Anticipate user needs and desires, and build a design that fosters an engaging experience. Users are more likely to return if the app not only meets their functional needs but also provides a pleasurable interaction.

Unexpected Findings

- Although this is a low-fidelity prototype, users have shown a strong demand for customized functions, which highlights the importance of personalization in meeting users' needs. This shows that users want to have a high degree of control over their music experience, and applications like Tapify should take customization as the basic aspect of design to enhance users' participation and satisfaction.

Successes

- The prototype effectively communicated core functionalities, and user enthusiasm suggests a strong interest in the app's potential. The core functions will be further optimized and expanded on the basis of user feedback.

Conclusion and Next Steps

Based on the findings and principles outlined above, it is essential to iteratively refine Tapify by incorporating user feedback, enhancing customization, integrating social features, and prioritizing performance. This iterative approach helped guide to a more functional prototype for subsequent development phases. Give priority to the implementation of collaborative playlist function, and optimize the application performance of users of large music libraries.

Process

Our initial user interviews provided valuable insights into the features users appreciate and regularly use in existing music applications. This helped us refine our focus from general music player capabilities to the specific features users find intuitive or frustrating. In our first prototype, participants shared their preferences, emphasizing the need for a straightforward, user-friendly UI. Based on this feedback, we implemented high-demand features like quick access to playlists, albums, and artists, along with options for a customizable homepage. We also experimented with AI-powered, non-intrusive recommendations tailored to users' listening habits, aiming to subtly suggest relevant music without disrupting their experience. However, due to API limitations and music copyright restrictions, we ultimately marked this recommendation feature as out of scope. This phase of the study provided a broad look at the potential of a comprehensive music app experience.

In the next prototype, we refined our focus to the core functions of a music player, narrowing the scope to serve a more targeted user base. Users of offline music player apps generally fall into two categories: dedicated music enthusiasts seeking niche content not widely available on standard streaming platforms, and individuals with limited connectivity who rely on offline access. By engaging with streaming app users, we pinpointed specific frustrations around basic music player expectations, refining our understanding of their primary needs.

With our second prototype, further testing revealed detailed preferences and areas for improvement. Some users noted missing essential features, such as the ability to search directly from the homepage or manage playlists more flexibly, which we integrated into the updated design. Others commented that certain features, like the shuffle function on every page or desaturated icons, felt overly complicated or visually uninspiring. We responded by enhancing feature visibility, adjusting the design for better usability, and updating icons to improve clarity and intuitiveness.