Enhanced Digital Music Jukebox

Daniel Semaya[1]
Princeton University
Department of Computer Science

There are many limitations with current jukebox technologies. These limitations apply to both traditional jukebox units, which use CDs or other media, and more modern software applications that call themselves "mp3 jukebox" applications. For this project we will focus on the user interface limitations of both kinds of jukeboxes. The limitation of most mp3 jukebox systems is that they fail to act as a traditional jukebox in many ways. With this project we will attempt to create a complete jukebox system which functions like a traditional jukebox and advances upon traditional jukebox usability features. The system will use touch screen input, be highly graphical and tested extensively in a real world environment.

1. Introduction

Music jukeboxes have often been used in bars and restaurants as an interactive way of playing music and providing entertainment. While jukeboxes are still used in the same locations, there have been very little advancement in the user interface and feature set of these systems. This project attempts to create an interactive jukebox system using a computer that plays digital music.

The problem this system attempts to solve is the lack of specialized computer jukebox systems. Currently there are no other systems out there that meet the needs of the intended environment for this jukebox system. The goal of this project was to create a fully functional and complete jukebox system to be used by real people. The steps taken to complete the system as well as the problems discovered will be described in detail.

1.1 Background

[1] dsemaya@princeton.edu

In January of 2003 I was approached by the officers of the Colonial Club eating club in Princeton about creating a jukebox system to replace the broken CD jukebox in the basement of the club. I originally put off the idea since this would be a significant project that I did not have time for. However in September it occurred to me that this sort of jukebox project presents very interesting problems. The problems include human interface problems and system issues. I decided to take up this project as independent work for the semester to attempt to solve these problems.

1.2 Traditional Jukeboxes

We shall a define traditional jukebox to be a physical jukebox unit that plays music from a physical media form such as records, tapes, or compact discs. These units typically cost several thousand dollars and are manufactured by brands such as Wurlitzer. The user interface to these systems is most often very limited. Many of these units have song options listed on paper labels, using only text, while more advanced systems have a catalog-like browsing system that shows album cover art as well. Typically a user chooses a song selection by a four digit number code consisting of two digits to signify the number of the album and two digits for the track number. This interface is less than ideal but the nature of these systems requires an interface of this sort.

Other drawbacks of traditional jukebox systems are that they are very large and very costly. Changing or updating the music catalog is also a very cumbersome and time-consuming process and often requires printing new labels and inserts.

1.3 Digital Jukeboxes

Today there are many products that claim to be “digital jukeboxes.” These products come in the form of both software applications for general-purpose computers and hardware units, which are typically portable and battery operated.

Many large software vendors produce software jukebox applications. Today’s jukebox software includes Windows Media Player by Microsoft, Winamp by AOL-owned Nullsoft, iTunes by Apple, RealJukebox by RealNetworks and Musicmatch jukebox. Most of this software is available for free with some programs, notably Musicmatch and Windows Media Player, requiring the purchase of additional packages for full functionality. While most of these programs use the term “jukebox” to describe their functions, most do not function very well at all as an actual jukebox system. The applications often give users too much control since users are not restricted to adding songs to the end of the currently playing queue. The inherent nature of using a computer with a mouse and keyboard is not ideal for a public setting.

Most hardware “jukebox” systems, such as the Apple iPod or the Nomad Zen Jukebox, are intended to be personal portable music players and not jukeboxes in the sense for which we have defined. These units are typically an extension of the software players and offer similar features. They are designed to be used by one person and are not useful for the purpose of a traditional jukebox.

The problem with both jukeboxes is that technology has yet to be applied to remedying the problems of the traditional jukebox system. Music player applications and hardware are replacements for the home stereo or personal Walkman but are not designed to function as a replacement for a traditional jukebox. The goal of this project is to create a digital jukebox with an advanced interface and enhanced feature set that would only be possible on a computer-based system.

2. My Approach

My Approach was to create a new jukebox system. The system was intended to have two main components. The first component is a highly graphical, simple song selection, superior to current jukeboxes for this purpose.

The second component was intended to be intelligent song selection; the jukebox will know when and why to play certain songs, based on certain criteria, and use discretion over what songs to play. This portion of the project changed dramatically after the first part of the jukebox was installed and operational.

