**Project Design Phase-II**

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

| Date | 31 January 2025 |
| --- | --- |
| Team ID | SWTID1741110351 |
| Project Name | Rhythmic Tunes |
| Maximum Marks | 4 Marks |

**Functional Requirements:**

Following are the functional requirements of the proposed solution.

**Functional Requirements – Music Streaming App**

| **FR No.** | **Functional Requirement (Epic)** | **Sub Requirement (Story / Sub-Task)** |
| --- | --- | --- |
| **FR-1** | **Music Search & Discovery** | Search for Songs, Albums, and Artists |
|  |  | View Trending and Recommended Music |
| **FR-2** | **Playback & Streaming** | Play, Pause, and Skip Songs |
|  |  | Display Album Art and Song Details |
| **FR-3** | **Playlist & Favorites** | Create and Manage Playlists |
|  |  | Add or Remove Songs from Playlists |
|  |  | Like / Favorite Songs |
| **FR-4** | **Audio Streaming** | Stream High-Quality Audio |

**Non-functional Requirements:**

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

| **NFR No.** | **Non-Functional Requirement** | **Description** |
| --- | --- | --- |
| **NFR-1** | **Usability** | The app should have an intuitive and user-friendly interface, ensuring smooth navigation and accessibility for users of all demographics. |
| **NFR-2** | **Security** | User authentication and data must be secured using encryption (e.g., HTTPS, OAuth for third-party logins). The app should prevent unauthorized access and follow best security practices. |
| **NFR-3** | **Reliability** | The app should ensure a consistent and uninterrupted music streaming experience, minimizing crashes and downtime. |
| **NFR-4** | **Performance** | Songs should load and stream with minimal buffering. The app should respond to user interactions (search, playback, playlist management) within 2 seconds. |
| **NFR-5** | **Availability** | The system should maintain an uptime of at least 99.9%, ensuring accessibility across different time zones. |
| **NFR-6** | **Scalability** | The app should handle increasing numbers of users and concurrent streams efficiently without performance degradation. The architecture should support future feature expansion. |