

SOFTWARE REQUIREMENTS SPECIFICATION



Emertxe Information Technologies (P) Ltd MP3 Tag Reader

VERSION: 0.1

REVISION DATE: 07-06-2014

Version	Date	Changed By	Modifications
0.1	07-06-2014	Emertxe	Initial Draft

Table of Contents

Introduction.....	4
Scope.....	4
Development Environment.....	4
User Interface Requirements.....	4
Functional Requirements.....	6
Design details.....	7
About (mp3) ID3 tag.....	7
ID3v1.....	7
ID3v2.....	8
Before you start.....	8
Coding guidelines.....	9
Testing guidelines.....	9
References.....	9

Introduction

MP3 tag reader is a software which will read and display MP3 (ID3) tag information from MP3 files. The software will be desktop based and not web based.

This documents is the System Requirements Specification (SRS) for MP3 tag reader. The product will be useful to individuals who wish to view and collect mp3 tag data. This project can be extended to implement a tag editor, where-in users can modify mp3 tag information.

Scope

The purpose of this SRS is to completely capture all the requirements pertaining to mp3 tag reader. It will be used for determining the architecture, and preparing design document.

Development Environment

We will be using J2SE platform for developing the application. Java swing can be used for developing the GUI.

User Interface Requirements

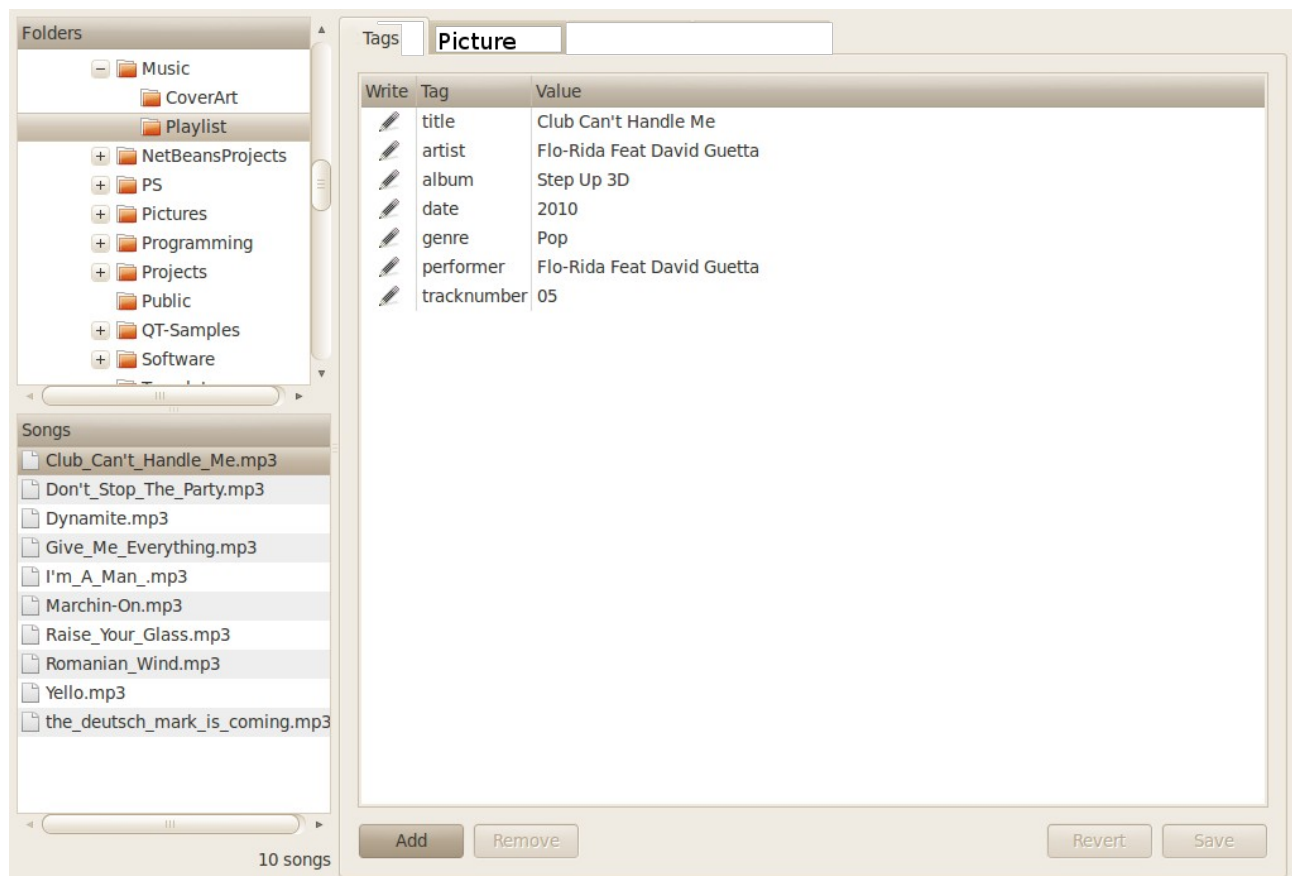


Illustration 1: MP3 tag editor - User Interface

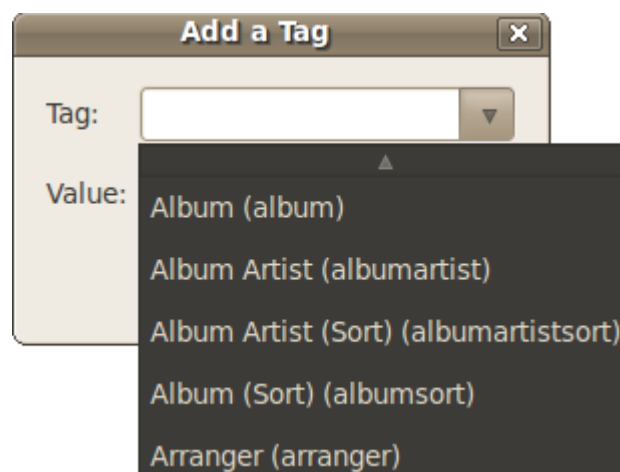
The Main UI has following components:

1. Folder/Directory tree panel
 - This panel should display a tree view for listing directories.
 - User should be able to navigate and select directories by clicking on the directory name.
2. Song/Track list panel
 - The track list view must display files from selected folder in folder tree panel.
 - It must only display mp3 file names
3. Tag Viewer/Editor frame
 - On selection of mp3 track from track list panel, mp3 tag information should be displayed in tag viewer frame.



Illustration 2: Add Tag Pop-up

Pop-up window to add tags to an mp3 file is shown in illustration 2.



*Illustration 3: Drop down list for tag selection
(Add tag pop-up)*

MP3 tags must be listed in add tag pop-up

Functional Requirements

This section list the functional requirements for the application.

Mandatory features:

1. On launch of the application, the main window should be displayed. A mock main window is illustrated in UI requirements section (Illustration 1).
2. On Selection of a folder in Folder panel, mp3 file names must be filtered and displayed in the song list panel.
3. On Selection of song from song list panel, mp3 tag information must be displayed in tag information frame.
4. Only available tags must be displayed in the tag information frame.
5. Application must support ID3v1 and ID3v2 (Support 2.3, need not support 2.4) file formats.
6. If a picture (like cover art) is available in mp3 file, display the picture in Picture tab frame.

Additional features: (Get extra credit)

Implement tag editor

1. In tag information frame, user should be able to edit existing tag values.
2. After editing the tag values, user should be able to save the tag information by clicking on the save button.
3. Add new tag using the add button. A new pop-up should be displayed as shown in illustration 2.
4. It should be possible to delete a selected tag by clicking on delete button.

Design details