2.1 Current Technology

Aside from the traditional and digital jukeboxes available there have been a handful of attempts at a jukebox system of this kind. A Windows software application, Touchtone Audio System from Riptide Innovations [3], acts as a graphical interface to Winamp and is designed to be used with a touch screen. The application provides a method of browsing music on the hard disk and adding files to a playlist. The interface contains no graphics and is less than ideal since it appears cluttered since the text is displayed very close together. This application includes directions on preparing a digital jukebox system but falls short of preparing users for the real world issues of creating such a system.

The iTunes music jukebox application contains a feature called "Smart Playlists" that allows for some intelligent automation [1]. The application can automatically create playlists containing music based on use-specified criteria. This system does not actually help the user choose what songs to play and the user must create the rules to decide which songs to put in the playlists.

2.2 User Interface

I considered a number of possibilities for the user interface of this jukebox system. The ideal display would be highly graphical, using large text and graphics that are easily identifiable by the user. The ideal method for input would be a touch-screen display, which would allow users to select songs by physically touching their selections on the screen. Other possibilities include the use of a keypad or joystick. The decision was made to use a touch screen because it would be easier to implement with a web-based interface and would provide the best user experience although a joystick may have been a less expensive alternative.

2.3. Limitations and problems

This project was bound by a limited budget. The standard funding provided by the Princeton University School of Engineering to undergraduate students doing single-semester independent work projects is $500 however this system required equipment well beyond that limit. Thanks to additional funding by the SEAS and equipment donations by the Colonial Club of Princeton this project could be funded adequately, albeit on a tight budget. Due to the limited budget a number of compromises had to be made.

The touch screen chosen was a glass overlay that would hang in front of a CRT display. This is the most cost effective solution because Colonial donated a CRT display. Ideally an integrated LCD touch screen would be used. This CRT and touch screen overlay combination had a number of drawbacks. First, the CRT sat on a shelf below the sound amplifier. The amplifier caused waves in the CRT screen. This could have been avoided by using an LCD display. Another problem was that the touch screen glass could not sit directly on the surface of the CRT and was therefore hung several millimeters in front of it. This meant that depending on what angle a user looks at the screen the user might touch the screen incorrectly. This could have been avoided by using a screen with an integrated touch screen instead of using a touch screen overlay.

The computer chosen was a 400 Mhz PowerMac G4 tower, however a faster Mac would have been preferred. Users often noticed that the interface appeared sluggish at times due to the slow computer with limited RAM. Using a faster Mac would have avoided this issue, however budget constraints made that impossible.

There were also a number of problems with the touch screen and driver software. Originally I had planned on purchasing third-party driver software for the touch screen since the manufacturer of the touch screen, Keytec, did not claim Mac OS X support on their web site. When I received the touch screen I discovered a CD with Mac OS X drivers included but the disc itself was empty. As I waited for a replacement disc I attempted to use a demonstration version of the third party software from a company in the UK called Touch-Base [4]. The Touch-Base drivers had many problems and included very complex configuration settings. When the Keytec drivers finally arrived they worked much better but contained no configuration options. Therefore the sensitivity and other aspects of the touch screen do not work perfectly.

2.4 Installation

The test environment for the jukebox is the tap room in the basement of the Colonial Club, a Princeton eating club on Prospect Avenue. This is a bar-like environment. The jukebox has been used regularly for major party events for over 4 weeks, by hundreds of students. There have been over 2000 song requests as of this writing.

The system was installed in a location previously occupied by a traditional jukebox system, which broke over a year ago. Needless to say, the members of Colonial club were anticipating a replacement jukebox. The only exposed part of the jukebox that is the touch screen, which is in the wall. The computer itself sits in a cabinet underneath the screen. The CRT sits on a behind the wall. The system is connected to the same audio amplifier and speaker system that the previous jukebox used. This audio system was in poor condition and resulted in poor quality audio playback but this could not be avoided. Most listeners did not seem to notice in the crowded noisy setting.

Installation required a live Internet connection to the computer. The process of running Ethernet cables required two days of work and unfortunately took down the club chef's Internet connection for several hours.

Club members assisted with the creation of the wooden housing to mount the touch screen monitor in the wall. The housing required two revisions to fit the touch screen adequately. The club plans a future revision to improve the aesthetics of the design. The housing is designed to prevent users from causing any damage to the jukebox and it hides all parts of the jukebox from the user other than the screen itself.

