

基于 MFC 的简单音乐播放器

伍晨啸

四川大学锦城学院计算机与软件学院 四川 成都 610000

【摘要】网络上面有很多不同种类的播放器,而自己做出来的播放器扩展功能不完全,但是界面简洁,小巧玲珑,而基于 MFC 和微软 MCI 的音乐播放器制作简单,功能强大,不仅能让自己获得成就感,而且还能够实现很多功能。开发采用 VC++6.0,实现了音乐打开,播放,暂停,停止播放,音量调节等功能。

该系统实现了音乐播放的基本功能,能自己听音乐自用,也能成为自己做过的一个小项目。

【关键词】音乐播放器; MFC; Visual C++

前言

网络播放器日新月异,百花齐放,许许多多不同的人用过许多不同的音乐播放器,精致的人用着各种带有精致皮肤的播放器,专业的人花钱买下各种专业功能的播放器,而笔者做出的自己的音乐播放器,可以自用,也可以收藏,虽然功能不如大部分网络上流行的音乐播放器强大,但是自己做出来的有自己的血汗和成就感,属于自己的小幸运。

1 系统内容

本系统针对个人欣赏习惯,可以自己设置界面,设置背景,不用花钱去各大音乐播放器去买各种自己还不太喜欢的皮肤和背景,主要目的是自己的突发制作兴趣,和自己能完完全全的 DIY 界面和各个按钮在哪里,而且基本功能也得以实现,适合自己偶尔突然开心,想维持这个生活中的小开心,打开属于自己的音乐播放器,看着自己最顺手的界面,和自己 DIY 的背景图片,听着自己最喜欢的歌,做着自己最喜欢的事情。

2 系统界面设计

2.1 总体设计

根据一个音乐播放器应有的基本功能,设计出四个功能性的按钮,分别是打开(打开音乐文件)、播放(播放打开后的音乐文件)、暂停/继续(暂停和继续播放音乐)、停止(取消音乐接下来的播放,选择自己其他的歌曲)、音量调节条(调节音量的大小)。

2.2 打开按钮的系统设计

实现打开操作,锁定选择文件的类型只能是 MP3、wma、wav、和所有文件,执行打开操作之后会将音乐文件打开并保存路径,方便其他功能的操作。

2.3 播放按钮的系统设计

在打开音乐文件之后,按下播放按钮即可实现播放音乐的功能,默认是从音乐的最开始的时候进行播放。

2.4 暂停/继续播放按钮的系统设计

只有一个按钮,暂停播放和继续播放共享按钮,默认为暂停播放,当音乐正在播放的时候按下暂停按钮,即可暂停音乐的播放,此时原本按钮上面的字样从暂停播放变成了继续播放,同时功能也改变成了将暂停的歌曲继续播放,再次按下继续播放即可将暂停后的音乐继续播放,而按钮也改成了暂停播放,与之相应的就是按钮的功能改成了暂停播放。

2.5 停止播放按钮的系统设计

停止播放音乐,并且关闭当前打开的音乐。

2.6 音量调节的系统设计

有一个音量调节可拖动的进度条,并且在音量调节条旁边有一个示例编辑框显示音量的大小(0-100),有个文本栏提示示例编辑框显示的是音量的大小,而不是其他数字。

3 功能实现

3.1 本例开发环境

本例使用 VC++6.0 进行开发,VC++6.0 是一个代码编写和编译软件。全称为 Visual C++6.0,是在 Windows 环境中工作的^[1]。在程序界面方面,笔者选择的是使用 MFC 来实现整个音乐播放器的界面,因为他小巧玲珑,适合做这种个人向的小程序。MFC 是一个 Windows 应用程序开发模式,对程序的控制主要是由 MFC 框架完成的^[2]。使用了微软和 IBM 合作的开发的 MCI.媒体控制接口(Media

Control Interface, 缩写 MCI) 是由微软和 IBM 开发的一个高级应用程序接口 (API), 功能是控制连接到一台 Microsoft Windows 或 OS/2 计算机的多媒体外部设备 (例如 CD-ROM 播放机、音频控制器)。

由于这个音乐播放器属于个人向, 所以并没有设计联网的功能, 而是直接选择打开本地文件, 个人向的听音乐的范围本来就是自己个人下载收藏的音乐, 所以直接打开本地文件就足够了, 本音乐播放器程序的文件交互使用 CFile 来实现。MCI_PLAY、MCI_FROM、MCI_NOTIFY 实现对播放按钮的播放、从哪里播放、和播放的时候进行其他操作。

