

HỌC VIỆN CÔNG NGHỆ BƯU CHÍNH VIỄN THÔNG
CƠ SỞ TẠI THÀNH PHỐ HỒ CHÍ MINH
KHOA CÔNG NGHỆ THÔNG TIN



ĐỒ ÁN HỌC PHẦN

TÊN HỌC PHẦN: LẬP TRÌNH ÂM THANH
(Audio-Application Programming: AudioPro)

MÃ SỐ LỚP HP: MUL14126-01

Tên đề tài: **LẬP TRÌNH HỆ THỐNG QUẢN LÝ FILES ÂM THANH CÒI XE**

Họ và tên : Trần Văn Ngạn
Mã số sinh viên : N20DCPT044
Lớp : D20CQPTUD01-N
Ngày nộp :/...../2023
Ký tên :

TP.HCM, ngày 07 tháng 03 năm 2023

HỌC VIỆN CÔNG NGHỆ BƯU CHÍNH VIỄN THÔNG
CƠ SỞ TẠI THÀNH PHỐ HỒ CHÍ MINH
KHOA CÔNG NGHỆ THÔNG TIN

ĐỀ TÀI ĐỒ ÁN HỌC PHẦN

Giảng viên giảng dạy: VÕ XUÂN THỂ

TÊN HỌC PHẦN: LẬP TRÌNH ÂM THANH
(Audio-Application Programming: AudioPro)

MÃ SỐ LỚP HP: MUL14126-01

Tên đề tài: LẬP TRÌNH HỆ THỐNG QUẢN LÝ FILES ÂM THANH CÒI XE

Họ tên sinh viên: Trần Văn Ngạn

Mã số sinh viên: N20DCPT044

Lớp: D20CQPTUD01-N

Tên sản phẩm đề tài: Hệ thống quản lý file âm thanh còi xe

Công cụ sử dụng: MS. Visual Studio .NET 4.8.1

với bộ đóng gói Setup: VSI_Extentions.exe

Thư viện (phiên bản): WMPLib trên .NET Framework 4.8.1

Ngôn ngữ lập trình sử dụng: C#.NET

DataBase lưu thông tin các file âm thanh: AP06TVNGAN

trên MS. SQL Server 2022 (ServerName: NGANCONDETHUONG Username: sa Password: 123)

Tập dữ liệu thực nghiệm: <https://www.pixabay.com>

Nhận xét của giảng viên:

.....

Điểm đánh giá:.....(.....)

Ngày/...../2023

Giảng viên: Ký tên

TP.HCM, ngày 07 tháng 03 năm 2023

LỜI CẢM ƠN

Em xin chân thành cảm ơn thầy Võ Xuân Thê đã tận tình hướng dẫn em trong quá trình thực hiện đồ án này. Nhờ sự hướng dẫn của thầy mà em có thể hoàn thành đồ án này một cách tốt nhất.

Một lần nữa, em xin cảm ơn thầy rất nhiều. Em hy vọng rằng mình có thể tiếp tục được học hỏi thêm từ thầy trong các học phần tiếp theo.

DANH MỤC CÁC TỪ VIẾT TẮT

Từ viết tắt	Giải thích
CSDL hoặc DB	Cơ sở dữ liệu: DataBase
NSD = User = Account	Quyền làm việc được cấp cho NGƯỜI SỬ DỤNG trên hệ thống App, quyền này thông thường được đảm bảo bởi tên đăng nhập (username) và mật khẩu (password); cũng có thể gọi là “tài khoản người dùng” [“tài khoản”]
C#.NET	Ngôn ngữ lập trình dựa trên ngôn ngữ C/C++ được dùng trên bộ công cụ MS. Visual Studio.NET
WMP	Windows Media Player: thư viện đa phương tiện của hệ điều hành Window
APP	Ứng dụng – Chương trình – Application

DANH MỤC CÁC THUẬT NGỮ

Giao diện người dùng (User-Interface)	Là hệ thống các màn hình giao tiếp cho phép người sử dụng tương tác với các thành phần phần mềm, điều khiển phần mềm hoạt động theo yêu cầu của người dùng - tương ứng các chức năng hiện có của phần mềm.
Người dùng (User) NSD: Tài khoản (Account)	Là một quyền làm việc trên hệ thống phần mềm được cấp phát cho một cá nhân thông qua tên tài khoản (username) và mật khẩu (password).

MỤC LỤC

LỜI CẢM ƠN.....	3
DANH MỤC CÁC TỪ VIẾT TẮT.....	4
DANH MỤC CÁC THUẬT NGỮ	5
MỤC LỤC 6	
PHIẾU GIAO NHIỆM VỤ ĐỀ TÀI ĐỒ ÁN HỌC PHẦN	8
Chương 1: GIỚI THIỆU VỀ ĐỀ TÀI.....	9
1.1. Tổng quan về đề tài	9
1.2. Nội dung chuyên môn chính của đề tài	9
1.3. Bố cục của báo cáo.....	9
Chương 2: CƠ SỞ LÝ THUYẾT VỀ LẬP TRÌNH ÂM THANH	10
2.1. Tổng quan về âm thanh học và kỹ thuật âm thanh.....	10
2.1.1. Các khái niệm	10
2.1.2. Thực tiễn của âm thanh	13
2.1.3. Khuynh hướng kỹ thuật âm thanh	13
2.1.4. Các đặc tính cơ bản của âm thanh	13
2.1.5. Các đặc tính cơ bản của file lưu trữ âm thanh (Digital)	14
2.1.6. Các dạng đuôi (phần mở rộng Extensions) file âm thanh.....	14
2.1.7. Các dạng nén file âm thanh thông dụng dựa vào BitRate	15
2.1.8. Một số dạng xử lý chất lượng âm thanh	15
2.1.9. Một số kỹ thuật thu thập âm thanh (Thu âm)	16
2.1.10. Các xử lý cơ bản trên tập tin âm thanh kỹ thuật số.....	16
2.1.11. Các xử lý cơ bản với đặc tính âm thanh kỹ thuật số	16
2.2. Công cụ và các hệ thống thư viện hỗ trợ Audio Pro... ..	17
2.2.1. Phần mềm công cụ sử dụng chính trong học phần.....	17
2.2.2. Các thư viện lập trình chính được sử dụng trong học phần.....	17
2.2.3. Tập các âm thanh mẫu được sử dụng trong đề tài của Học phần.....	17
Chương 3: GIỚI THIỆU CÁC THƯ VIỆN AudioPro SỬ DỤNG	18
3.1. Nạp các thư viện xử lý âm thanh (Audio Libraries)	18
3.2. Nạp BỘ CÔNG CỤ của các thư viện.....	19
Chương 4: GIỚI THIỆU VÀ CÀI ĐẶT SẢN PHẨM	20
4.1. Mô tả sản phẩm	20
4.2. Cài CSDL lưu trữ thông tin âm thành	20
4.2.1. Lập DB SQL chỉ 1 Table:.....	20
4.2.2. Nhập dữ liệu thực nghiệm:	21
4.3. Lập Audio App Winform mới: AP06TVNGAN.....	22

4.4.	Thiết kế và Lập trình Winform màn hình chính: FormMain06.cs	22
4.4.1.	Thiết kế (Design)	22
4.5.	Thiết kế và Lập trình Winform màn hình quản lý âm thanh dùng WMPLib	26
4.5.1.	Thiết kế (Design)	26
4.5.2.	Một số số mã lệnh ban đầu (Codes)	29
4.6.	Sử dụng công cụ trong Toolbox (đối với một số thư viện có hỗ trợ công cụ)	30
4.7.	Lập trình (Codes) XỬ LÝ âm thanh sử dụng thư viện	31
4.8.	Lập trình (Codes) QUẢN LÝ âm thanh sử dụng thư viện	32
4.8.1.	Giới thiệu	32
4.8.2.	Nạp thêm file âm thanh (Load)	33
4.8.3.	Sửa thông tin file âm thanh (Modify)	36
4.8.4.	Ghi âm file âm thanh (Record)	37
4.8.5.	Xóa file âm thanh (Delete)	37
4.9.	Lập trình (Codes) XỬ LÝ âm thanh sử dụng thư viện irrKlang	39
4.9.1.	Thiết kế Forms	39
4.9.2.	Lập trình (Codes)	40
Chương 5: ĐÓNG GÓI BỘ CÀI ĐẶT SETUP CỦA SẢN PHẨM ĐỀ TÀI		41
5.1.	Giới thiệu chung	41
5.2.	Công cụ sử dụng thiết lập bộ cài đặt Setup cho App	41
5.3.	Thiết lập bộ cài đặt Setup cho App	41
Chương 6: KẾT LUẬN VÀ HƯỚNG PHÁT TRIỂN CỦA ĐỀ TÀI		45
6.1.	Kết luận	45
6.1.1.	Những kết quả đạt được	45
6.1.2.	Hạn chế	46
6.2.	Hướng phát triển	46
6.2.1.	Hướng khắc phục các hạn chế	46
6.2.2.	Hướng mở rộng đề tài	46
DANH MỤC TÀI LIỆU THAM KHẢO		47