3. Interface Design

The interface to the jukebox is web-based. The computer is running a full screen kiosk browser called Saft, which is a plug-in for Apple's Safari web browser. The interface and browsing code is written in PHP and is generated on the fly based on the music library files existing at the time. The text on the screen is white on a black background for maximum readability and is smoothed using Apple's Quartz technology, which is native to Mac OS X and is used in the Safari browser. The display font chosen is Arial and the font size is 25 point. This is ideal because it presents large and clean-looking text to the user. One advantage of the web-based interface is that these font styles were easily experimented with using style sheets until the proper style was decided upon.

Most text is double spaced whenever possible so that it is not too close to any other text and it is therefore easier for the user to select the correct text. However there were still issues where users would accidentally select the wrong item. This is partially due to the problem of the touch screen glass resting several millimeters in front of the screen. Taller users would be looking down at the screen from a high angle and would touch the screen off-target. Shorter users would have similar problems. Another reason that users would accidentally select the wrong item is that some users mistakenly thought to drag the mouse pointer around the screen instead of just touching the screen. Apparently since the users saw a mouse pointer they assumed that it should be dragged like a computer mouse. The current workaround solution is to disable the mouse pointer on the screen entirely. Unfortunately this made debugging and administering the system difficult since the mouse pointer is invisible.

3.1 Artist Selection

The user is first presented with a list of all available artists on the system. This originally was one screen but as more and more artists were added to the system a paged browsing system had to be devised. Currently twenty artists are listed per page and users can click back and forth between artist listing screens.

This page-based approach presented a new problem. The system had a time-out feature so that if nothing is pressed for 60 seconds the system displays the starting page again. This page is the first twenty artists in alphabetical order, which typically would show ‘N Sync (the apostrophe is sorted to come first in PHP’s sorting algorithm) to Coldplay. A problem then arises for users who know exactly which artist they are looking for, and that artist happens to be towards the end of the alphabet (U2, ZZ Top, etc.). A user would have to press “show more artists” several times to reach the artist that they are looking for. This can be resolved by including an alphabetical list at the bottom of each screen that brings the user to a list of twenty artists starting with the letter of the alphabet selected.

3.2 Album Selection

After selecting an artist the user is presented with a list of albums available by the selected artist. The albums are shown with small versions of the album’s cover artwork (128 x 128 pixels). These thumbnails serve as visual cues for users to distinguish between the different albums since with many albums the cover artwork is more recognizable than the album title. The user also has the ability to go back to the artist listing from this screen.

For a brief period the system would automatically bring the user to the track-listing page for the album if there were only one album by the selected artist available. This feature was removed due to the confusion that it seemed to cause for users. This feature would cause inconsistency in how different artists were handled and therefore was taken out.

3.3 Song Selection

After selecting an album, the user is presented with a screen showing a larger version of the album artwork and a track listing for the album. A user can select a track by touching the track name. The track listing raised some interesting problems.

First there is the problem of double albums and boxed sets of three or four albums. These “greatest hits” collections of classic artists proved to be among the most popular collections in the system however displaying the track listing for multiple discs is a tricky concept. There is the option of first presenting the user with a list of the three discs of which the user could choose and then choose the track from that disc. That seemed to be a bad idea since most users did not necessarily know what disc of the set a particular song would be on. Instead I opted to show the track listing for the first disc of the set on the first screen with the option for the user to get to any other disc of the set from that same screen. Each disc’s screen would have the option to view any other disc’s screen, therefore in the case of the Led Zeppelin four disc boxed set the user would first be presented with the track listing for the first disc but could navigate to any of the other three discs immediately. This concept appears to be the best for the purposes of this jukebox system. Users appear to understand the concept easily.

When a song selection is made the user is presented with a screen that confirms the user's choice. Occasionally a user accidentally presses a song unintentionally and then goes back and selects the correct song. (Some of the reasons for incorrect song selection are described elsewhere in this paper.) This leads to two songs in the queue by the same artist and on the same album and contradicts the diversified music strategy of this system. The concept of preventing consecutive selections by the same artist or album (described elsewhere in this paper) would not solve this problem adequately since the first song that was entered was not the intended song. Instead the adequate solution was to add a "Cancel" option to the confirmation screen. The user has five seconds to press cancel before the song entered into the queue.

The confirmation screen currently displays text that claims the song selection will play shortly. To be blunt, this is often a blatant lie. The truth of the matter is that the jukebox can often have well over 200 requests in the queue and many song selections will never be heard. If a user sees that their song selection will not play for several hours (or days) the user will most likely leave the party. By "lying" to the users more users are enticed to stay and wait for their music to play. While this can be seen as poor user interface design this was a conscious decision and is adherent to the goals of a jukebox system, which is to be an attraction and a form of entertainment at the party.

