# SRS For Musical Note:

# **Template of SRS:**

1. **Introduction**
   1. Purpose
   2. Document Conventions
   3. Intended Audience and Reading Suggestions
   4. Product Scope
   5. References
2. **Overall Description**
   1. Product Perspective
   2. Product Functions
   3. User Classes and Characteristics
   4. Operating Environment
   5. Design and Implementation Constraints
   6. User Documentation
   7. Assumptions and Dependencies
3. **External Interface Requirements**
   1. User Interfaces
   2. Hardware Interfaces
   3. Software Interfaces
   4. Communications Interfaces
4. **System Features**
   1. System Features one by one

##### **5. Other Nonfunctional Requirement**

##### 1. Performance Requirements

* 1. Security Requirements
  2. Safety Requirements
  3. Software Quality Attributes
  4. Business Rules

##### 1. Introduction

This software requirement specification (SRS) report expresses complete description about Musical note Project. This document includes all the functions and specifications with their explanations to solve related problems.

##### 1.1 Purpose

This document is intended for the following group of people:-

1. Developers for the purpose of maintenance and new releases of the software.
2. Documentation writers.
3. Testers.
4. This document is intended for both the stakeholders and the developers of the system.

##### 1.2 Scope

The goal is to design a recommender system on music domain. The system will be composed of server-side components and client-side components. The server-side component will manage the database operations and algorithms that produce recommendation results. The client-side components will be graphical interfaces that are integrated into corresponding larger systems.

Generally, a music recommender system consists of three key components: users, items and user-item matching algorithms. So, the background of our product is becoming more important than foreground component. At the background, the system will gather data from both users and items

##### 1.3 Definitions, Acronyms, and Abbreviations

|  |  |
| --- | --- |
| SRS | Software Requirements Specification |
| Node JS | Node.js is an open-source, cross-platform, back-end JavaScript runtime environment that runs on the V8 engine |
| HTML | Hypertext Markup Language |
| CSS | Cascading Style Sheet |
| JS | JavaScript (JS) is a lightweight, interpreted, or just-in-time compiled programming language with first-class functions. While it is most well-known as the scripting language for Web pages. |
| TS | TypeScript is a programming language developed and maintained by Microsoft. It is a strict syntactical superset of JavaScript and adds optional static typing to the language. TypeScript is designed for the development of large applications and transcompiles to JavaScript. |
| Angular | **Angular** is a platform for building mobile and desktop web applications. |
| BootStrap | Bootstrap is a free and open-source CSS framework directed at responsive, mobile-first front-end web development. |

##### 1.4 References

The references for the above software are as follows:-

1. [www.google.co.in](http://www.google.co.in/)
2. [www.wikipedia.com](http://www.wikipedia.com/)
3. [www.spotify.com](http://www.spotify.com)
4. [www.w3schools.com](https://www.w3schools.com/)
5. [Bootstrap.com](http://www.getbootstrap.com)
6. [fontawesome.com](https://fontawesome.com/)
7. [Angular](https://angular.io/)

##### 1.5 Overview

Section 1.0 discusses the purpose and scope of the software.

Section 2.0 describes the overall functionalities and constraints of the software and user characteristics. Section 3.0 and 4.0 details all the requirements needed to design the software.

##### 2. The Overall Description

In this part, background information about specific requirements of the system will be

provided briefly. General issues that affect the product and outline of the functional

requirements will be mentioned, too. In short, this section will mainly give information about

product perspective, product functions, User Characteristics, constraints, assumptions and dependencies.

2.1 Product Perspective

Considering the users who love music we designed the product with a advanced music identification strategy and also the Music will be updated with the latest Music. Included Wishlist where user can add favorite music to wishlist.

##### 2.2 Product Functions

The major functions that **Musical Note** performs are described as follows:-

1) User can access to the website and songs without registering.

2) User can add his favorite songs to his Wishlist.

3) Enhanced Search option is available.

4) Login option is available for best user experience.

##### 2.3 User Characteristics

**2.3.1 User group 1: Kids**

Kids are consumers in the age 7-17 are very active in building and sharing playlists. They might have difficulty understanding English, so default settings and visual interaction is a must.