HỌC VIỆN CÔNG NGHỆ BƯU CHÍNH VIỄN THÔNG
CƠ SỞ TẠI THÀNH PHỐ HỒ CHÍ MINH
KHOA CÔNG NGHỆ THÔNG TIN
PHIẾU GIAO NHIỆM VỤ ĐỀ TÀI ĐỒ ÁN HỌC PHẦN

Giảng viên giảng dạy: VÕ XUÂN THỂ	
Tên học phần: Lập trình âm thanh (AudioPro) Mã số lớp HP: MUL14126-01	
Tên đề tài: LẬP TRÌNH HỆ THỐNG QUẢN LÝ FILES ÂM THANH CÒI XE	
Sinh viên thực hiện: Trần Văn Ngạn – N20DCPT044	
Thời gian thực hiện: 03/01/2023 đến 07/03/2023	
<p>Yêu cầu của đề tài</p> <p>Lý thuyết: Vận dụng kiến thức lập trình âm thanh vào việc phát triển một hệ thống ứng dụng quản lý âm thanh dùng trong một lĩnh vực chuyên môn thực tiễn, minh họa với ngôn ngữ lập trình C#.NET dùng công cụ Microsoft Visual Studio .NET với một số thư viện hỗ trợ lập trình âm thanh, như: WMPLib (Windows Media Player Library), Irrklang, OpenAL (Open Audio Library). Tham khảo tập âm thanh mẫu https://.pixabay.com</p> <p>Thực hành: LẬP TRÌNH HỆ THỐNG QUẢN LÝ FILES ÂM THANH CÒI XE với C#.NET gồm:</p> <ul style="list-style-type: none"> + Thư viện chính được dùng: WMPLib và irrKLang + Giải thuật chính: Lập trình sự kiện bằng Events trên WinForms của .NETFramework 4.8.1. + File âm thanh tập thực nghiệm chủ yếu được sử dụng: .wav và .mp3 + Các chức năng chính của sản phẩm: <p style="padding-left: 20px;">FormMain06.cs: Màn hình chính</p> <p style="padding-left: 20px;">Fr2WMP06.cs: Quản lý và xử lý âm thanh dùng thư viện WMPLib.dll</p> <p style="padding-left: 20px;">Fr3irrKLang.cs: Xử lý âm thanh cơ bản dùng thư viện irrKLang.dll</p>	
GIẢNG VIÊN	Ngày 03 tháng 01 năm 2023 Sinh viên thực hiện

Chương 1: GIỚI THIỆU VỀ ĐỀ TÀI

1.1. Tổng quan về đề tài

Tìm hiểu về âm thanh học và kỹ thuật âm thanh, đồng thời dựa trên công cụ lập trình MS. Visual studio .NET với ngôn ngữ lập C#.NET sử dụng các thư viện hỗ trợ lập trình âm thanh (Audio Program), như: WMPLib.dll và irrKLang.dll để phát triển ứng dụng (App) dạng Winforms (tức là WinApp) cho phép quản lý và xử lý các tập tin âm thanh sử dụng trong HỆ THỐNG QUẢN LÝ FILE ÂM THANH CÒI XE

Tập dữ liệu thực nghiệm chính được tham khảo (thu thập) từ <https://pixabay.com>. Chủ yếu là các dạng files âm thanh: .wav, .mp3

1.2. Nội dung chuyên môn chính của đề tài

- + Thư viện: WMPLib.dll và irrKLang.dll
- + Công cụ: sử dụng công cụ MS. Visual Studio .NET dạng Winform lập trình sự kiện Event với ngôn ngữ lập trình C#.NET
- + Sản phẩm đề tài: Tên sản phẩm: **AP06TVNGAN**

ỨNG DỤNG QUẢN LÝ & XỬ LÝ ÂM THANH: LẬP TRÌNH BỞI TRẦN VĂN NGẠN
HỌC VIÊN CÔNG NGHỆ BUỒI CHÍNH VIỄN THÔNG TP.HCM

1.3. Bố cục của báo cáo

Chương 1: Giới thiệu tổng quan về đề tài lập trình WinApp quản lý và xử lý âm thanh Hệ thống quản lý files âm thanh còi xe

Chương 2: Cơ sở lý thuyết của học phần: gồm những lý thuyết cơ bản về

- + Âm thanh học và kỹ thuật âm thanh AudioPro liên quan đến đề tài.
- + Kỹ thuật lập trình Winforms trên MS. Visual Studio .NET với ngôn ngữ C#.NET dùng thư viện WMPLib.dll

Chương 3: Giới thiệu về thư viện và thủ tục nạp các thư viện vào App

Chương 4: Thiết kế và lập trình sản phẩm của đề tài: WinApp: AudioPro quản lý và xử lý âm thanh Hệ thống quản lý file âm thanh còi xe

Chương 5: Thủ tục thiết lập bộ cài đặt Setup.exe của sản phẩm hệ thống AudioPro: Hệ thống quản lý files âm thanh còi xe

Chương 6: Kết luận về kết quả đạt được và những tồn tại, trên cơ sở đó đề xuất các giải pháp khắc phục tồn và hướng mở rộng đề tài.

Chương 2: CƠ SỞ LÝ THUYẾT VỀ LẬP TRÌNH ÂM THANH

2.1. Tổng quan về âm thanh học và kỹ thuật âm thanh

2.1.1. Các khái niệm

2.1.1.1. Âm thanh

Khái niệm: Âm thanh là các dao động cơ học (biến đổi vị trí qua lại) của các phân tử, nguyên tử hay các hạt làm nên vật chất và lan truyền trong vật chất như các sóng. Âm thanh, giống như sóng, được đặc trưng bởi tần số, bước sóng, chu kỳ, biên độ và vận tốc lan truyền (tốc độ âm thanh).



2.1.1.2. Phân loại âm thanh

Về cơ bản có 7 loại âm thanh phổ biến:

Âm thanh Stereo

Âm thanh STEREO là âm thanh từ nhiều nguồn âm được phân bổ "Từ phải sang trái hoặc ngược lại ". Ví dụ : một người nói bên phải của bạn, một chú chó sủa bên trái của bạn, một chú chim hót giữa người và chó ...

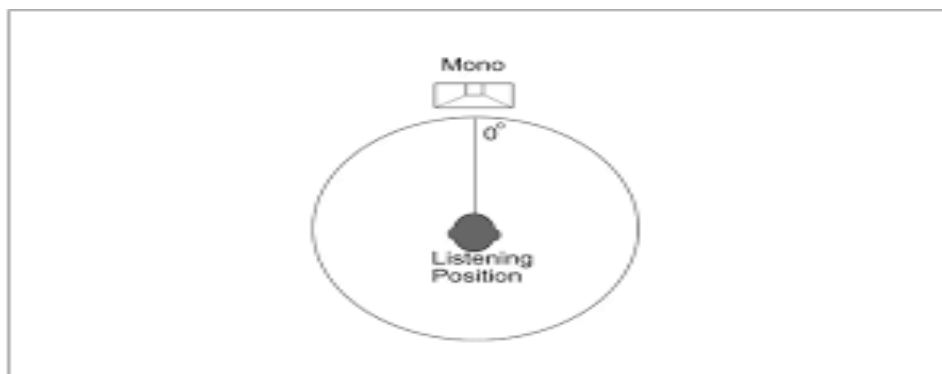
Hoặc một nguồn âm nhưng chạy từ trái sang phải hoặc ngược lại. Ví dụ : một chiếc xe hơi chạy từ trái sang phải, chú chim vừa bay vừa hót từ phải sang trái ...