4. Music Acquisition

The process of building the music library for this system was much more time consuming than anticipated. The vast majority of music for the jukebox was obtained by ripping audio CDs in iTunes on the PowerMac itself. It soon became apparent that ripping one CD at a time on the PowerMac would be far too time consuming. To help speed up the process, the Colonial Club's computer lab of 10 PCs were used to rip music simultaneously using the newly released iTunes for Windows. While no scientific measurement was taken to calculate the amount of the time saved it seems to be considerable.

All music added to the library was properly tagged. iTunes automatically labeled and numbered most tracks properly but many albums needed to be edited manually. Additionally a select group of albums were not ripped using iTunes and so the ID3 information had to be entered manually in iTunes as well. All of this music organization was very time consuming.

The intended goal of the music organization was to have perfectly tagged and organized music files as well cover artwork for all albums. The best source for high quality cover artwork graphics on the Internet is walmart.com since the site has the highest resolution scans of album cover artwork of the online music retailers. Amazon.com has a web service API that allows for the automated downloading of cover artwork from Amazon.com, which could have been used to automate the process but in the interest of quality walmart.com was chosen and all artwork was downloaded and renamed manually.

Choosing walmart.com as the artwork source had an additional unforeseen drawback. Walmart does not sell many albums. These albums are typically hip-hop albums with explicit language including CDs by Eminem. Artwork for these albums had to be obtained elsewhere.

By the end of the test period the jukebox contained 11.54 GB of music, which was163 albums by 92 artists.

5. Player Backend

The player backend of the system is the Apple iTunes music jukebox application. This application was chosen due to its enhanced music organization features. The iTunes application keeps music files organized in such a way that it is easy for the PHP code to understand the music library file structure and interpret the library. iTunes also provides enhanced music organization features including support for cover artwork, playback counting, song rating and information about the playback dates of each track, the date the track was added to the music library and much more. These features appeared to be extremely useful for our purposes. Additionally, the iTunes application is "scriptable" using Apple's AppleScript language. AppleScript can be integrated into other languages including shell scripts and PHP using the "osascript" tool provided by Apple.

One difficulty with integrating the PHP front end with the iTunes player was problems with user permissions. Apple has designated that the only user that can control the iTunes player is the user that has launched the iTunes application. Unfortunately the Apache web server is installed on Mac OS X in such a way so that the apache user cannot log into the computer graphically to run applications. This presented a problem since the PHP script did not have the permissions to talk to iTunes via AppleScript. This was resolved by passing information through files and using piping and the "tail –f" command. It was determined that the delay time in using text files was small enough for this to be practical.

The iTunes application can create and maintain "playlists" which are lists of songs to be played in order. When a user selects a song on the jukebox the song gets added to a special playlist. If there is currently no music playing then the track is played immediately, and if there is music currently playing then the song gets added to the end of the playlist. The iTunes application has a cross fade feature that automatically cross fades between consecutive songs in a playlist. This makes for a smoother listening experience and eliminates pauses in the music. The application also has a "sound check" feature that normalizes the volume of each track so that all music played is the same volume. This feature has also helped to create a smoother listening experience. However there were still problems with music volume.

6. The Volume Problem

One of the most significant unforeseen problems has to do with the relative volume of the music. On the first night of usage it became apparent that the volume of the music needed to change dramatically depending on the number of people in the room. It was not uncommon for the music at a particular volume level to sound excessively loud when there was a small crowd in the room and yet be inaudible when there was a large crowd. The jukebox needed a way of knowing how many people were in the room.

6.1 Noise Sensitivity

Several methods were considered to allow the jukebox to know the number of people in the room at any given time. The first potential method was noise sensitivity. It was thought that if the jukebox could measure the volume of the ambient noise in the room it could determine how many people were in the room and therefore what volume level to play music. The problem with this concept is that the jukebox is already playing very loud music that is contributing to the ambient noise in the room. It would be necessary to remove the music from the room noise to interpret the volume of the room.

One potential way to do this would be to use a microphone to input the current sound in the room and mix that with the inverse wave of the music that is currently playing out of the jukebox. This would have the effect of canceling out the music and leaving only the noise. This is similar to how noise-canceling headphones work. Ironically, in this case the idea is to remove the music in order to better hear the noise while most often the goal of this sort of activity is to remove the noise in order to hear the music better.

