**Project Design Phase-II**

**Solution Requirements (Functional & Non-functional)**

|  |  |
| --- | --- |
| Date | 31 January 2025 |
| Team ID | SWTID1741256970149949 |
| Project Name | Rhythmic Tunes |
| Maximum Marks | 4 Marks |

**Functional Requirements:**

Following are the functional requirements of the proposed solution.

**Functional Requirements – Music Streaming App**

**User Registration & Authentication**

* 1. **Description:** Users must be able to sign up, log in, and manage their accounts.
  2. **Rhythmic Analogy:** Just as every concert begins with the tuning of instruments, user authentication sets the stage by ensuring each participant is ready for the performance.

**Profile & Preferences Management**

* 1. **Description:** Users can update their profile, set preferences, and manage settings.
  2. **Rhythmic Analogy:** Similar to a musician choosing their favorite instruments and tuning them to perfection, users customize their experience to suit their style.

**Music Playback & Streaming**

* 1. **Description:** The app should support continuous streaming with essential controls like play, pause, skip, rewind, and volume control.
  2. **Rhythmic Analogy:** Like the steady pulse of a drumbeat or the flow of a melody, smooth playback ensures the tune remains uninterrupted and engaging.

**Search & Music Discovery**

* 1. **Description:** Users need to search for songs, artists, albums, and genres, with filters to refine results.
  2. **Rhythmic Analogy:** Imagine a seasoned DJ sifting through a vast library of beats to find the perfect track—the search function lets users explore the endless repertoire of rhythmic patterns.

**Playlist Creation & Management**

* 1. **Description:** Users can create, edit, and delete playlists, as well as add or remove tracks.
  2. **Rhythmic Analogy:** Crafting playlists is like arranging a setlist for a live show—each track is a note that contributes to the overall harmony of the performance.

**Personalized Recommendations & Playlists**

* 1. **Description:** The app should analyze user behavior to offer song and playlist recommendations.
  2. **Rhythmic Analogy:** Think of a conductor who senses the mood of the audience and adjusts the tempo—recommendations keep the rhythm alive by playing what resonates best with the listener.

**Social Sharing & Interaction**

* 1. **Description:** Users can share their favorite tunes or playlists on social media, follow other users, and see trending tracks.
  2. **Rhythmic Analogy:** Like an impromptu jam session where musicians and fans share their passion, social features create a community that celebrates and amplifies the music.

**Offline Mode & Downloading**

* 1. **Description:** Allow users to download songs or playlists for offline listening.
  2. **Rhythmic Analogy:** Just as a musician keeps a personal set of recordings for inspiration on the go, offline mode lets users carry their favorite beats wherever they are.

**High-Quality Audio Streaming**

* 1. **Description:** Support multiple streaming qualities and adaptive streaming to match network conditions.
  2. **Rhythmic Analogy:** Every note in a live performance must be crisp and clear; high-quality streaming ensures that the richness of the sound is preserved, whether in a quiet room or a bustling street.

**Analytics & User Feedback**

* 1. **Description:** Collect and analyze user interaction data to continuously improve the app, and provide channels for user feedback.

**Non-functional Requirements:**

Following are the non-functional requirements of the proposed solution.

· **Performance (Steady Beat)**

* · **Requirement:** The system must stream audio with minimal latency and seamless transitions to maintain continuous playback.
* **Rhythmic Analogy:** Like a drummer keeping a steady beat, the app must deliver a consistent tempo without any hiccups or interruptions.

· **Scalability (Expanding Orchestra)**

* · **Requirement:** The system should handle a growing number of users and concurrent streams without degrading performance.
* **Rhythmic Analogy:** Imagine an orchestra that gracefully adds new instruments while preserving harmony—the system should scale up smoothly as demand increases.

· **Security (Protected Score)**

* · **Requirement:** Ensure user data, streaming content, and transactions are secured through encryption and robust authentication measures.
* **Rhythmic Analogy:** Think of security as safeguarding a priceless musical score in a vault, ensuring that every note and beat remains protected.

· **Reliability (Consistent Metronome)**

* · **Requirement:** The application must provide stable connectivity and high uptime, minimizing unexpected downtimes or interruptions.
* **Rhythmic Analogy:** Just as a metronome delivers a constant, unwavering beat, the system should reliably keep the music playing under various conditions.

· **Usability (Intuitive Melody)**

* · **Requirement:** The user interface should be intuitive, allowing users to easily search, discover, and control their music.
* **Rhythmic Analogy:** Like a catchy melody that instantly resonates with listeners, the app's design must be straightforward and engaging.

· **Maintainability (Well-Tuned Instruments)**

* · **Requirement:** The system should be designed for easy updates, debugging, and enhancements to adapt to evolving user needs.
* **Rhythmic Analogy:** Just as instruments are regularly tuned to produce their best sound, the application must be maintained efficiently to sustain optimal performance.

· **Availability (Open Stage)**

* · **Requirement:** The service should be accessible 24/7, ensuring users can stream music at any time, from any location.
* **Rhythmic Analogy:** Like a live concert that welcomes its audience at all hours, the app should be ready to perform whenever called upon.

· **Portability (Universal Instrumentation)**

* · **Requirement:** The app must offer a consistent experience across various platforms and devices, from smartphones to desktops.
* **Rhythmic Analogy:** Much like a timeless tune that sounds great whether played on a grand piano or an acoustic guitar, the application should maintain its quality no matter where it's experienced.

· **Interoperability (Harmonious Collaboration)**

* · **Requirement:** The system should integrate smoothly with third-party services, social media platforms, and external devices.
* **Rhythmic Analogy:** Just as different instruments collaborate to create a harmonious performance, the app must work well with external systems to enrich the overall user experience.