Âm thanh Mono

Âm thanh monophonic chỉ có một kênh, xuất phát từ một phương hướng. Nó chỉ dùng một amplifier và một loa để phát lại. Đối với thính giác của bạn, tất cả những chi tiết của

âm nhạc, như là giọng hát, những nhạc cụ v.v hình như được phát ra từ một điểm trong không gian.



Nếu bạn gắn hai loa vào vào một nguồn monophonic, âm thanh bây giờ giống như là nó được phát ra từ một điểm ở chính giữa hai loa, tạo ra một kênh ảo (phantom channel). Dễ hiểu nhất : Âm thanh mono là âm thanh được phát ra từ một nguồn âm thanh từ một địa điểm cố định.

Âm thanh Surround

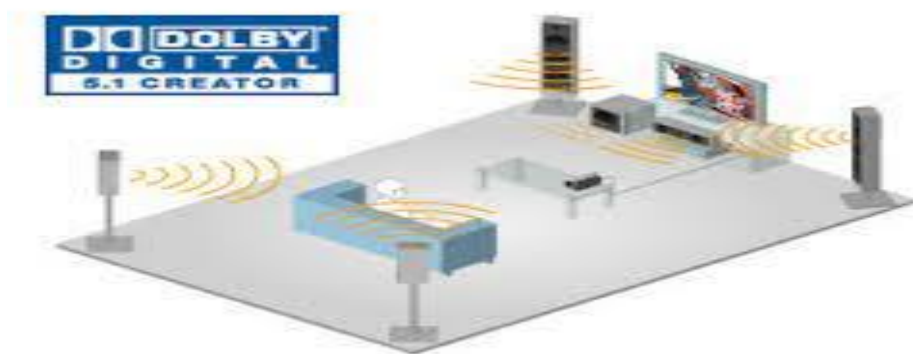
Có người gọi là âm thanh lập thể, nhưng theo mình, từ chính xác nhất phải là : âm thanh bao quanh bạn. Hệ thống âm thanh này nhằm mục đích mô tả lại những âm thanh diễn ra xung quanh bạn, tạo cảm giác thực nhất khi xem phim.

Ví dụ : tiếng chim hót ở bên trái, Tiếng đạn bắn bên phải, tiếng bom nổ đằng sau thì những âm thanh này xuất hiện trên phim như nào thì bạn sẽ cảm giác đúng ý như vậy với vị trí ngồi của bạn.



Âm thanh bốn kênh rời và Quadraphonic

Vào cuối thập niên 60 và vào đầu của thập niên 70, có hai sự phát triển trong âm thanh mà nó cố gắng giải đáp những giới hạn của âm thanh Stereo. Đó là bốn-kênh-rời và Quadraphonic. Khó khăn với bốn-kênh-rời.



Trong hệ thống bốn-kênh-rời, cần phải có bốn amplifier giống nhau, và nó rất là đắt tiền. Đây là những năm tháng của Bóng đèn và Transistors, không phải là IC như ngày hôm nay.

Dolby Pro Logic

- Dolby Pro Logic vượt qua những giới hạn của Dolby Surround thông thường bằng cách gắn thêm vào phần cứng và phần mềm trong IC giải mã bằng cách nhấn mạnh những tín hiệu quan trọng về phương hướng trong âm thanh của phim. Nói một cách khác, bộ giải mã sẽ nhấn mạnh vào sự định vị của âm thanh bằng cách tăng biên độ của các âm này trong các kênh tương ứng.
- Qui trình này, mặc dầu không quan trọng trong sự thâm âm của âm nhạc, nhưng rất là có hiệu quả trong phim. Nó làm cho những sound effects như là tiếng bom nổ, phi cơ bay ngang đầu... được thể hiện rõ ràng hơn. Âm thanh có sự phân biệt rõ ràng giữa các kênh. Thêm vào đó, Dolby Prologic trích ra một kênh chuyên dùng là kênh Center, dùng để đặt những lời đối thoại vào giữa.

Âm thanh vòm Dolby Surround

- Phương pháp âm thanh vòm Dolby bao gồm việc mã hoá bốn kênh của tín tức - Front Left (trái trước), Center (trung tâm), Front Right,(trái sau) và kênh Rear (Đằng sau) vào trong hai kênh tín hiệu. Một IC giải mã sau đó sẽ chuyển tín hiệu này đến nơi thích hợp như là kênh trái, kênh phải, kênh sau và kênh giữa. (kênh giữa là do tích hợp giữa hai kênh trái và phải).
- Hiệu quả của Dolby Surround là một môi trường nghe cân bằng. Trong đó những âm thanh chính được phát ra từ kênh trái và kênh phải. Lời ca hay là lời đối thoại được phát ra từ kênh giữa, và không khí của âm nhạc (ambience) được phát ra từ sau lưng người nghe.

Dolby Digital

Với định dạng Dolby Digital, bạn có 6 kênh riêng biệt, mỗi kênh cho một loa. Dolby Digital hỗ trợ tới 5 kênh độc lập toàn dải (loa trung tâm, loa trái, phải trước, loa surround trái và phải) và kênh thứ 6 chuyên tần số thấp cho loa siêu trầm. Do có 5 kênh độc lập và một kênh siêu trầm, Dolby Digital được gọi là hệ thống 5.1.

Tất nhiên, không phải tất cả các âm thanh ghi bằng Dolby Digital đều là 5.1, một số có thể ít hơn (có thể 2.0 hay 5.0...), nhưng định dạng này hỗ trợ tối đa 5.1 kênh.

Trên đây là 7 loại âm thanh cơ bản nhất mà chúng ta thường gặp. Hy vọng với những kiến thức âm thanh cơ bản này sẽ giúp bạn am hiểu hơn, để việc lựa chọn, bố trí sắp đặt loa trong phòng sao cho hay nhất.

2.1.2. Thực tiễn của âm thanh

Âm thanh là một loại tín hiệu. Và nó rất cần thiết trong cuộc sống không những của con người mà còn của các loài vật khác.

Âm thanh là phương tiện trao đổi thông tin, giúp thông tin nhanh chóng đến được đối tượng cần truyền tin.

2.1.3. Khuynh hướng kỹ thuật âm thanh

2.1.4. Các đặc tính cơ bản của âm thanh

- Cao độ của âm thanh \approx Tần số âm thanh (sound frequency [Hz]):
Âm trầm (bass), Âm trung (mid), Âm cao (treble)
- Cường độ âm thanh (sound intensity) [decibel = dB = 1/10 B || Bel = B]
- Trường độ của âm thanh = Băng thông (Độ trễ || Bandwidth) [Hz]
- Âm sắc (Timbre) (theo đối tượng phát ra âm).

NGOÀI RA, CÒN CÓ:

- Thanh áp (áp suất âm thanh, ký hiệu p [miliBar = mB, Bar])
- Công suất âm thanh $P = p_{sv}$ [Watt] { s: thiết diện, v: vận tốc âm thanh, p: thanh áp }
- Độ xuất hiện đột ngột (transients)
- Độ rung (vibrato)
- Độ thay đổi (envelope modulation).

...

CHÚ Ý:

+ Tone (tông) của âm thanh ["Giọng" của âm thanh] = trường độ + cường độ + cao độ + âm sắc

VD: Tông Nam, Tông Nữ trong Karaoke

2.1.5. Các đặc tính cơ bản của file lưu trữ âm thanh (Digital)

- + Sample (giá trị biên độ tần số lấy mẫu)
- + Sample Rate (Số lần lấy mẫu/s) [T] => Sample Frequency (tần số lấy mẫu) [f]; VD: 44100 Hz
- + BitDepth (số lượng bit số trong mẫu được lấy, còn gọi là độ “nét”: resolution); VD: 16 bits, 24 bits
- + Channel (số kênh tín hiệu số được tách ra); VD 2 kênh

Đây là cơ sở xác định âm thanh Mono (âm thanh “phẳng” = đơn kênh = 1 channel) hay Stereo (âm thanh “nổi” = đa kênh); có nhiều dạng

2D = 2 channel = kênh “trái” và kênh “phải” = 2 chiều

3D (3 Dimensions = 3 chiều) = 3 kênh, . . .

