

**Music app - Flutter**

System Requirements Study Report

Submitted in partial fulfilment of the requirement for the

2nd sem. evaluation of

**Masters of Computer Application**

**To**

**RK University, Rajkot**

**Submitted By**

|  |  |
| --- | --- |
| Rushik Gohel | 23SOECA21080 |

**Under the guidance of**

**Prof. Nikunj Vadher**

RK University

Rajkot

**Software Requirements Specification (SRS) for Music App**

1. **Introduction**

**1.1 Purpose**

The purpose of this document is to define the requirements for the development of a music app in Flutter. This app aims to provide users with features for listening to music, playing and stopping music, exploring artists, ensuring security, and including a splash screen.

**1.2 Scope**

The music app will allow users to browse, search, and listen to music tracks, explore artists, and enjoy an enhanced music listening experience.

**1.3 Definitions, Acronyms, and Abbreviations**

- SRS: Software Requirements Specification

- Flutter: An open-source UI software development toolkit created by Google

**1.4 References**

- Flutter Documentation

- Music API Documentation

**1.5 Overview**

This document outlines the functional and non-functional requirements for the music app, including specifications for listening, playing and stopping music, exploring artists, security measures, and the splash screen.

1. **Functional Requirements**

**2.1 Listening**

- Users should be able to browse and search for music tracks.

- Users should be able to play selected music tracks.

- Users should be able to pause, resume, and stop music playback.

**2.2 Play Music**

- Users should be able to play music from the beginning or from a specified time.

- Users should be able to adjust volume levels.

**2.3 Stop Music**

- Users should be able to stop music playback at any time.

- The app should provide a clear indication when music playback is stopped.

**2.4 Artists**

- Users should be able to browse and search for artists.

- Users should be able to view information about artists, including biographies and discographies.

- Users should be able to view artist images.

**2.5 Security**

- Users should be able to log in securely to access personalized features.

- User authentication should be implemented securely, possibly using OAuth or similar protocols.

**2.6 Flash Screen**

- The app should display a splash screen upon launch, providing visual feedback to users while the app loads.

- The splash screen should be visually appealing and consistent with the app's design.

1. **Non-functional Requirements**

**3.1 Performance**

- The app should load quickly and respond promptly to user interactions.

- Music playback should be smooth and uninterrupted.

**3.2 Usability**

- The user interface should be intuitive and easy to navigate.

- The app should provide feedback to users about the actions they perform.

**3.3 Reliability**

- The app should handle errors gracefully and provide meaningful error messages to users.

- The app should maintain session state to resume playback after interruptions such as incoming calls.

**3.4 Compatibility**

- The app should be compatible with a wide range of devices running on Android and iOS platforms.

- The app should adapt to different screen sizes and orientations.

**3.5 Security**

- User data should be encrypted and stored securely.

- Secure communication protocols should be used to protect user data during transmission.

**3.6 Maintainability**

- The app's codebase should be well-organized and documented to facilitate future updates and maintenance.

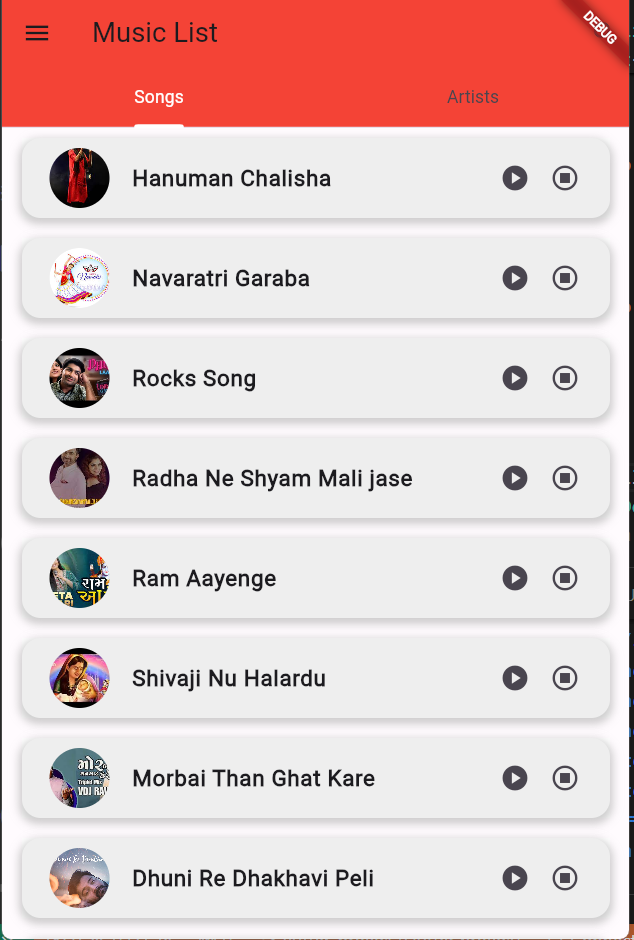
- Code should follow best practices and design patterns to ensure scalability and ease of maintenance.

