WT MINI PROJECT

for

MShare

Version 1.2 approved

Prepared by Kishan Singh Rathore,

Adit Sadiwala,

Rishabh Sahlot.

28th September 2017.

Table of Contents

Table of Contents ii

Revision History ii

1. Introduction 1

1.1 Purpose 1

1.2 Document Conventions 1

1.3 Intended Audience and Reading Suggestions 1

1.4 Product Scope 1

1.5 References 1

2. Overall Description 2

2.1 Product Perspective 2

2.2 Product Functions 2

2.3 User Classes and Characteristics 2

2.4 Operating Environment 2

2.5 Design and Implementation Constraints 2

2.6 User Documentation 2

2.7 Assumptions and Dependencies 3

3. External Interface Requirements 3

3.1 User Interfaces 3

3.2 Hardware Interfaces 3

3.3 Software Interfaces 3

3.4 Communications Interfaces 3

4. System Features 4

4.1 System Feature 1 4

4.2 System Feature 2 (and so on) 4

5. Other Nonfunctional Requirements 4

5.1 Performance Requirements 4

5.2 Safety Requirements 5

5.3 Security Requirements 5

5.4 Software Quality Attributes 5

5.5 Business Rules 5

6. Other Requirements 5

Appendix A: Glossary 5

Appendix B: Analysis Models 5

Appendix C: To Be Determined List 6

**8**

# Introduction

## Purpose

## Mshare aims to revolutionize the music sharing industry. This website provides free peer-to-peer music sharing, uploading, and listening. It allows user to rate and comment on uploaded music and get recommendations on it. Online streaming and getting a list of top trending users is also possible.

## Document Conventions

The System Requirements Specifications document follows the standard IEEE SRS formatting.

## Intended Audience and Reading Suggestions

This SRS is intended for project managers for the project, the MShare Finance, Marketing and Legal teams, investors and the senior management of this company

This SRS is also intended for Music producers, developers of this project and developers who vouch for the domain of peer-to-peer file sharing.

## Product Scope

This projects aims at unrecognized prodigies who want to showcase their talent and music lovers who yearn to discover new music with fast, lossless and ad-free streaming. With a new song coming out every other day, the total memory of the music available is growing exponentially. With the availability of 4G internet to every economic class of the society many users have switched over to online streaming where one can listen to any song of their own choice, with the most ease.

FUTURE SCOPE

Profits can be earned by the music producers as well investors, through the ‘downloads’ option available in the pro pack at a monthly basis.

## References

Title: Mshare

Author: Kishan Rathore, Adit Sadiwala, Rishabh Sahlot

Location: Dwarkadas J Sanghvi College of Engineering.

IEEE Recommended Practice for Software Requirements Specification.

# Overall Description

## Product Perspective

This project originated with the idea of ‘Soundcloud’ and ‘Spotify’. Though they are already pre-existing, they overflowing with ads and other disturbances and have a reduced bitrate. They don’t have an ad-free environment either.

## Product Functions

reviews......comment

trending.......yes

playlist.....yes

genre based...yes

rating system.......yes

fast streaming...yes

## User Classes and Characteristics

All sections of society.

## Operating Environment

Internet Explorer, Safari and Mozilla Firefox, Google Chrome, and Microsoft edge. The website would be responsive and work on all sizes of phones, tablet, laptops and personal computers.

## Design and Implementation Constraints

In mobile uploads it might require a few permissions. Only pro pack users will be able to download from other users. The website will be available in English Language Only. Memory storage would require your google cloud storage account credentials. User won’t be able to use the application without logging into the first page. The songs which user wants to upload shouldn’t be copy of any other song.

## User Documentation

The MShare application lets you browse, play back songs uploaded to the list as well as you’re your own music library from internal storage. The music you upload is automatically stored in out MShare server. If you download the music it is readily available to play else you can live stream with no speed restrictions.

## Assumptions and Dependencies

The model created is user friendly. Any age group can use this application for listening to music. Application contains some third party template which are not subject to copyright. User shouldn’t misuse the file and use it for personal interest. The project is dependent on external source such as google drive for storing the data. Third party payment gateway is also required. As application provides free downloading it solely relies on pro pack as way of income.

# External Interface Requirements

## User Interfaces

Once your profile is set, upload songs by clicking “Upload” on right side of navigation bar. With With a free account, you can upload up to 10 minutes-worth of tracks.