Ghi âm lời nói nên Mono, nhưng Âm nhạc thì nên Stereo, 2D, 3D, Surround,...

- + Bit-Rate (số lượng bit số của âm thanh/s) [kbps (Kilobits per second)] ;

VD: Một phút nhạc 128 kbps có dung lượng khoảng 1 MB và
bản nhạc 320 kbps thì chắc chắn sẽ hay hơn bản nhạc 128 kbps.

2.1.6. Các dạng đuôi (phần mở rộng || Extensions) file âm thanh

Hiện nay có một vài dạng file âm thanh phổ biến sau:

+ MP3: MP3 - MPEG-1 audio Player 3 hay Motion Pictures Expert Group 1 Layer 3. Là định dạng âm thanh được tạo ra trong quá trình cắt bỏ bớt các chi tiết ở dãy âm quá thấp và quá cao khi nén âm thanh. Do đó, nó có dung lượng cực nhẹ, dễ cho việc chia sẻ và lưu trữ. Nhưng nó có nhược điểm là chất lượng sẽ giảm đi nhiều so với bản gốc studio hay CD. Đây là định dạng phổ biến nhất hiện nay.

+WMA: WMA - Windows Media Audio. Định dạng này do Microsoft tạo ra để cạnh tranh cùng MP3. Với ưu điểm là dung lượng còn nhẹ hơn cả MP3 nhưng chất lượng lại tương đương nên nó cũng được yêu thích bởi khá nhiều người chơi.

+WAV: WAV - Waveform Audio File Format. Được xem là sự thay thế cho các bản gốc studio hoặc CD và được tạo ra bởi Microsoft hợp tác cùng IBM. Nhưng cũng vì thế, dung lượng của nó khá nặng và chất âm không thua âm thanh phòng thu.

+FLAC: FLAC - Free Lossless Audio Codec. Được tạo ra bởi công nghệ nén cho phép giữ lại các dữ liệu và chi tiết cần nên chất lượng khá đảm bảo dù dung lượng chỉ bằng một nửa so với WAV. Nó là sự lựa chọn của nhiều người bởi yêu cầu dung lượng không quá nặng và chất âm phải tốt.

+AAC: AAC – Advanced Audio Coding. Có thể xem nó là anh em song sinh với MP3 khi cùng sử dụng kỹ thuật nén âm thanh mà không làm mất đi nhiều dữ liệu. Ưu điểm của nó so với MP3 là có thể trình diễn các dải tần thấp ở đa kênh mà MP3 hiếm khi làm được.

+OGG: OGG là một file âm thanh sử dụng phương pháp nén Ogg Vorbis nhưng chất lượng cực tốt và nhỏ hơn cả MP3 dù dung lượng tương đương nhau.

+AIFF: AIFF là các file âm thanh chất lượng cao, có dung lượng tương đương với file WAV và dùng để lưu trữ âm thanh CD. Đối với máy tính hệ điều hành Windows, các file AIFF thường có phần đuôi .AIF.

+ALAC: ALAC (M4A) - Apple lossless audio code. Là định dạng được tạo ra bằng phương pháp nén bảo toàn dữ liệu gốc độc quyền của Apple. Nó có nhiều điểm tương đồng với FLAC nhưng chưa được sử dụng nhiều.

2.1.7. Các dạng nén file âm thanh thông dụng dựa vào BitRate

+ CBR (Constant BitRate) = Dùng giá trị BitRate cố định (chọn), đơn giản, ít hiệu quả

+ ABR (Average BitRate) = Dùng giá trị bình quân của BitRate, phức tạp hơn, hiệu quả hơn

+ VBR (Variable BitRate) = Dùng các giá trị BitRate khác biệt cho từng đoạn nén, rất phức tạp, hiệu quả rất cao



2.1.8. Một số dạng xử lý chất lượng âm thanh

Có nhiều công đoạn trong quy trình xử lý chất lượng âm thanh. Một trong số đó là:

+ Xử lý tạp âm (Noise Processing) và lọc nhiễu (Noise Filter) dùng “bộ lọc”

+ Cộng hưởng âm = Cộng hưởng qua tần số của âm (cùng tần số f sẽ cộng hưởng âm) = Khuếch đại âm thanh

+ Trích lọc âm và tách âm

+ Hòa âm (Kết hợp các dạng âm thanh) – phối khí (phối hợp các nhạc cụ)

+ Âm thanh Hi-Fi (Chất lượng cao) và Hi-End

+ Hợp âm (chord) trong âm nhạc: Đồ, Rê, Mi, Fa, Son, La, Si = kết hợp nhiều âm thanh (âm thanh phát ra từ nhiều dây đàn).

2.1.9. Một số kỹ thuật thu thập âm thanh (Thu âm)

Chương trình phần mềm, có thể:

- + Ghi âm
- + Nạp files âm thanh
- + Tạo file âm thanh bằng phần mềm

2.1.10. Các xử lý cơ bản trên tập tin âm thanh kỹ thuật số

- + Xử lý tập âm
- + Chuẩn hóa biên độ âm thanh
- + Loại bỏ khoảng lặng không cần thiết
- + Khuếch đại âm thanh
- + Chuyển âm sắc
- + Chuyển tông giọng

2.1.11. Các xử lý cơ bản với đặc tính âm thanh kỹ thuật số

Ta sẽ nói đến 3 xử lý cơ bản đó là: chuyển đổi, lưu trữ và truyền tải âm thanh kỹ thuật số.

+ **Chuyển đổi:** Để chuyển đổi âm thanh kỹ thuật số cần một ADC chuyển đổi tín hiệu tương tự thành tín hiệu số. ADC chạy ở tốc độ lấy mẫu đã được chỉ định từ trước và chuyển đổi độ phân giải bit đã biết.

Một hệ thống âm thanh tương tự chuyển đổi các dạng sóng vật lý của âm thanh thành các biểu diễn điện của các dạng sóng đó bằng cách sử dụng đầu dò, chẳng hạn như *micro*. Các âm thanh sau đó được lưu trữ trên một phương tiện tương tự như băng từ hoặc được truyền qua một phương tiện tương tự như đường dây điện thoại hoặc radio. Để tái tạo, quá trình bắt buộc phải được đảo ngược: tín hiệu âm thanh được khuếch đại ra bên ngoài, sau đó, thông qua loa chuyển đổi trở lại thành dạng sóng vật lý.

+ **Lưu trữ và truyền tải:** Trong quá trình truyền tải hoặc lưu trữ tín hiệu, nếu gặp bất kỳ lỗi sai nào thì âm thanh kỹ thuật số có thể được mã hóa tùy ý nhằm sửa đổi sai sót. Kỹ thuật này được các chuyên gia trong ngành gọi là mã hóa kênh, có vai trò rất quan trọng trong các hệ thống kỹ thuật số được phát sóng hoặc ghi lại để duy trì độ chính xác bit.

2.2. Công cụ và các hệ thống thư viện hỗ trợ Audio Pro...

2.2.1. Phần mềm công cụ sử dụng chính trong học phần

1. Microsoft Visual Studio = .NET 2022
 - Hệ thống thư viện phần mềm: .NET Framework 4.8.1
 - Ngôn ngữ lập trình: C#.NET
 - VSI_Extensios: Bộ dịch vụ đóng gói bộ cài đặt Setup trên .NET
2. Hệ quản trị CSDL: MicroSoft SQL Server 2022
 - Chỉ sử dụng để lưu trữ các thông tin của đối tượng âm thanh cần quản lý

2.2.2. Các thư viện lập trình chính được sử dụng trong học phần

Các thư viện xử lý Audio / .NET

2.2.2.1. WMPLib

Windows Media Player Library: Thư viện hỗ trợ lập trình phát triển ứng dụng đa phương tiện trên hệ điều hành MS. Windows (Có sẵn trên windows):

Đường dẫn C:\Windows\System32\wmp.dll

2.2.2.2. Irrklang

Thư viện IrrKlang có đính kèm đồ án.