4 **UI**

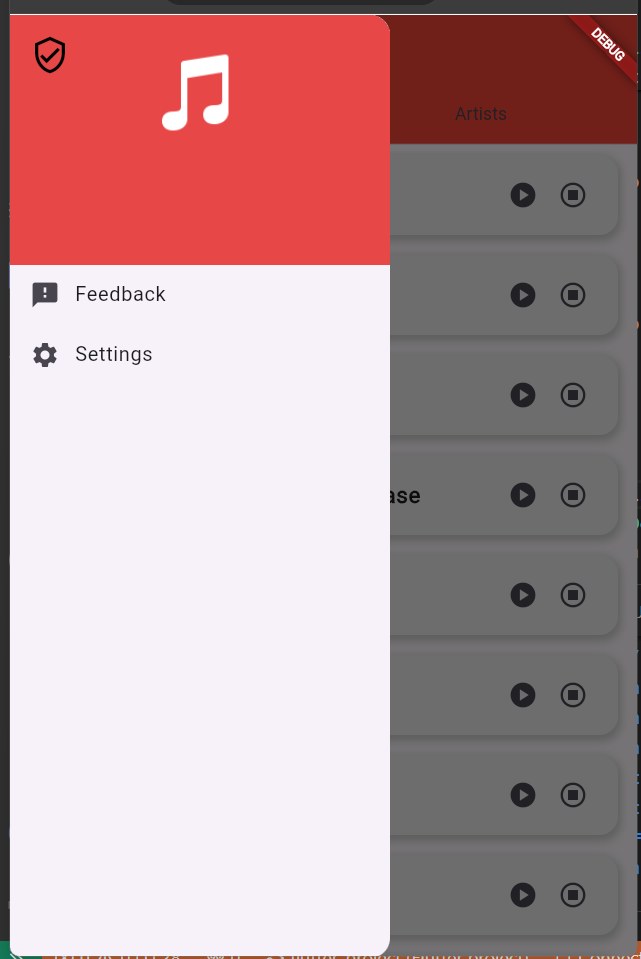
1. **splash screen**

****

1. **Main Screen**



1. **drawer screen**



**5) Hardware & software interfaces**

## As our application is developed using an cross platform framework so we also have an web application version of our application so we don’t require very much hardware configuration to run or use our application.

## This Social Music App Project is developed using Flutter & Dart which is an open source cross platform application development framework developed by Google so as its going to be a cross platform app it will support most of the devices and will be able to connect users using different kind of devices like android phones, iPhone etc. and it can also be used in web browser so we can say that hardware configurations will not be much of an issue for the users but still for the users it is recommended to have latest tech devices to get the best experience some of the recommended software and hardware configurations is as below.

## Hardware configuration for mobile apps (recommended)

## RAM: 2 (minimum), 8 GB (recommended)

## Internet Connection: Good internet connection with Average speed of at least 100 kbps or more recommended.

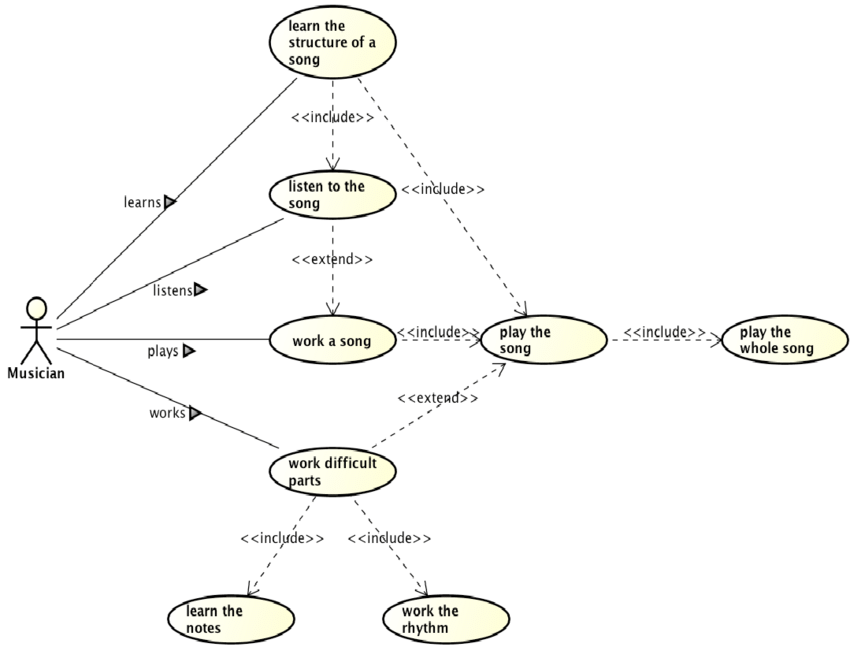
## Software configuration for mobile apps (recommended)

## Android Version: Android 5.0 (API level 21) or higher

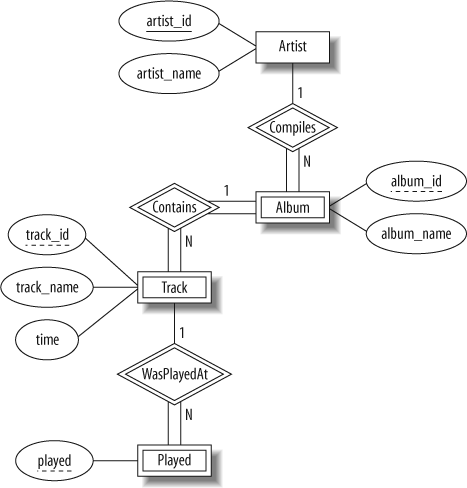
## iOS Version: iOS 11 and later

6) Diagram

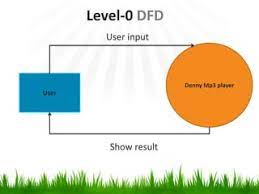
6.1) Use Case Diagram



6.2) ER Diagram



6.3) Data Flow Diagram



DFD 1 LEVEL

