**Project Design Phase-II**

**Technology Stack (Architecture & Stack)**

|  |  |
| --- | --- |
| Date | 31 January 2025 |
| Team ID | SWTID1741244908156274 |
| Project Name | **RhythmicTunes: Your Melodic Companion** |
| Maximum Marks | 4 Marks |

|  |  |
| --- | --- |
| K S. Arul sakthi | sakthisakthi15062004@gmail.com |
| R. Sathish Kumar | sathish2631715@gmail.com |
| MI. Manikandan | manimrkingofficial@gmail.com |
| CH Manashkumar | mmanashkumar41@gmail.com |
| S. Jagadeesh | jagadeeshjagadeesh230@gmail.com |

**Technical Architecture:**

The Deliverable shall include the architectural diagram as below and the information as per the table1 & table 2

**Example:** Rhythmic Tunes

**Reference: <https://open.spotify.com/>**

**Table-1 : Components & Technologies:**

| **S.No** | **Component** | **Description** | **Technology** |
| --- | --- | --- | --- |
|  | User Interface | Web-based interface for music streaming | HTML, CSS, JavaScript / React Js etc. |
|  | Application Logic-1 | Music streaming and metadata management | React js, Node js |
|  | Application Logic-2 | Playlist and user preference management | React js, Node js |
|  | Database | Stores Songs, playlists, and metadata | JSON Web Server |

**Table-2: Application Characteristics:**

| **S.No** | **Characteristics** | **Description** | **Technology** |
| --- | --- | --- | --- |
|  | Open-Source Frameworks | Frontend frameworks | React.js, Node.js, BootStrap, Tailwind CSS |
|  | Scalable Architecture | 3-tier architecture with RESTful APIs | Microservices |

**References:**

**[React.js Documentation](https://react.dev/)**

**[Node js Best Practice](https://nodejs.org/en/learn/getting-started/introduction-to-nodejs)**

**[JSON Web Server Referance](https://www.npmjs.com/package/json-server)**

**<https://medium.com/the-internal-startup/how-to-draw-useful-technical-architecture-diagrams-2d20c9fda90d>**