2.2.2.3. OpenAL

Open Audio Library: Thư viện hỗ trợ lập trình âm thanh 3D đa kênh

C:\Windows\SysWOW64\openal32.dll OR C:\Windows\System32\openal32.dll

Giải nén OpenAL.rar => oalinst.exe, OpenAL11CoreSDK.exe

2.2.3. Tập các âm thanh mẫu được sử dụng trong đề tài của Học phần

Tập tin mẫu trong đề tài được lấy từ <https://pixabay.com>

Những tập tin được sử dụng trong đồ án này chỉ được sử dụng cho mục đích học tập và **KHÔNG** dành cho mục đích thương mại

Chương 3: GIỚI THIỆU CÁC THƯ VIỆN AudioPro SỬ DỤNG

3.1. Nạp các thư viện xử lý âm thanh (Audio Libraries)

1. WMP [WMPLib]

- wmp.dll (Windows Media Player của MS.Windows)
- wmp.dll là thư viện phục vụ việc xử lý âm thanh đơn giản được cung cấp sẵn bởi các Hệ điều hành Microsoft Windows.
- File thư viện (mặc định) được lưu trữ trong thư mục:
C:\Windows\System32\wmp.dll

2. Irrklang

- Thư viện hỗ trợ xử lý âm thanh 2D và 3D trên nhiều hệ điều hành khác nhau, như Windows, Linux và MacOS

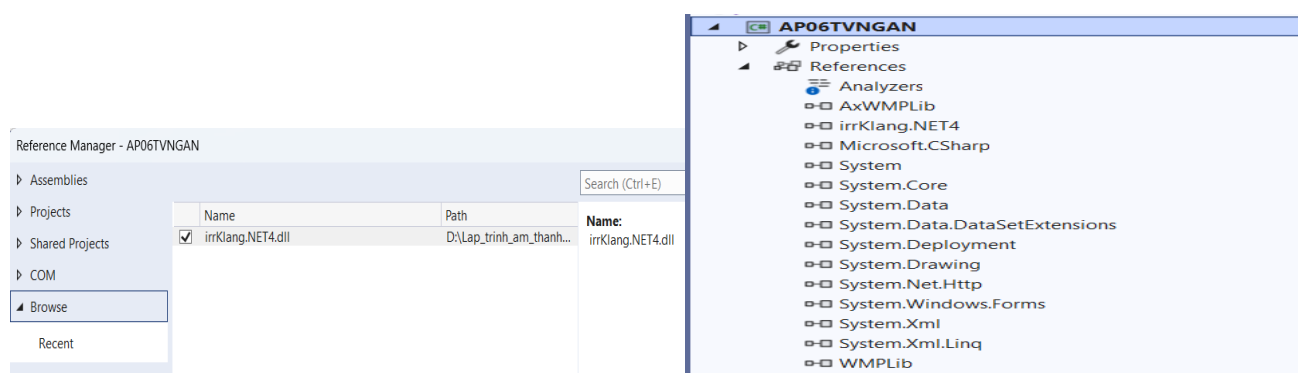
* Các bước nạp thư viện, như hình Hình 3-1:

Bước 1: Chuột phải[Project] => Add → Reference...

Bước 2: Trên hộp thoại xuất hiện:

- + Chọn tab Browse, sau đó
- + Chọn nút Browse..., tìm file thư viện tương ứng
(ví dụ: wmp.dll trong đường dẫn C:\Windows\System32),
- + Đánh dấu tích vào thư viện cần nạp (Ví dụ: wmp.dll) và chọn nút OK

Kết quả nạp các thư viện xử lý âm thanh vào App (Reference), như Hình 3-2 Hình 3-1



Hình 3-1 Nạp các thư viện (wmp.dll, OpenAL32.dll, irrKlang.NET4.dll) xử lý âm thanh

Hình 3-2 Kết quả nạp các thư viện xử lý âm thanh trong References của Audio App

3.2. Nạp BỘ CÔNG CỤ của các thư viện

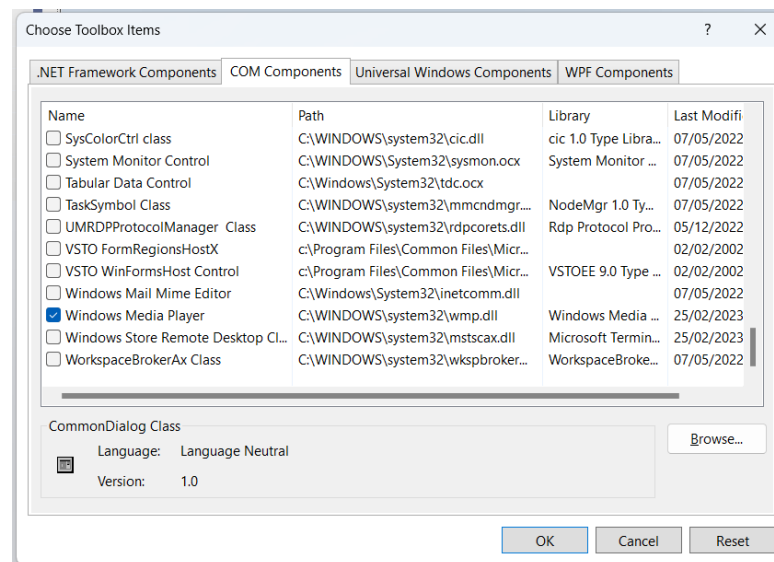
* Bộ công cụ tương ứng các thư viện (thường tự động bổ sung vào Toolbox sau khi nạp thư viện như nêu trên) có thể nạp như sau:

Ví dụ: WMPLib

B1: Trong Toolbox: tên nhóm tool của App: AP06TVNGAN Components

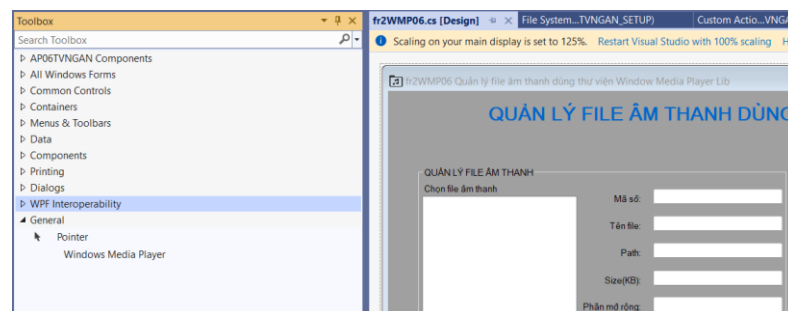
B2: Phải[tên nhóm AP06TVNGAN Components]=>Choose Items...: COM Components

B3: Chọn: Windows Media Player, như Hình 3-3 **Error! Reference source not found.**



Hình 3-3 Nạp công cụ thư viện WMP vào Toolbox

Kết quả là trong Toolbox sẽ có công cụ Windows Media Player, như Hình 3-4



Hình 3-4 Công cụ WMP trong Audio App sau khi nạp

Chương 4: GIỚI THIỆU VÀ CÀI ĐẶT SẢN PHẨM

HỆ THỐNG QUẢN LÝ FILE ÂM THANH CÒI XE

4.1. Mô tả sản phẩm

Tên sản phẩm: **AP06TVNGAN**

**ỨNG DỤNG QUẢN LÝ & XỬ LÝ ÂM THANH: LẬP TRÌNH BỞI
HỌC VIÊN CÔNG NGHỆ BƯU CHÍNH VIỄN THÔNG TP.HCM
HỆ THỐNG QUẢN LÝ FILE ÂM THANH CÒI XE**

Thực hiện bởi: Trần Văn Ngạn - N20DCPT044 – D20CQPTUD01-N

Nền tảng công nghệ: **.NETFramework 4.8.1. MS. Visual Studio .NET 2022**

Dữ liệu lưu trữ thông tin Files âm thanh: MS. SQL Server 2022

Ngôn ngữ lập trình: **C#.NET**

Setup.exe: Bộ cài đặt dùng VSI_Extentions nằm trong phần Extentions (có tên là Microsoft Visual Studio Installer Project)

Thư viện chính: **WMPLib** và **irrKLang**

Tập dữ liệu được tham khảo (thu thập) từ: <https://www.pixabay.com>

4.2. Cài CSDL lưu trữ thông tin âm thanh

4.2.1. Lập DB SQL chỉ 1 Table:

* Sử dụng MS. SQL Server 2022: **AP06TVNGAN**

- ComputerName: NGANCONDETHUONG
- SQL Server Authentication
- UserName: sa
- Password: 123
- Data Source=NGANCONDETHUONG;Initial Catalog=AP06TVNGAN;Persist Security Info=True;User ID=sa;Password=123