This method was deemed to be too difficult to accomplish for a number of reasons. First of all the one PowerMac was already heavily burdened already and the added task of heavy audio processing seemed to be too difficult. Additionally there would be a delay between the music that the jukebox is playing and the sound that the jukebox is receiving. This delay time would need to be compensated for by delaying the inverse wave of the original music before mixing it with the received sound. Since the computer’s behavior is often unpredictable depending on what task it is currently doing the delay time could therefore be unpredictable. This could be resolved by either using a separate computer to do the audio processing or to use one faster computer, however the budget for this project was approved for one PowerMac so this was not possible.

6.2 Temperature Sensitivity

A second potential method for determining the number of people in the room was temperature sensitivity. Since humans give off body heat, it was thought that if the temperature in the room could be accurately monitored then it would be possible to determine the number of people in the room based on the heat given off by those people. The problem with this system is that the windows of the room are often opened when people are smoking and therefore the temperature readings would not accurately represent the number of people in the room. On a cold winter night the open windows have a strong effect on the temperature of the room. Therefore temperature sensitivity would not be a viable option.

6.3 Motion Sensing

Motion sensing is the third potential method of discerning the number of people in the room at any given time. The goal would be to count the number of people who walk into the room through the room's one main door. This could be accomplished using an infrared sensor in the doorway. This inevitably becomes more complicated because a single sensor would not be able to determine whether or not a person is walking into or out of the room and therefore a second sensor would be needed. Then software would have to be written to address both sensors and discern motion directions between the two.

This type of work falls beyond the scope and budget of this project, however it seems that it would be a reliable method if it were to be implemented.

6.4 Activity Monitoring

Activity monitoring is a method suggested by Professor Brian Kernighan. This method would monitor the actual usage patterns of the jukebox to decide whether or not there are people in the room, and how many there are. If the jukebox is used frequently then it would be assumed that there are many people in the room and the volume would be high while if the jukebox sits untouched for a period of time then it would be assumed that there are not many people in the room and that the volume would be lowered. While this method would be the easiest to implement there are a number of problems with making these assumptions. First of all many times a single person will stand at the jukebox and make many selections, therefore causing increased activity at the jukebox yet this one person is no indication of how many people there are in the room. Often there is a small group of people and one person is making a number of selections for that small group of people (never mind the fact that if it is late at night these songs may never play). Then there is the issue of the jukebox sitting untouched for a period of time. By observation it is clear that the jukebox can often go untouched for extended periods of time, even when the tap room is crowded and therefore it would be impossible to assume that there are a reduced number of people in the tap room based on an idle time period.

6.5 Volume Scheduling

Another method considered was volume scheduling. This is the idea of predicting the number of people in the room and giving the jukebox a schedule to set the volume of the system by. The problem with this idea is that the number of people in the room is inherently unpredictable. Even if the club's schedule of events could be accurately entered into the jukebox there is no way of predicting how successful any particular event may be and how many people will show up.

Additionally in any given night the size of the crowds can change very quickly for unanticipated reasons. For example, if there is a live band playing upstairs in the club then the crowd in the tap room would be relatively small since most people are upstairs watching the band. As soon as the band takes a break there is often an influx of people down to the tap room. It would be impossible to predict such breaks by the band.

6.6 Workaround

Ultimately it was decided not to pursue any of these methods and a simple workaround was devised. A wireless mouse was connected to the computer and the scroll wheel on the mouse was programmed to control the overall system volume. Access to the mouse is limited to club officers and staff (the people running the party) and the volume is changed several times per night if a club officer notices a volume discrepancy. Ideally this mouse would be moved to the opposite side of the room where there is always an officer on duty behind the bar but the wireless mouse only works within several feet of the computer.

7. Intelligence Features

Originally the intention of the jukebox was to include intelligent song selection that would guide users into choosing good music and would automatically make selections. Once the trial period began it because clear that these feature would not be necessary and may actually be a hindrance to users. There were usually over 100 requests per night, and people seemed to like the music that was selected, aside from the repetition problems. After adding many of the requested artists and albums to the music library users found the music that they were looking for.

