Product Requirements Document: Echo

1. Introduction

Echo is an innovative web-based audio entertainment application designed to transform how users interact with sound, offering both deeply personal auditory experiences and dynamic real-time collaborative music creation. It aims to provide a unique blend of ambient soundscapes for individual focus and relaxation, alongside an interactive platform for collective sonic exploration.

Vision: To be the go-to platform for personalized audio ambiance and accessible real-time collaborative sound creation, fostering a global community around shared auditory experiences.

2. Goals (Hackathon Focus)

This PRD outlines the scope for a one-day hackathon project, prioritizing core functionality and demonstrating scalability.

- **Creativity:** Deliver a novel experience that combines personalized mood-based soundscapes with real-time collaborative music jamming, simplifying complex audio creation for everyday users.
- **Practicality:** Develop a functioning web solution leveraging readily available APIs for rapid implementation, demonstrating clear user motivation and a path for real-world scalability.
- **Presentation:** Create a compelling and engaging demonstration of both "Flow Mode," "Jam Mode," and the new "Play Mode" with intuitive UI/UX.
- **Design:** Ensure a user-friendly and aesthetically pleasing interface that enhances user engagement and interaction with audio elements.
- **Technical Complexity:** Showcase robust real-time synchronization (WebSockets) for collaborative audio and game state, and intelligent API integration for content curation, achieving significant functionality within a short timeline.

3. Target Audience

- Individuals seeking personalized ambient soundscapes for focus, relaxation, or mood enhancement.
- Friends, casual musicians, and creative individuals looking for a low-barrier way to jam and create music together in real-time.
- Online communities interested in shared, interactive entertainment experiences, including social gaming.

4. User Stories & Features

4.1. Core Features (Applicable to all modes)

- **User Authentication:**
  - As a user, I want to sign up/log in securely so I can save my preferences and access my created content.
  - As a user, I want to easily sign in (e.g., via Google/email) to quickly access the app.
- **Navigation:**
  - As a user, I want to easily switch between "Flow Mode," "Jam Mode," and "Play Mode" from the main interface.

4.2. Flow Mode (Personalized Soundscapes)

Purpose: To provide a continuous, dynamically mixed soundscape tailored to the user's mood, activity, or environment, with enhanced integration for personal music.

- **Mood/Activity Selection:**
  - As a user, I want to select my current mood (e.g., "Calm," "Energetic," "Focused") or activity (e.g., "Studying," "Sleeping," "Working Out") from a predefined list.
  - As a user, I want the soundscape to dynamically adjust based on my selection, blending appropriate ambient tracks and effects.
- **Personal Music Integration (NEW - Limited Commercial Music Use):**
  - As a user, I want to connect my personal Spotify account (or similar streaming service) to Flow Mode.
  - As a user, I want EchoFlow to *analyze the mood/genre* of the music I'm currently playing on my *connected streaming service* (e.g., Spotify).
  - As a user, I want EchoFlow to *suggest or automatically overlay* ambient sounds or visualizer themes from its *royalty-free library* that complement my currently playing commercial music.
  - *Note: EchoFlow will not stream or play the commercial music itself; it will only react to music played by the user on their own licensed streaming service.*
- **Contextual Modifiers (Stretch Goal for Hackathon):**
  - As a user, I want the soundscape to subtly adapt based on the local weather (e.g., add rain sounds if it's raining outside).
  - As a user, I want the soundscape to subtly adapt based on the time of day (e.g., softer sounds in the evening).
- **Dynamic Visualization:**
  - As a user, I want to see a simple, responsive visualizer (e.g., color gradients, abstract shapes) that reacts to the current soundscape and/or my personal

music, enhancing the immersive experience.

- **Playback Controls:**
  - As a user, I want basic playback controls (play/pause, volume) for EchoFlow's generated soundscape.
- **Sleep/Focus Timer (Stretch Goal for Hackathon):**
  - As a user, I want to set a timer for the soundscape to fade out after a specified duration (e.g., 30 minutes for sleep).

4.3. Jam Mode (Collaborative Sonic Canvas)

Purpose: To enable real-time, synchronized audio creation and manipulation among multiple users, with a nod to commercial music.