* Lập 1 Table: ms, filepath, filename, size, extension, length, description như Hình 4-1

NGANCONDETHUON...dbo.AUDIOFILES			
	Column Name	Data Type	Allow Nulls
▶	ms	nchar(20)	<input type="checkbox"/>
	filename	nchar(100)	<input checked="" type="checkbox"/>
	filepath	nchar(150)	<input checked="" type="checkbox"/>
	size	float	<input checked="" type="checkbox"/>
	ext	nchar(10)	<input checked="" type="checkbox"/>
	length	int	<input checked="" type="checkbox"/>
	description	nvarchar(250)	<input checked="" type="checkbox"/>
			<input type="checkbox"/>

Hình 4-1

4.2.2. Nhập dữ liệu thực nghiệm:

NGANCONDETHUON...dbo.AUDIOFILES							
	ms	filename	filepath	size	ext	length	description
▶	01	044197_201...	d:\AP06TV...	352	.mp3	18	no desc
	02	075714_la-c...	d:\AP06TV...	94	.mp3	3	no desc
	03	086287_car-...	d:\AP06TV...	20	.mp3	1	no desc
	04	182474_ke...	d:\AP06TV...	745	.wav	4	no desc
	05	2000-nissan...	d:\AP06TV...	281	.mp3	14	no desc
	06	20131210_c...	d:\AP06TV...	307	.mp3	16	no desc
	07	20150826_sf...	d:\AP06TV...	32836	.mp3	1681	no desc
	08	402958_ins...	d:\AP06TV...	6527	.wav	25	no desc
	09	allahabad_t...	d:\AP06TV...	7559	.mp3	387	no desc
	10	ambience-a...	d:\AP06TV...	1196	.mp3	61	no desc
	12	berlin_street...	d:\AP06TV...	3617	.mp3	185	no desc

Hình 4-2

4.3. Lập Audio App Winform mới: AP06TVNGAN

* Lập App dạng C#.NET trên MS. Visual Studio: Visual Studio 2022

[File]=>New=>Project; Chọn:

- Visual C#
- Windows Form Application
- "D:\AP06TVNGAN_Coi_Xe\Project\DESIGN_APP "
- AP06TVNGAN
- Phiên bản .NET Framework 4.8.1

4.4. Thiết kế và Lập trình Winform màn hình chính: FormMain06.cs

4.4.1. Thiết kế (Design)

Màn hình chính được thiết kế như Hình 4-3

+ Properties:

. Name: FormMain06

. Text: AUDIO PROGRAM BY 06 TRẦN VĂN NGẠN, D20CQPTUD01-N: PTIT HCM

. No background

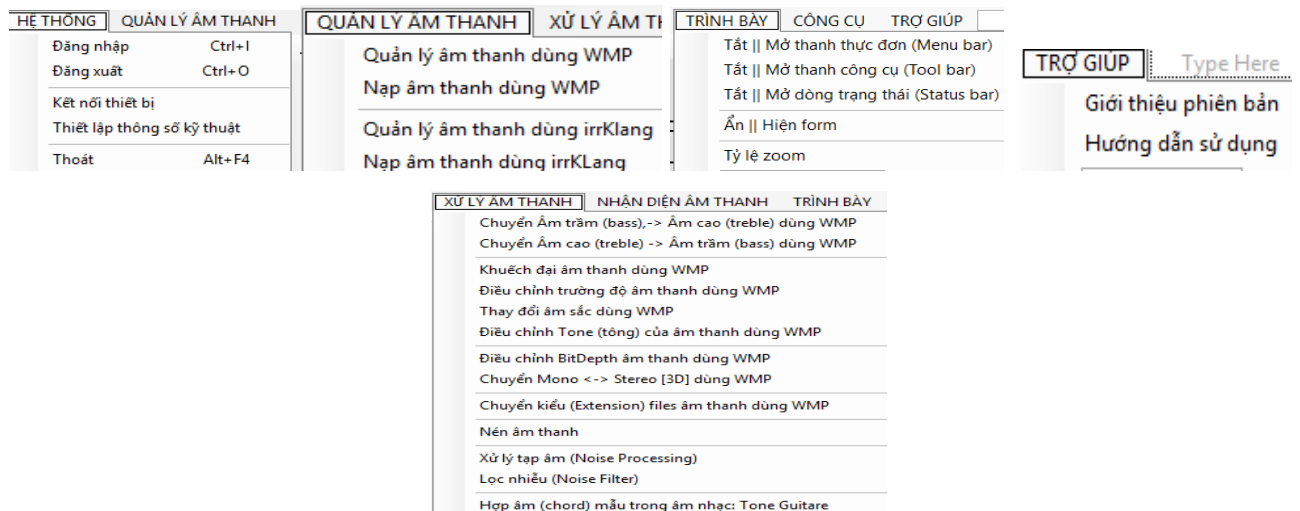


Hình 4-3 Thiết kế màn hình chính của Audio App

. ContextMenu = contextMenuStrip1:

+ Nhóm Menu & Toolbars:

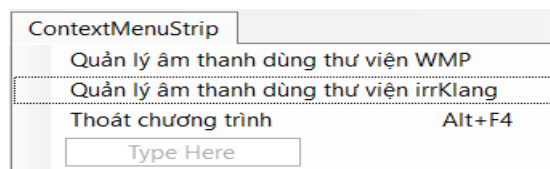
. MainMenu = menuStrip1 với các Menu như Hình 4-4



Hình 4-4 Thiết kế các thực đơn (Menu) của màn hình chính Audio App

. ContextMenu (Menu & Toolbars): Thực đơn click phải, như Hình 4-5

Name = contextMenuStrip1



Hình 4-5 ContextMenu thực đơn click phải trên màn hình chính

. NotifyIcon (Common Controls): Biểu tượng của App xuất hiện trên khay hệ thống (System Tray), tương tự Unikey.

1. Text = AUDIO PROGRAM BY 06 TRẦN VĂN NGẠN, D20CQPTUD01-N: PTITHCM
2. ContextMenu = contextMenuStrip1
3. Visible = True (hiển thị Icon trên SysTray khi App chạy)
4. Icon = chọn file .ico

