|  |
| --- |
| ECSE 321 – Introduction to Software Engineering |
| Requirements Specification |
| Mission 1: Client Meeting and Requirements Specification Document |

|  |
| --- |
| Group 1: Samuel Cormier-Iijima (260174995), Simon Foucher (), Stefanos Koskinas (), Bertin LeBlanc (260191026), Amin Mirzaee (260209556)3/3/2009 |

Contents

[1 Introduction 4](#_Toc223802308)

[1.1 Purpose 4](#_Toc223802309)

[1.2 Scope 4](#_Toc223802310)

[1.3 Objectives and Success Criteria 4](#_Toc223802311)

[1.4 Document Conventions 4](#_Toc223802312)

[1.5 References 4](#_Toc223802313)

[2 Description 4](#_Toc223802314)

[2.1 Overview 4](#_Toc223802315)

[2.2 Features 4](#_Toc223802316)

[2.3 Definitions 5](#_Toc223802317)

[2.4 Assumptions 5](#_Toc223802318)

[3 System Requirements 5](#_Toc223802319)

[3.1 Functional Requirements 5](#_Toc223802320)

[3.1.1 Music Player 5](#_Toc223802321)

[3.1.2 Music Library 5](#_Toc223802322)

[3.1.3 Metadata Editor 6](#_Toc223802323)

[3.2 Interface Requirements (Input/Output) 6](#_Toc223802324)

[3.2.1 Main Window 6](#_Toc223802325)

[3.2.2 Metadata Editor Window 6](#_Toc223802326)

[3.2.3 Import Window 7](#_Toc223802327)

[3.2.4 Help Window 7](#_Toc223802328)

[3.2.5 Audio Output 7](#_Toc223802329)

[3.3 Services (TODO) 7](#_Toc223802330)

[3.4 Non-Functional Requirements 7](#_Toc223802331)

[3.4.1 Usability 7](#_Toc223802332)

[3.4.2 Performance 7](#_Toc223802333)

[3.4.3 Reliability 8](#_Toc223802334)

[3.4.4 Security 8](#_Toc223802335)

[3.4.5 Portability 8](#_Toc223802336)

[3.4.6 Legality 8](#_Toc223802337)

[4 Appendix A: Glossary 8](#_Toc223802338)

# Introduction

## Purpose

The objective of this project is to provide a full-featured yet simple-to-use desktop music player, capable of playing audio files stored on the local hard drive as well as over the network.

## Scope

The music player will be used by people interested in a free, small, simple music player that has the capability of playing songs stored locally as well as over a network.

## Objectives and Success Criteria

* Produce a music player capable of playing and organizing audio files of various formats.
* Provide a simple, clean, and easy-to-use interface with playlist and search functionality.
* Support playing music over a network.

## Document Conventions

After the statement of each requirement, an estimate of its importance and difficulty, respectively, on a scale of 1 to 4 will be written in brackets. For example, [3, 4] signifies an importance of 3 and a difficulty of 4.

## References

* A summary of the client interview can be found in *clientmeeting.pdf*.
* An audio transcript of the client interview can be found in *interview.mp3*.

# Description

## Overview

There are many different music players that are available today, ranging from open source command line utilities like mplayer to full-featured applications such as iTunes and Windows Media Player. This document describes the requirements for our music player, named “Staccato”.

Given the wide range of music players currently on the market, there exists is at least one to satisfy almost every plausible need. Of the many music players analyzed by our team, most lacked intuitive methods to share music between on different networks. Staccato will, along with providing all of the features integral to a robust music player, allow for sharing of playlists and music over the internet.

## Features

Staccato aims to be a simple desktop music player. It will provide the user with a graphical interface for playing music files, cataloguing files according to tag fields (e.g. artist, title, and album), browsing available music files, creating and editing playlists, and viewing album cover artwork. It will also provide a tool to connect to other users and share playlists.

## Definitions

***(Music) Player:*** A software system allowing users to listen to digital audio files.

***Audio File / Song:***A file that stores encoded audio data, along with one or more metadata tags. The file may or may not be physically located on the local computer.

***Metadata Tag:*** An attribute associated with an audio file which contains additional information regarding the contents of the file.

***Library:*** The collection of all audio files on the computer which are accessible by the player.

***Playlist:*** A collection of audio files. The playlist may be stored locally and editable by the user or from another computer over a network.

## Assumptions

The client did not specify which specific file formats needed to be handled by the music player, stating only that the “most popular formats” should work. An analysis shows that the most commonly-used audio formats today are:

* WAV (uncompressed)
* AIFF (uncompressed)
* FLAC (open source, lossless)
* WMA (lossy or lossless)
* MP3 (lossy)
* AAC (lossy)
* Ogg Vorbis (open source, lossy)