One of the original ideas was a concept of "current hits" which would suggest songs based on the current hits on the music charts. After listening to the requests from users about the music library it became clear that the music charts would not help this system because the majority of the music is old and no longer on the charts. The most popular music appears to come from the boxed sets of classic rock artists such as Led Zeppelin, Bruce Springsteen, Billy Joel and Elton John. Once these collections were added to the jukebox there was no need to suggest newer music to users because the older music is mostly what they were looking for.

Another of the original intelligence ideas is the concept of "house favorites" which would maintain a list of the most requested songs in the location and suggest that music, or even insert the music into the playlist automatically. The system would also maintain information about the time of the day and day of the week that certain songs were requested so it would know to suggest or play them at similar times. This idea eventually lost its relevance due to the odd pace in which the music tastes of the room change. Often the people in the room would have a reason for liking a particular song but that reason could go away at any moment. This kind of behavior seems impossible to track.

When preparing to do this project I spoke with professional DJs about how to approach the problem of intelligence. One DJ, an employee of Soundtracks, a popular DJ company at Princeton believed the most important issue was knowing “who” was in the room at the time to listen to the music. He believed that having a vast music library that contained favorites for every generation and listening demographic was essential to the system. Fortunately for my sake, the demographic of the music listeners is extremely limited. This made for an easier time when choosing music. The majority of the music in the jukebox library would be considered “college music” and may not be appropriate for a jukebox system elsewhere.

This DJ also mentioned what the most popular requested songs are of each demographic. He mentions the Bon Jovi song Livin’ On A Prayer (which had been the most requested song in the jukebox thus far) because according to him, everyone sings along to the song. As our conversation was ending he paused and made me listen to the music currently echoing up from the tap room, and the voices singing along with it. The song was Livin’ On A Prayer and there were a number of student voices singing along to the song. Needless to say this same song was the motivation in creating a number of features in the jukebox to limit the repetition of the music. Too many people had complained about hearing this one song too many times so it had to be addressed.

8. Repetition Problems

There were a number of problems with repetitive music with the jukebox. This problem usually was a result of a few users making a large number of selections of songs. It is impossible for the jukebox to discern between different users because the users do not log in or authenticate in any way to use the jukebox and doing so would be too much of a hassle for users. Therefore measures need to be taken to avoid repeating the same songs, artists and albums.

8.1 Same Song Repetition

The biggest problem with giving users control over the song selection has been repetition. The same music can often be heard multiple times per night. The most popular songs are Livin' on a Prayer by Bon Jovi, Hey Ya! by Outkast, and Yellow by Coldplay. These songs can often be heard several times per night. Quite often a user will request one of these songs, completely unaware that the song has been requested earlier in the same night.

There are a number of possible ways to resolve this issue. One way would be to limit the number of requests for each song per night. This would mean that a song could be requested once per night. If it requested again then the request would not be accepted. I eventually decided upon a 3-hour minimum between requests of the same song. This seems to have resolved this repetition problem.

Even though it was requested, I decided against removing the most popular songs from the jukebox or disabling the playback of those songs. Even though some people didn't want to hear those songs at all due to their high play count if they are the most popular songs then they should continue to be heard.

8.2 Artist and Album Repetition

Quite often a user would make multiple selections from an album or an artist consecutively. This would often lead to a long period of time of very similar music. This situation was feared originally and it certainly has become a reality. In order to resolve this problem I have decided on a five- artist minimum between songs. This means that once a song is selected then five more songs have to be selected by five different artists before another song by the artist of the first song could be played. This will prevent someone from playing the entire 'N Sync album in order, which has been done by a user and very much annoyed people in the room.

9. Maintenance Issues

In order to clean up temp files the jukebox resets itself every 24 hours. It was difficult to choose what time to pick for this reset due to the unpredictability of the college students who frequent the location. There are often nights where students are out partying until dawn and it would be quite frustrating if the music were to turn off suddenly. Originally a time of 9 AM was chosen for reset. This reset would also stop the current music so that the club staff did not need to hear the music playing all day long since the club officers would often forget to turn the jukebox off at the end of the night. It was soon apparent that 9 AM was too late because much of the staff shows up at 7 AM and one night a silent fire alarm went off at 5 AM and the fire department showed up to a loud yet empty basement. Ideally that situation should be avoided in the future so the reset time was changed to 5 AM.

The resetting is controlled by a cron job. The cron job runs a script that deletes all temp files and creates new ones. It sets up the piping to allow communication between the different system components and it clears the jukebox playlist so that it can be repopulated with new music. This kind of script was necessary because previously this sort of setup had to be done manually. Now the jukebox can run indefinitely without administration thanks to the automated maintenance scripts.