- **Room Creation/Joining:**
  - As a user, I want to create a new "Jam Room" and get a unique shareable link.
  - As a user, I want to join an existing "Jam Room" using a shareable link.
  - As a user, I want to see a list of public Jam Rooms I can join.
- **Shared Audio Library:**
  - As a user, I want to browse a curated library of royalty-free audio loops, instrumental snippets, and sound effects within the Jam Room.
- **Commercial Song Suggestion (NEW - Limited Commercial Music Use):**
  - As a user, I want to search for and suggest a commercial song (by title/artist) from a music metadata API (e.g., Spotify).
  - As a user, I want to see a *short, non-playable preview snippet* (if available via API, e.g., 30-second preview URL) of the suggested commercial song in the Jam Room. This is for reference/inspiration only; the full song will *not* be played or mixed within the app.
  - As a user, I want to see other participants' suggestions.
  - *Note: The actual jam will still be created using the royalty-free loops and effects. This feature is for inspiration and social interaction around commercial music, not its live playback.*
- **Real-time Layering & Manipulation:**
  - As a user, I want to select a sound from the library and trigger its playback in sync with other participants.
  - As a user, I want to control the volume of my individual sound contributions within the shared mix.
  - As a user, I want to see which other participants are currently playing sounds.
- **Basic Text Chat:**
  - As a user, I want to send text messages to other participants within the Jam Room to coordinate or comment on the music.
- **Shareable Visual Jam Snippets (POP FEATURE):**

    - As a participant/host, I want to record a short (e.g., 5-10 second) video/GIF snippet of the visualizer reacting to the live jam, along with the audio.
    - As a user, I want to easily share this recorded visual snippet on social media platforms.
- **Emoji Reactions (Stretch Goal for Hackathon):**
    - As a user, I want to send quick emoji reactions that appear briefly for all participants, expressing my real-time feedback on the jam.

4.4. Play Mode (Social Mini-Games) - NEW FEATURE

Purpose: To provide interactive, turn-based social gaming experiences within EchoFlow's collaborative environment.

- **Game Room Creation/Joining:**
    - As a user, I want to create a new "Game Room" or join an existing one, separate from Jam Rooms, for playing mini-games.
    - As a user, I want to invite friends to a Game Room via a shareable link.
- **Mini-Game Selection:**
    - As a user, I want to choose from a selection of simple, turn-based mini-games (e.g., Collaborative Trivia, Word Association, Song Charades).
- **Real-time Game State & Scoring:**
    - As a user, I want to see game questions, my turn status, and the real-time scoreboard.
- **In-Game Chat/Interaction:**
    - As a user, I want to communicate with other players during the game (e.g., answer trivia questions in chat).

Example Mini-Games (Hackathon Scope for MVP):

- **Collaborative Trivia:**
    - As a user, I want to answer trivia questions pulled from an external API.
    - As a user, I want to see if my answer is correct and view the correct answer.
    - As a user, I want to see the scores update in real-time.
- **"Guess the Flow" (Cross-Mode Game):**
    - As a user (the "guesser"), I want to hear a short, randomly generated soundscape from Flow Mode.
    - As other users, I want to guess the mood or activity that the soundscape was generated for.
    - As the host, I want to reveal the correct answer and award points.
- **Song Charades:**
    - As a player, I want to receive a random song title/artist/album from a music metadata API (e.g., Spotify).

    - As a player, I want to act out the song title/artist/album using mime or by triggering sounds/effects from the Jam Mode library.
    - As a guessing player, I want to type my guesses into the chat.
    - As a player, I want to earn points for correct guesses (both actor and guesser).