+ Thiết lập đồng hồ trên dòng trạng thái

. Timer (Componenets): Đối tượng đồng hồ

1. Enable = True: kích hoạt đồng hồ chạy.
2. Interval = 100 (mặc định), tức là 1s nhảy đồng hồ 1 lần (đơn vị %s)

Đề xuất hiện đồng hồ trên dòng trạng thái (StatusBar): Thêm vào Status Bar: Status Label

```

using System;
using System.Collections.Generic;
using System.ComponentModel;
using System.Data;
using System.Drawing;
using System.Linq;
using System.Text;
using System.Threading.Tasks;
using System.Windows.Forms;

namespace AP06TVNGAN
{
    public partial class FormMain06: Form
    {
        public FormMain06()
        {
            InitializeComponent();
        }

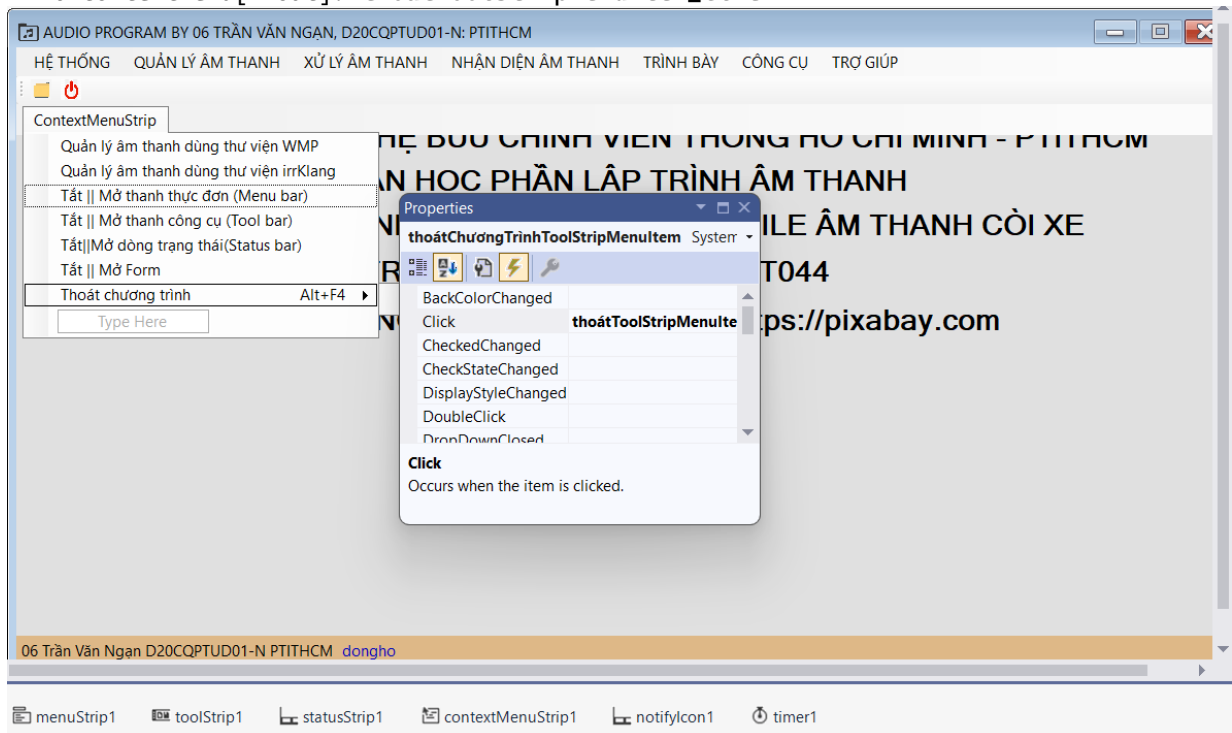
        private void Form1_Load(object sender, EventArgs e)
        {
            timer1_Tick(sender, e);
        }

        private void timer1_Tick(object sender, EventArgs e)
        {
            toolStripStatusLabel2.Text = System.DateTime.Now.ToShortDateString() + ": " +
            System.DateTime.Now.ToLongTimeString();
        }

        private void thoátToolStripMenuItem_Click(object sender, EventArgs e)
        {
            DialogResult ch = MessageBox.Show("Bạn có chắc chắn muốn thoát không?", "Xác nhận",
            MessageBoxButtons.YesNo, MessageBoxIcon.Question);
            if(ch ==DialogResult.Yes)
            {
                Application.Exit();
            }
        }
    }
}

```

GÁN VÀO ContextMenu[Thoát]: thoátToolStripMenuItem_Click



Hình 4-6


```
//GỌI FORM QUẢN LÝ ÂM THANH fr2WMP06.cs dùng WMPLib
private void quảnLýÂmThanhDùngWMPToolStripMenuItem_Click(object sender, EventArgs e)
{
    fr2WMP06 obj = new fr2WMP06();
    obj.ShowDialog();
}

private void quảnLýÂmThanhDùngIrrKlangToolStripMenuItem_Click(object sender, EventArgs e) {
    try{
        fr3irrKLang06 obj = new fr3irrKLang06();
        obj.ShowDialog();
    }
    catch (System.Exception ex)
        MessageBox.Show(ex.Message, "Loi khi mo form");
    }
}
```

GÁN VÀO ContextMenu[Quản lý âm thanh dùng WMP]:quảnLýÂmThanhDùngWMPToolStripMenuItem_Click

```
//TẮT || MỞ THANH THỰC ĐƠN (menu)
private void tắtMởThanhThựcĐơnToolStripMenuItem_Click(object sender, EventArgs e)
{
    menuStrip1.Visible = !menuStrip1.Visible;
}
GÁN VÀO ContextMenu[Tắt || Mở thanh thực đơn (Menu bar)]:tắtMởThanhThựcĐơnToolStripMenuItem_Click
```

```
//TẮT || MỞ THANH CÔNG CỤ (ToolBar)
private void tắtMởThanhCôngCụToolStripMenuItem_Click(object sender, EventArgs e)
{
    toolStrip1.Visible = !toolStrip1.Visible;
}
GÁN VÀO ContextMenu[Tắt || Mở thanh công cụ (Tool bar)]:tắtMởThanhCôngCụToolStripMenuItem_Click
```

```
//TẮT || MỞ THANH DÒNG TRẠNG THÁI ở chân màn hình (Status Bar)
private void tắtMởDòngTrạngTháiStatusBarToolStripMenuItem_Click(object sender, EventArgs e)
{
    statusStrip1.Visible = !statusStrip1.Visible;
}
GÁN VÀO ContextMenu[Tắt||Mở dòng trạng thái(Status bar)]:tắtMởDòngTrạngTháiStatusBarToolStripMenuItem_Click
```

```
//TẮT || MỞ MÀN HÌNH (Form)
private void tắtFormToolStripMenuItem_Click(object sender, EventArgs e)
{
    this.Visible = !this.Visible; //con trỏ This là Forn hiện tại
}
GÁN VÀO ContextMenu[Tắt || Mở Form]:tắtFormToolStripMenuItem_Click
```

```
}//public partial class FormMain06
{//namespace AP06TVNGAN
```

THIẾT LẬP APP CHẠY TỪ FORM FormMain06.cs: trong Program.cs

```
static void Main()
{
    Application.EnableVisualStyles();
    Application.SetCompatibleTextRenderingDefault(false);
    Application.Run(new FormMain06());
}
```

4.5. Thiết kế và Lập trình Winform màn hình quản lý âm thanh dùng WMPLib

fr2WMP06.cs:Màn hình Quản lý âm thanh dùng thư viện Windows Media Player (WMP) là thư viện quản lý và xử lý âm thanh có sẵn trên Hệ điều hành (OS: Operating System) Microsoft Windows.

4.5.1. Thiết kế (Design)

4.5.1.1. Màn hình Quản lý âm thanh

dùng WMP được thiết kế như Hình 4-7, gồm

+ 2 Group Box (Containers):

.Name: groupBoxAudioInfo.Text = QUẢN LÝ FILES ÂM THANH

.Name: groupBoxAudioTest.Text = KIỂM THỬ FILES ÂM THANH

+ Name: groupBoxAudioInfo: Gồm

. listBox1 (Common Controls): Cho phép NSD chọn file âm thanh.

. Các Labels và TextBox hiển thị thông tin chi tiết về File âm thanh

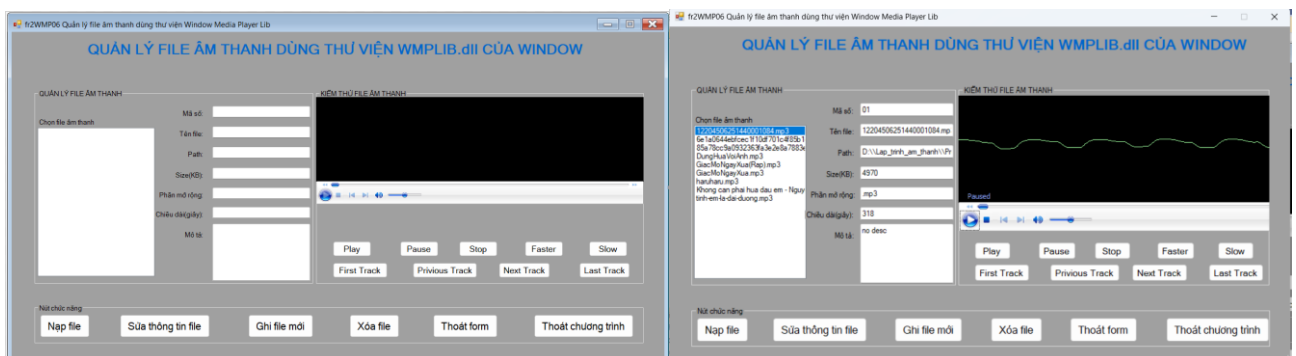
Trong đó: textBox5 = TextBox mô tả thông tin file âm thanh (Description) thuộc tính Multiline = True.