Once you upload a song, verify the title, add a description to the song. User will be provided link to the song so it can be easily shared. Another cool feature is that user can comment, also like, to a particular track.

One of the best feature about MShare is following other users and let others follow you. This creates friendly environment among community. Don’t worry if you have just started using this application you can select favorite genre, based on that you will be suggested tracks. Also turn on your location to follow and seek into your surroundings interests.

## Hardware Interfaces

MShare doesn’t require any special hardware requirements as such. Though the device should have certain storage space if he desires to download the song on local storage.

## Software Interfaces

Mshare is supported on all operating systems. Though operating system should have any type of browser. The website is integrated with google and facebook api’s for user login. It uses sqlite database for storing data.

## Communications Interfaces

It is integrated with https protocol for user data security and online transaction safety. User can only validate its account after 2 step authentication that includes verifying email id through our system generated mail then entering one time password which will be generated after verifying the link from the mail. As our servers are based on new version of internet protocol that is ipv6 it provides better support for home automation. Communication security is enhanced using this protocol. Https protocol provides end-to-end encryption which is necessary for maintaining data integrity of user.

# System Features

The main System features include Personalized music sharing, Recommendation bases on likes, Preferred Genre, Trending, Comments on each song and album, Email activation. <This template illustrates organizing the functional requirements for the product by system features, the major services provided by the product. You may prefer to organize this section by use case, mode of operation, user class, object class, functional hierarchy, or combinations of these, whatever makes the most logical sense for your product.>

## Personalized music sharing

4.1.1 Description and Priority

This feature allows user to share music on a private portal. User can even like share and comment on the that specific song. Developers have presented the information in systematic format. The priority of this component is rated 8 in this system features.

4.1.2 Stimulus/Response Sequences

User can upload the music by signing up for the Application. Then the user is given a form to add friends as soon as user adds songs he can share music to their friends or just listen to uploaded music. If user has already made an account, user can directly login, hence user is provided with two options either login or signup.

4.1.3 Functional Requirements

Requirement 1: Distinct username, email id, Password.

Requirement 2: Atleast one friend to share music.

## Trending Songs

4.1.1 Description and Priority

This feature allows user to watch the music which are getting more views and likes in last month. Developers have programmed the system in such a way that user gets the trending songs according to their genre.

4.1.2 Stimulus/Response Sequences

There is no such response needed from user. User only needs to select his preferred genre.

4.1.3 Functional Requirements

Requirement 1: Distinct username, email id, Password.

Requirement 2: Select preferred genre.

## Feeds

4.1.1 Description and Priority

This feature allows user to follow other artist. When other artist uploads new song the user can view that in main feed. This is another way in which user can explore new songs.

4.1.2 Stimulus/Response Sequences

User will have to follow more artist to explore new songs. Other user can also follow him/her so that he/she gets more exposure to the music world.

4.1.3 Functional Requirements

Requirement 1: Distinct username, email id, Password.

Requirement 2: Follow other artist.

# Other Nonfunctional Requirements

## Performance Requirements

No such requirements.

## Safety Requirements

User should use 8 words long password. Which contains at least one upper case letter, one lower case letter and one special character. If user forgets password he can only retrieve his password using the email id he/she is using with that account.

## Security Requirements

User should use 8 words long password. Which contains at least one upper case letter, one lower case letter and one special character. If user forgets password he can only retrieve his password using the email id he/she is using with that account. If user activates his account through google+ or Facebook and forgets his/her password, the account will not work till he/she logins into the account.

## Software Quality Attributes

<Specify any additional quality characteristics for the product that will be important to either the customers or the developers. Some to consider are: adaptability, availability, correctness, flexibility, interoperability, maintainability, portability, reliability, reusability, robustness, testability, and usability. Write these to be specific, quantitative, and verifiable when possible. At the least, clarify the relative preferences for various attributes, such as ease of use over ease of learning.>

## Business Rules

Once songs is purchased it cannot be refunded.<List any operating principles about the product, such as which individuals or roles can perform which functions under specific circumstances. These are not functional requirements in themselves, but they may imply certain functional requirements to enforce the rules.>

# Other Requirements

User should have decent internet speed for better user experience.

Appendix A: Glossary

HTML: Hyper Text MarkUp Language

CSS: Cascading Sytle Sheet

Django: Python Framework

Appendix B: Analysis Models













