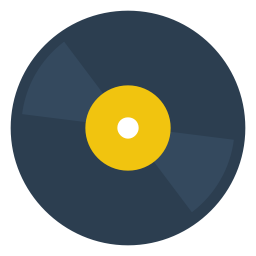
Richard Jones | Computer Science - OCR | March 20, 2017

Music Center Program

A LEVEL COMPUTER SCIENCE cOURSEWORK



# **3.1 – Analysis of the Problem**

## 3.1.1 – Problem Identification

### Proposal

The proposal for this piece of Coursework is to make a music player program that will allow the end user to efficiently navigate and play their music library using an integrated music playing system.

The Program will let the User import information about their music files (.mp3 exclusively) to the Music Player’s database by reading the ID3 tags of the files. As well as playing local music, the program will also use an online music database to recommend other songs from similar genres and artists.

The program will use the Python programming language, SQLite for database management and PyQT Designer to create the visual user-interface.

### Why this Proposal is amenable to a computational approach

This proposal aims to solve the problem of playing music and storing information about said music. Firstly, the program will solve the problem of playing music by using libraries in the programming language to output sound from the .mp3 files. Secondly, the program will be able to solve the storage of .mp3 data by using a database for them. Allowing the program to efficiently collect and store the information.

### Why this Proposal is a suitable Solution

Music players have a large target market as there will always be a demand for programs that play music on devices.

However, music playing apps are no new innovation to the world. They have been almost perfected by companies such as Windows (with Windows Media Player) and Apple (with ITunes). My program hopes to produce a more lightweight solution to the choice of music players. Meaning that it will be less tasking on devices when compared to other programs such as Windows Media Player and ITunes.

## 3.1.2 – Stakeholders

### End Users

My End User will be anyone who wants to play music on their device(s). However, it will specify in catering for people who are conscious of the effect of these programs on the performance of their machines. For example, people with machines that are of a lower-tier technology or people who do not want a music player program detracting from the performance of their device.

Several members of my tutor group have expressed concerns while using other music playing programs because they have found their machines slowing down. They also often don’t use or need many of the features they offer. This solution would be suitable for them as it is unlikely to slow down their computers and will contain a simple and easy to use interface. Later on, many of my peers will be interviewed about this program.

My program will be specifically judged by three members of my tutor group. Any updates that I make will be with their permission and based on their feedback. They will be in control of the design and how they want the program to look.

## 3.1.3 – Research

### Similar Solutions

The main competitors for the music playing solutions are Microsoft’s Windows Media Player and Apple’s ITunes.

Windows Media Player:

* Windows Media Player is programmed in Visual Basic and is currently in its 12th release.
* The program provides playback for several music formats such as .asf, .asx, .wax, .wm, .wma, .wmd, .wmv, .wvx, .wmp, and .wmx files.
* The program also provides Video playback, CD Ripping and Burning facilities and allows users to create playlists.
* Windows Media Player is only available on the Windows Desktop Platform, with the OSX and Portable PC versions being discontinued.

ITunes:

* ITunes is programmed in C++ and is also in its 12th Release.
* The program provides playback for several music formats such as .mp3, .aiff and .wav.
* ITunes also provides Video Playback, CD ripping and burning facilities, allows users to create playlists and provides an online store where the users can buy digital music.
* Itunes is currently available on Windows, OSX and iOS operating systems.

Both of these solutions are very effective at providing media playing features well, making them very efficient as large scale projects (to provide for a large target market). However, many of these features are not used by the vast majority of people. I propose to make a program that caters for a smaller audience but provide a more efficient solution to playing music on their devices.

### Interviews

2 students from my peer group were interviewed about their ideas for the solution and also what they would like to see from my solution

|  |  |
| --- | --- |
| Interview 01 |  |
| Name | Sam Barrett |
| Age | 17 |
| Vocation | Full Time Student (Year 13) |
| Devices Used | Desktop and Laptop (Windows 10), Phone (Android) |
| Music players Used | Windows Media Player, Spotify |
| Signature |  |

1. **Why do you use your chosen music players?**

“I use Windows Media Player because it is the fastest media player I know. I do not have a need for ITunes as I don’t have an Apple Device, so I’m fine to use Windows Media Player. However, my actual music library is very small and I mainly use Spotify, which seems to be the best solution for playing music.”

1. **What are the limitations of your chosen music players?**

“I feel that there are many features to Windows Media Player that I have no need for. Most people I know really only use media players to play music and perhaps play video. I would feel more comfortable using a program that just handles music. Spotify is a very effective solution to playing streamed music. However, it is a subscription service and requires constant money to keep running.”

**Interview 02**

|  |  |
| --- | --- |
| Name | Daniel Hawes |
| Age | 18 |
| Vocation | Full Time Student (Year 13) |
| Devices Used | Laptop (Windows Vista), Phone (Nokia 3310) |
| Music Players Used | Windows Media Player |
| Signature |  |

1. **Why do you use your chosen media players?**

**“**I use Windows Media Player because it is the default media player. I’m not too fussed about how I play music on my computer as long as it works.”