3.2 打开按钮系统实现

双击程序开始运行了之后, 就可以把整个音乐播放器的界面展示在使用者的面前, “笔者的音乐播放器” 整个界面有四个按钮第一个按钮就是打开按钮, 当单击打开按钮之后, 即可选择需要打开的音乐文件, 音乐文件只能是 mp3、wma、wav 和所有文件的格式。选择自己想要选择的本地音乐文件即可打开音乐文件。所有的中文显示均使用的 unicode 编码, 在前面加上大写的 L, L 告诉编译器字符串是一个宽字符串, 使用的是 unicode 编码可以保证中文显示不乱码。使用 MFC 的 CFileDialog dlg 来打开文件选择的对话框, 用 dlg.DoModal 来弹出对话框, 并且用 CFileDialog dlg 来规定选择文件的类型是 mp3、wma、wav 和所有文件这四种类型。在后面接一个 if 语句来确实使用者是否选择了文件, 如果没有选择文件则返回不确定路径, 如果选择的某一个文件则继续往下执行来获取文件的路径。使用 dlg.GetPathName 来获取选择的文件的路径并且保存到自己保存的一个歌曲变量里面来。并且可以使用 MessageBox 来显示文件的路径检查是否正确。获取文件的路径之后就准备用微软提供的 MCI 来进行执行选择音乐文件的实现各项功能。使用 MCI_OPEN 打开直接保存在之前定义的歌曲路径变量, 并街上 MCI_WAIT 等待下一步操作。然后获取打开后的音乐文件的 ID, 把他作为一个成员变量保存起来, 并且在构造函数里面给他一个初始值 0, 打开音乐之后再给他赋值。

3.3 获取错误信息的系统实现

使用一个 if 语句, 如果错误将使用 mciGetErrorString 将错误信息保存在自己设置的一个字符串变量里面, 并且用 MessageBox 将错误信息提示弹出, 随后返回值。让用户知道自己到底哪个步骤出了问题, 而不是单纯地不执行报错。

3.4 播放按钮的系统实现

在选择完了本地的音乐文件之后, 点击播放按钮即

可开始播放音乐。一般的网络音乐播放器都是默认从头开始播放音乐, 所以笔者选择使用 MCI_FROM 在确认从规定位置开始播放音乐, 并且设定变量让他从头开始播放音乐, 然后用 MCI_NOTIFY 来让笔者们在播放音乐的时候能回应笔者们其他的一些操作比如调音量等等, 最后使用 MCI_PLAY 来实现音乐的播放。

3.5 暂停 / 继续按钮的系统实现

在播放音乐的过程中, 单击暂停 / 继续按钮可以实现音乐播放的暂停和继续, 初始默认为暂停, 在播放时, 单击按钮即可暂停音乐, 并且按钮变为“继续”, 同时功能改为继续播放暂停后的音乐, 在单击按钮继续播放后, 按钮变为“暂停”。首先使用 GetDlgItemText 获取按钮的文本 (默认初始的时候是“暂停”), 再进行判断如果获取的按钮的文本是“暂停”则使用 MCI_PAUSE 暂停音乐的播放, 同时使用 SetDlgItemText 将按钮的文本改成“继续”, 再加一个判断如果获取的按钮文本是“继续”则使用 MCI_RESUME 继续暂停音乐的播放, 并且一样的, 同时使用 SetDlgItemText 将按钮的文本改成“暂停”。

3.6 停止按钮的系统实现

使用 MCI_STOP 停止音乐的播放, 再接着用 MCI_CLOSE 关闭选择的音乐文件, 得以选择下一个本地音乐文件。

3.7 音量调节的系统实现

滑动可滑动的音量调节大小的进度条可以调节音量, 左边则用一个示例编辑框来数字化音量百分比大小, 在示例编辑框周围用文本框提示这是音量的大小, 音量大小 0-1000, 显示大小 0-100 (更人性化)。给滑动条和示例框设置一个变量, 将滑动条的变量范围设置成 0-1000, 可以让音量调节更加精细再用 SetPos 将初始音量设置成 500。在类向导里面给滑动条添加一个 NM_CUSTOMDRAW 的消息, 再用 GetPos 获取滑动控件的位置将其除以 10 之后赋值给音量变量, 并用 UpdateData (False) 来获取的状态。使用 MCI_DGV_SETAUDIO_VALUE 和 MCI_DGV_SETAUDIO_ITME 来将音量的值赋值为滑动条的值, 而不是音量的值, 因为音量的值仅仅只是显示百分之, 是 0-100, 而实际的音量大小是滑动条的值 0-1000。

4 结束语

虽然说本次的音乐播放器实现了音乐播放的一个最基本的功能如播放、暂停、音量调节等等, 但是其实还是有很多额外功能更加方便的功能是没有实现的, 有很大的完善空间, 比如设置歌单和切换歌曲, 而且本文不

讨论界面的设置和各个控件的位置,背景和图片的设计,因为这个音乐播放器本身就属于 DIY 界面和根据自己的需求去改变某一个功能的个人向播放器。这个音乐播放器很简洁,但是自己做出来的东西在精神上享受的东西会比其他一些实用性质的东西更加有成就感和舒心。还有什么比在一个懒洋洋的下午坐在太阳下的躺椅上,用

着自己做的音乐播放器听着自己喜欢的音乐小憩一会更舒服的事情呢?

【参考文献】

- [1] 谭浩强 .C 程序设计 [M]. 北京 : 清华大学出版社 ,2010.
- [2] 初阳 , 杨永祥 .MFC 应用简介 [J]. 通讯世界 ,2016(08).