# System Requirements

## Functional Requirements

### Inputs

#### Audio Output Management

##### Compatibility with audio files containing encoded data in major audio formats

##### Controls to manage the playback state of the active song (start, stop, pause, seek)

#### Audio Filter Manager

##### Master volume control [4, 2]

##### Output level for individual frequency bands (Equalizer) [4, 2]

#### Metadata Manager

##### Controls to create, update, and delete metadata associated with an audio file [4, 3]

##### Tagging custom keywords to audio files

##### User preference ratings for audio files

##### Custom artwork associated with audio files

##### Toggle automatic management of metadata

#### Playlist Manager

##### Controls to add, remove, and duplicate songs to/from playlists

##### Sorting and prioritization of tracks within playlist

##### Search terms and options to query for tracks within playlist

#### Music Library Manager

##### Importing audio files into user library via Local/Network locations

##### Selection of default music library

#### Network Manager

##### Network locations to browse for audio files

##### Specification of remote files to be imported to music library

##### Sharing and publishing partial/complete portions of library and playlists

##### Toggle automatic synchronization of music library and playlists with remote repositories

#### Software Assistance

##### Display and search through product documentation

### Outputs

#### Audio Output

##### Clean and undistorted playback through speakers/headphones

##### Properly rendered sound levels and audio filters

#### Audio Filter Manager

##### Accurate display of current volume levels

##### Feedback regarding currently enabled/available audio filters

#### Music Player State

##### Accurate and intuitive indication of active track/playlist

##### Indication of current playback state (playing/paused, time left)

##### Title, Album, and Artist information pertaining to active track

##### Album artwork and/or visualizations

#### Metadata Manager

##### Navigable of all songs matching indicated query

#### Playlist Manager

##### Navigable list of all tracks in the selected playlist

##### Aggregate playlist information (total time, average track length, number of songs)

##### Current ordering and track prioritization within playlist

#### Music Library Manager

##### Queryable list of all available audio files in the active music library

##### Local/Network location, file type, and encoding information of all files in library [4, 3]

##### Playback statistics for each track in library

#### Network Manager

##### Availability status and aggregate information per available remote repository

##### List of currently available audio tracks within selected location

##### Available metadata and tags for files within the selected location

##### Sortable, Queryable list of added network music repositories

#### Software Assistance

##### Help information and user documentation

### Services

#### Network Manager

##### Network crawler which maintains a list of shared locations/files accessible on the local network

##### Configurable synchronization service tracking select remote repositories

#### Metadata Manager

##### Automatic retrieval/updating of song metadata

##### Tracking of playback statistics for songs/playlists

### Initialization

#### Database

##### Opening and loading information from database corresponding to default music library

##### Creating and initializing a new database to store library information upon first execution

#### Services

##### Configuring and initializing audio filters, network utilities, and music library caches to ensure smooth operation.

##### Scheduling necessary background tasks with the host environment’s process scheduler

### User Interface

#### Main Window

##### The main window shall provide a menu bar, supporting: [4, 1]

###### Importing audio files or directories using the *Import Window* (see 3.2.3) [4, 1]

###### Exporting the contents of the library, e.g. for backup purposes [3, 1]

###### Editing the *metadata tags* of the currently selected songs in the *playlist viewer* (see 3.2.1.4) through the *metadata editor window* (see 3.2.2) [4, 1]

###### Exiting the player [4, 1]

##### The main window shall provide a toolbar providing support for: [4, 1]

###### Play/Pause [4, 1]

###### Stop [4, 1]

###### Toggle shuffle [4, 1]

###### Toggle repeat (None, Song, All) [4, 2]

###### Volume control [4, 2]

###### Current song location and seeking [4, 3]

###### Search [4, 2]

##### The main window shall provide a *playlist picker*, allowing the user to display either the contents of the library or a playlist in the *playlist viewer* (see 3.2.1.4). [4, 3]

##### The main window shall provide a *playlist viewer*, showing a list of songs sortable by their *metadata tags*. [4, 3]

###### Double-clicking on a song in the playlist viewer shall cause that song to start playing. [4, 1]

###### Dragging songs from the playlist viewer to playlists in the playlist picker (see 3.2.1.3) shall add them to the playlist onto which they were dropped. [4, 3]

#### Metadata Editor Window

##### The metadata editor window shall provide input boxes to allow editing the metadata tags for the currently selected songs. [4, 3]

###### Track title [4, 1]

###### Artist [4, 1]

###### Album [4, 1]

###### Album artist [4, 1]

###### Year [4, 1]

###### Track number [4, 1]

###### Genre [4, 1]

###### Rating [4, 1]

###### Custom tags [4, 3]

##### The metadata editor shall provide a button for saving the edited fields back to the audio file. [3, 1]

##### The metadata editor shall provide a button or discarding and reverting values back to old ones. [3, 1]

#### Import Window

##### The import window shall allow the user to select either an audio file or directory on the local hard drive for importing.

###### Importing an audio file shall cause that file to be added to the library.

###### Importing a directory will cause all audio files contained recursively within it to be added to the library.

#### Help Window

##### The help window shall provide a way of viewing documentation about the player’s usage. [4, 1]

## Non-Functional Requirements

### Usability

#### The user interface must be sufficiently similar to other available music players to allow easy and intuitive usage. [4, 2]

#### The player must be packaged and distributed with documentation regarding its usage. (See 3.2.4.1) [4, 1]

### Performance

#### The player must be able to handle up to a thousand music files in a playlist in under a second. [2, 2]

### Reliability

#### In the case of failure, the user’s music files and library information must not be corrupted. [4, 3]

### Security

#### It must not be possible for users on the network to modify or in any way access resources on the computer through the networking functionality, unless explicitly enabled by the user. [4, 3]

### Portability

#### The player must be able to run on any modern computer desktop environment. [2, 2]

### Legality

#### The player must conform to all Canadian laws and regulations regarding computer software. [4, 1]

# Appendix A: Glossary