10. Platform Decisions

The decision was made to use a Mac running Mac OS X due to the availability of the iTunes software and the strong apache and PHP support built into the platform. During the planning stages of this project in September 2003 the iTunes for Windows software was just a rumor. It would have been possible to base the system around the Winamp software on Windows. The problem with a Windows-based solution is that PHP is difficult to install for use with the Windows version of Apache and IIS web server software is often expensive and would most likely be cost prohibitive. Now that iTunes for Windows is a reality it should be noted that the Windows version of iTunes is not scriptable, since AppleScript does not exist on Windows. It would be possible to use iTunes as a library organization tool and use Winamp for playback but that would add another layer of complexity to the project that already consists of many layers. Additionally the font rendering of Mac OS X is vastly superior to that of Windows due to Quartz anti-aliasing built into Mac OS X and this ideal for a kiosk-like system.

The idea of commercializing this jukebox to be sold to bars and restaurants was discussed in an assignment for a class on High Tech Entrepreneurship (ELE 491). It was decided that if this project were to be commercialized it would most certainly need to be built on top of Linux to make use of the cheap hardware and free software. It would be impossible to sell jukeboxes based on Apple's hardware and software since Apple does not sell their hardware to licensees and does not license their software to other hardware vendors. It would not be practical to use new Macs to create jukeboxes due to their high price. While the PHP software could be easily ported to Linux, much of the functionality of iTunes would need to be recreated. This would significantly extend the development time of the jukebox and it would most likely be impossible to complete as a single semester project.

It is clear that there was a need for this project to be based on Mac OS X and on iTunes because the iTunes software allowed for concentration on the task at hand, which was to create a jukebox system and not to organize and play music.

11. Additional Features

Additional features have been considered for the jukebox but not implemented.

11.1 Network Music

One feature that was considered but not added is the ability to play music from other sources, not just the local music library. Students often have music shared on the campus network and they would like the ability to play that music from the jukebox. The reason this feature was never added is because it would allow users to play music that was not pre-selected for use. The music library was carefully created to provide music for the intended environment. Giving users the ability to bring in outside music without approval would destroy that work.

Additionally the feature would require creating a browser interface that can navigate network shares and possibly allow users to log in by entering a name and password. This would require extending the jukebox functionality well beyond the scope of this project.

11.2 Administrative Override

Many club members and officers requested a manual override feature that would allow them to skip songs that they didn't want to hear or allow their selections jump to the front of the queue. I considered these ideas however in the end decided that these features again would be beyond the scope of this project.

Adding these features would go against the concept that the jukebox could be run without administration. The override features could easily become abused by whoever has the control to access them and limiting that access would be a difficult process.

12. Conclusions

This project resulted in a functional jukebox that after much work and revision is a robust system and can function independently, without administration, forever. The system is not "intelligent" in the same sense that was previously planned, however it has more useful functionality instead.

The system was met with extremely positive user feedback. I would often hear comments from users about the "coolness" of the system and how much fun it is to use. I would observe groups of girls make a selection in the jukebox and run away giggling or single guys try to look as calm as possible while making their selections.

This type of system has not been created before to my knowledge. This is a complete jukebox system that can be used in other environments as well. There is a real world need for jukebox systems of this kind. If this project were to be commercialized this jukebox would be a very competitive and attractive alternative to traditional jukebox systems.

For these reasons I believe that the jukebox project is a success. The jukebox successfully serves its function on a nightly basis and the user base is satisfied with the system.

13. References

[1] iTunes – Smart Playlists. Apple Computer.
http://www.apple.com/itunes/smartplaylists.html

[2] Keytec, Inc. http://www.magictouch.com/

[3] Riptide Innovations. http://www.mp3touchscreens.com/

[4] Touch-Base. http://www.touch-base.com/

14. Appendix

14.1 Proposed Project Budget

Power Mac G4	
Used Power Mac G4 400 MHz (from ebay)	$504.00
Shipping:	$14.95
PayPal insurance:	$28.79
Additional 256 MB of RAM from 18004memory.com:	$36.95
Power Mac total:	$584.69

Touch Screen Display	
Touch Screen Driver Software:	$90
USB Touch Screen adapter	--
CRT Monitor	--
Display total:	$90
Project Total	**$ 674.69**