Create a simple design document before writing the code for the project. It is important to keep in mind that design should be object oriented and modular. Having modular code helps in allocating different modules/functionality to every team member. Thereby speeding up the implementation. Ensure that every team member make significant contribution to every phase of the project including design and implementation.

To re-emphasise, this is a team project and not an individual project. Please ensure good communication and team spirit. Choosing a team leader help in co-ordination and decision making.

Prepare a class diagram and list the members and functions for the class. Design a flow chart as part of the design document. Also define the data structure which will be used in the design.

For GUI design use Java swing. You are free to use design tools for GUI design.

About (mp3) ID3 tag

ID3 is a metadata container most often used in conjunction with the MP3 audio file format. It allows information such as the title, artist, album, track number, and other information about the file to be stored in the file itself.

There are two unrelated versions of ID3: ID3v1 and ID3v2. You will have to first detect the ID3 version before getting the tag details.

ID3v1

The ID3v1 tag occupies 128 bytes, beginning with the string TAG. The tag was placed at the end of the file (last 128 bytes).



Figure 1: ID3v1 Tag format

Song Title	30 characters
Artist	30 characters
Album	30 characters
Year	4 characters
Comment	30 characters
Genre	1 byte

Table 1: ID3v1 tag elements

ID3v2

ID3v2 tags are of variable size, and usually occur at the start of the file, to aid streaming media. They consist of a number of *frames*, each of which contains a piece of metadata. For example, the *TIT2* frame contains the title, and the *WOAR* frame contains the URL of the artist's website. Frames can be up to 16MB in length, while total tag size is limited to 256MB.

Please refer the ID3.org website for more details on the header, frame format and supported frames.

Before you start

- Understand the project specification – Please ensure that you understand the project specification. Clarify all doubts. Visualize what is the input and output.
- Do a feasibility study – Check if the project is feasible to implement with the allocated time and resources. If you feel certain features could not be implemented within the allocated time, talk to your instructor.
- Gain technology skills – If you need to learn specific technologies, please get prepared on the required topics.

Coding guidelines

Follow coding standards to ensure that code is more readable, re-usable and well commented.

- Use proper naming conventions for classes, variables, functions, data types, file names and directories.
- Do not hard code values.
- Arrange class definition into separate headers and functions in separate .java files.
- Add block comments for files, and functions. Apart from this add comments wherever applicable.

Testing guidelines

In SDLC testing is an important phase of the project. Ensure that there are enough test cases listed. Applicable is of good quality only if it passes all the test cases.

References

1. Wikipedia article on ID3 tag - <http://en.wikipedia.org/wiki/ID3>
2. ID3 tag standard website - <http://id3.org>
3. ID3 tag version 2.3 standard - <http://id3.org/id3v2.3.0>