5. Technical Considerations (Hackathon Stack)

- **Frontend:** React (preferred for component-based development and rapid UI building) or plain HTML/CSS/JavaScript.
    - **Styling:** Tailwind CSS for rapid, responsive UI development.
    - **Audio Processing:** Web Audio API for client-side sound generation, mixing, and manipulation.
    - **Visuals:** HTML Canvas or simple CSS animations for dynamic visualizations.
    - **Video Recording (for snippets):** MediaRecorder API for capturing canvas and audio output.
- **Backend (Minimal/Serverless):**
    - **Authentication:** Supabase Auth (__initial_auth_token for Canvas environment, signInAnonymously as fallback).
    - **Real-time Data Sync (Jam & Play Modes):** Supabase Realtime with onSnapshot listeners for real-time room state, participant presence, audio event synchronization, and **game state management (questions, answers, scores, charade prompts)**. This will act as the WebSocket alternative for rapid development.
    - **Data Storage (User preferences, saved jams, game data):** Supabase Database for structured data storage.
    - **API Key Management:** Handled by the Canvas environment for Google APIs.
- **Key APIs for Rapid Development:**
    - **Supabase (Auth, Database, Realtime):** For user management, room state, real-time sync, and **game state**.
    - **Spotify Web API:** For Flow Mode content *curation* (searching for ambient tracks, instrumental playlists based on mood tags), **for fetching song/artist/album metadata for Song Charades**, and **for fetching preview URLs/audio analysis for Flow Mode's Personal Music Integration and Jam Mode's Commercial Song Suggestion**. *Crucially, NOT for real-time Jam Mode full-track playback sync, nor for full song audio playback in Play Mode due to licensing and technical complexity.*
    - **FreeSound API / Other Royalty-Free Audio Libraries:** For sound effects and loops for both modes.
    - **Open Trivia Database API:** For fetching trivia questions and answers in Play

Mode.
    - **OpenWeatherMap API:** For weather-based contextual soundscapes (Flow Mode stretch).
    - **Google Maps Places API / Browser Geolocation API:** For location-based contextual soundscapes (Flow Mode stretch).
    - **Optional (Cloud Storage API):** For saving and sharing recorded Jam snippets (e.g., Google Cloud Storage, if recording is implemented).

6. Success Metrics (Hackathon Judging Criteria)

- **Functionality:** All three modes (Flow, Jam, Play) demonstrate core features reliably.
- **User Experience:** Intuitive navigation between modes, clear controls, and engaging visuals/interactions.
- **Real-time Sync (Jam & Play Modes):** Smooth and low-latency synchronization of audio events and game state across multiple participants.
- **Personalization (Flow Mode):** Clear demonstration of mood-based soundscape generation and reaction to personal music.
- **Scalability Potential:** The design clearly shows how user-generated content (saved jams, new loops, shared visual snippets) and network effects (more users in jams/games) would drive future growth.
- **API Usage:** Effective and intelligent use of external APIs to accelerate development and provide rich features across all modes, including the nuanced integration of commercial music metadata.
- **Presentation:** A clear, engaging, and well-rehearsed demonstration that highlights the unique aspects and potential of EchoFlow as a comprehensive social entertainment hub.

7. Future Enhancements (Post-Hackathon)

- **Advanced Audio Manipulation (Jam Mode):** Add effects (reverb, delay), pitch shifting, tempo control for individual loops.
- **User-Uploaded Loops (Jam Mode):** Allow users to upload their own short audio snippets to the shared library.
- **AI-Powered Mood Detection (Flow Mode):** Integrate sentiment analysis (text input) or even facial expression recognition (via webcam) to automatically detect and set the user's mood.
- **More Mini-Games (Play Mode):** Expand the library of games (e.g., Pictionary-style drawing game with a simple drawing canvas, collaborative storytelling, music-themed charades).
- **Monetization Models:** Premium features (more loops, advanced effects, longer

recording times, exclusive game content), artist collaborations, virtual goods.
- **Cross-Device Sync:** Seamless experience across desktop and mobile.
- **Community Features:** User profiles, following other "jammers" or "gamers," curated "best jams" showcases, game leaderboards.
- **Integration with Smart Devices:** Control Flow Mode from smart speakers or smart home systems.
- **Full Commercial Music Integration (Long-term, Post-Hackathon):** Explore partnerships and licensing agreements with music labels and publishers to enable full, synchronized playback of commercial music in Jam Mode, potentially through direct integrations with streaming service SDKs designed for such use cases (if they become available). This would be a significant undertaking beyond a hackathon.