**2.3.2 User group 2: Adults**

Adults are in the ages 18-upwards and have well organized playlists for different modes and occasions. Whenever new features are released these groups need to be notified, they will not enjoy discovering things by curiosity.

##### 2.4 Constraints

The major constraints that the project has are as follows: -

* Internet obligatory.
* The product will be operating in a Windows environment. It will also be compatible with the IE 6.0. Most of the features will be compatible with the Mozilla Firefox & Opera 7.0 or higher version. They will also be compatible with Internet Explorer, Google Chrome and Microsoft Edge. The only requirement this online platform would utilize would be a proper Internet connection.
* Millions of data will be needed to test the software. At this stage developers will need huge amount of disk space and clusters.

##### 2.5 Assumptions and Dependencies

The requirements stated in the SRS could be affected by the following factors:

* It is assumed that everyone will have proper internet access.
* However, we can also study in other platforms (It looks similarly to other platforms).
* User should have basic knowledge of music applications and its components.

##### 3. System Features

**1.Functional requirements**

* Music Online: Users will be redirected to the home page and all the users will be given the access to play any music they need.

##### Wish list: To Wishlist a particular song or Music Track user need to login for updating his details so logging in is mandatory for adding Wishlist.

* Latest Music: Admin will be updating the music frequently basing upon the new song in the market where user will be provided a best music track and songs
* Sequencing Information: The information about the users and their account should be entered into the database prior to any of the transactions and the backup be maintained for all account information
* Error Handling/ Response to Abnormal Situations: If any of the above validation/sequencing flow does not hold true, appropriate error messages will be prompted to the user for doing the needful.

##### 4. External Interface Requirements

**4.1 User Interface Requirements**

* 1. Quality User Interface
  2. Included Contact us Page for detailed clarification for users.
  3. Login and Registration is available
  4. For Registered users, wishlist is added to add their favourite songs.

##### **4.2 Software Interface Requirements**

The product will be operating in a Windows environment. It will also be compatible with the IE 6.0.

Most of the features will be compatible with the Mozilla Firefox & Opera 7.0 or higher version. They will also be compatible with Internet Explorer, Google Chrome and Microsoft Edge.

The only requirement this online platform would utilize would be a proper Internet connection.

The database used to keep record of user accounts shall be Firebase.

##### 4.3 Communication Interface Requirements

The user needs to communicate with the database for each session for various functions such as login verification, account access etc. so the following are the various communication interface requirements that are needed to be fulfilled in order to run the software successfully: -

Protocol used for data transfer shall be Hypertext Transfer Protocol (HTTP).

##### 5. Other Non-functional Requirements

1. Performance Requirements

The following list provides a brief summary of the performance requirements for the software:

1. Capacity: The Application shall provide customers a 24-hour service.

2. Dynamic requirements: When there is a Stable Internet connection the Web Application will be open in 2-3sec where as it takes a bit longer when there is no Stable connection.

3. Quality: Updating or Deletion of the user data is done in no time, Best hand picked and sorted Music will be delivered through the Web Application

The primary objective is to produce quality Web Application. As the quality of a piece of Web Application is difficult to measure quantitatively, the following guidelines will be used when judging the quality of the Application:

1. Consistency: Application will be consistent with respect to the style. (Implied when Music is played)
2. Test cases – All functionality will be thoroughly tested
   1. Software System Attributes

* Reliability: The data communication protocol shall be such that it ensures reliability and quality of data and voice transmission in a mobile environment. The memory system shall be of non-volatile type.
* Availability: The product will have a backup power supply in case of power failures. Any abnormal operations shall result in the shutting down of the system
* Security: The password shall be 6-14 characters long. Passwords shall not contain name of customers as they are easy to be hacked. Passwords can contain digit, hyphen and underscore.
* Maintainability: The system components i.e. modem, memory, disk, drives shall be easily serviceable without requiring access to the vault. The application will be monitored every time and if any problem is faced it will be solved and we update the music with the current top lists.