1. **What are the limitations of your chosen music players?**

“My laptop isn’t very fast and has trouble processing many applications. It’s hard to have Media Player in the background on my machine because it slows it down and makes it hard to run any other applications.”

**Conclusion from Candidate 01 and 02**

The interviews from these two candidates have highlighted a few points that can contribute to my solution.

‘Candidate 01’ mentioned that many of Windows Media Player’s features are not appropriate to his needs. Many media players, including Windows Media Player, are problem solutions for large-scale audiences (worldwide) and must cater for as many needs as possible. However, on a smaller scale, this number of features can be seen as inefficient. This supports my solution as it focuses only on provided music playback and music library management and therefore making it a better solution for this smaller target market.

‘Candidate 02’ said that the media player he uses (Windows Media Player), slows down his PC too much. Many of the large competitors in music playing solutions also have this problem, specifically ITunes. My program will be written in python, which is not a very demanding language and is unlikely to significantly slow down any current machines.

However, ‘Candidate 01’ did point out a limitation in my solution. My program cannot stream music from the internet and it is unlikely that this could be a possible objective. In order to legally stream music, you need to obtain certain permissions from the artists in general and pay royalties to them. This would not be a good decision for a small-scale project such as this as it would be too tasking to maintain as well as too costly.

As a whole, these two interview have shown that there are definitely some existing stakeholders for my solution. These include people who desire lightweight solutions to playing music as well as wanting a small and simple number of features in their music players.

## 3.1.4 Solution

### Objectives

**Importing Files:**

* Allow the user to input the location of a directory and then scan said directory for .mp3 files.
* Read the ID3 tags of each .mp3 file found and input the data from the following fields into the database:
  + Song Name
  + Album Name
  + Artist Name
  + Genre
  + Track Number
  + Year of Release
* Calculate the track length and location of each file and append it to the database also.

**User Interface:**

* Provide the User with an organized view of their library. E.g. Alphabetical/Chronological.
* Provide buttons that allow the user to change the view of their library. E.g. View from albums, artists, songs, years.
* Provide an interactive User interface so that the User can navigate the library, e.g. Clicking on an album name would display all of the songs in said album.
* Allow the user to search for a specific song using a search box.

**Playing Music:**

* Allow the user to play a song by selecting the song name in the library view.
* Allow the user to pause and resume a song with the help of a button.
* Allow the user to skip to the next/previous song in an album/library using the skip forward and skip backward buttons.
* Display the details of the song that is currently being played.
* Have a progress slider that will inform the user how much of the song they have listened to at a certain point.

Recommendations:

* Provide a section that the user can navigate to with recommendations for music that is not in their library, selected by similar genres and artists that the user already has in the library.
* Provide a link directing the user to a site where the User can pay and download the songs.

Settings:

* Provide a settings menu so that users can change where they want their files to be imported from.

Accounts:

* Allow different users to hold different accounts.
* Let each account hold a password, which is hashed and held in the database.
* Let users to personalize their music library for separate accounts. E.g. have songs A, B and C in User 01’s library and songs D and E in User 02’s library.

Data Manipulation:

* Store how many times a song is played.
* Store how many times an artist is played.
* Form a playlist of most played songs.
* Form ‘mood’ playlists based on most played songs in specific genres.

# **3.2 – Design of the Solution**

### 3.2.1 Decomposition of Problem

### Program Design Mockup:

During my brief discussion and interview with the stakeholders for this program, I drew up this initial design of the program in order to cater to their needs. My stakeholders were happy with this draft.

**Login Screen:**

Please Login

Login

Create Account

Password

Username

This is the first screen that will be shown when the program is run. The screen will contain two input text boxes that will allow the user to input their user details, the two boxes will be titled with two labels. There will also be two buttons: The ‘Login’ button will allow the user to check their details and gain access to the rest of the program while the ‘Create Account’ button will allow the user to create a new account. The ‘Login’ button will take the user to the ‘Main Program’ screen while the ‘Create Account’ page will take the user to the ‘Create Account’ screen. Finally, there is a ‘Please Login’ Message, if the information the user enters is not found, the message will inform the user of what was wrong.

**Create Account Screen:**

Password

Please Enter Your Details

Create Account

Username

This screen will be where the user can create a new account. The screen has two boxes that are titled with ‘Username’ and ‘Password’, this is where the user will input their desired username and password. There will also be a ‘Create Account’ button where the user can submit their details to be checked/validated by the program. If the validation is successful, the user will be taken back to the login screen. If the validation is unsuccessful, a message will be displayed on the ‘Enter Your Details’ title.

**Main Program Screen:**

Artists

TABLE

Settings

Pause/Play

Skip Backward

Skip Forward

Now Playing

Albums

Genre

Playlists

Songs

This

T

This will be the main screen for the program and will be used for navigating the library and playing the music. The screen is split up into three sections which organize the components of the screen by function. The top right section of the screen is the ‘Music Library’ table, this is where the database files from the Music Library database will be displayed. The top left section