+ Name: groupBoxAudioTest

. Các nút lệnh, thuộc 2 nhóm: Test file âm thanh: Play, Stop, Pause, Faster, Slower

Chọn file test (Track): First, Previous, Next, Last

. Công cụ chạy file âm thanh



Hình 4-7 Thiết kế màn hình Quản lý và xử lý âm thanh dùng WMPLib

4.5.1.2. Kết nối DB vào listBoxAuFile

Với Table đã lập trong 4.2 là dbo.AUDIOFILES, như Hình 4-2

[listBoxAuFile]Task=>chọn: Use DataBound Items

DataSource: Add Project DataSource

Thiết lập chuỗi kết nối DB: New connection [chọn: Save my password]

Chọn: Yes, include sensitive data in the connectionString

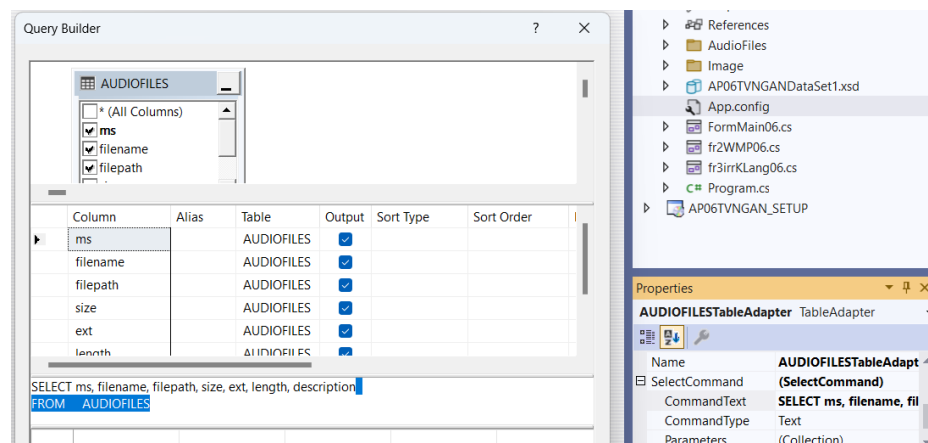
Tên chuỗi và nội dung chuỗi ConnectionString là:

name="AP06TVNGAN.Properties.Settings.AP06TVNGANConnectionString1"

connectionString="Data Source=NGANCONDETHUONG;Initial

Catalog=AP06TVNGAN;Persist Security Info=True;User ID=sa;Password=123"

Chuỗi kết nối này cũng có thể kiểm tra và xem được trong tập tin: **app.config**



Hình 4-8 Thiết lập kết nối DB thông tin file âm thanh vào Listbox

Câu lệnh SQL:

SELECT ms, filename, filepath, size, ext, length, description

FROM AUDIOFILES

[listBoxAudioFile]Task=>chọn:

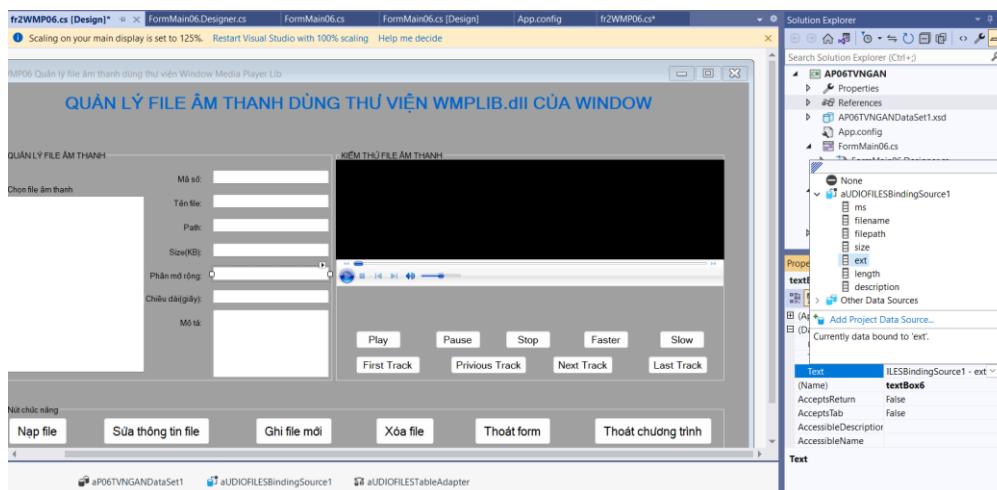
Display Member: filename (tên file âm thanh) và Value Member: filepath (đường dẫn)

+ Với mỗi kết nối DB (bên dưới: Designer) sẽ xuất hiện 3 đối tượng:

...DataSet..., ...BindingSource, ...TableAdapter: tblAudioFileTableAdapter

+ DataBinding các thông tin của files từ ListBox sang các ô TextBox (bên phải) như Hình 4-9

. TextBox..[Properties]=>Data:DataBindings: Text=>Chọn Fields của DB tương ứng



Hình 4-9 DataBinding dữ liệu (VD: Extension) từ DB đã kết vào ListBox vào các TextBox tương ứng.

. Nhập dữ liệu / SQL đủ để thực nghiệm

4.5.1.3. Tổ chức lưu trữ các files âm thanh

+ Lập thư mục lưu chứa các files âm thanh cho App / Solution Explorer

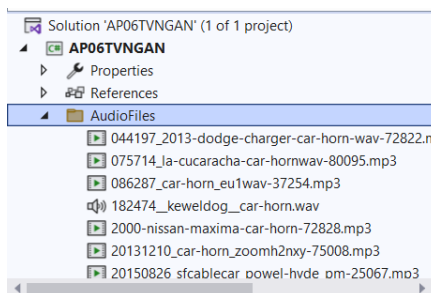
. Lập thư mục mới:

Phải[Project:...]=>Add=>New Folder: đặt tên thư mục, VD: **AudioFiles**

. Nạp các files âm thanh vào thư mục trên

Phải[tên thư mục vừa tạo:AudioFiles]=>Add=>Existing Items: chọn các files âm thanh

Ghi chú: chọn All files (dưới, phải)



Hình 4-10

. Cập nhật dữ liệu các files âm thanh vào Tables / SQL

. Phải[tên thư mục vừa tạo:AudioFiles]=>Open Folder in Files Explorer

Copy đường dẫn vật lý, nơi lưu các files âm thanh dùng để lấy âm thanh copy vào thư mục app và cập nhật đường dẫn file vào DB: filepath /SQL

D:\AP06TVNGAN_Coi_Xe\Project\DESIGN_APP\AP06TVNGAN\AudioFiles

. Cập nhật tên file trong thư mục trên vào DB: filename vào cuối filepath

(KHÔNG nên thay đổi tên các Thư mục của App, vì sẽ ảnh hưởng Path của files)

4.5.2. Một số số mã lệnh ban đầu (Codes)

```
using System;
using System.Collections.Generic;
using System.ComponentModel;
using System.Data;
using System.Drawing;
using System.Linq;
using System.Text;
using System.Threading.Tasks;
using System.Windows.Forms;
using static System.Windows.Forms.VisualStyles.VisualStyleElement.ProgressBar;

namespace AP06TVNGAN
{
    public partial class fr2WMP06: Form{
        public fr2WMP06(){
            InitializeComponent();

            private void fr2WMP06_Load(object sender, EventArgs e){
                // TODO: This line of code loads data into the
                'aP06TVNGANDataSet1.AUDIOFILES' table. You can move, or remove it, as needed.
                try{
                    this.aUDIOFILESTableAdapter.Fill(this.aP06TVNGANDataSet1.AUDIOFILES);
                }
                catch(System.Exception ex) {
                    MessageBox.Show("Có lỗi khi kết nối với database", ex.Message);
                }
            }

            private void fr2WMP06_FormClosing(object sender, FormClosingEventArgs e){
                DialogResult ch = MessageBox.Show("Trở về màn hình chính?", "Xác nhận",
                MessageBoxButtons.YesNo, MessageBoxIcon.Question);
                if (ch == DialogResult.Yes){
                    axWindowsMediaPlayer1.Ctlcontrols.stop();
                }
                else
                    e.Cancel = true;
            }

            private void button14_Click(object sender, EventArgs e){
                DialogResult ch = MessageBox.Show("Trở về màn hình chính?", "Xác nhận",
                MessageBoxButtons.YesNo, MessageBoxIcon.Question);
                if (ch == DialogResult.Yes){
                    this.Close();
                }
            }

            private void button15_Click(object sender, EventArgs e) {
                DialogResult ch = MessageBox.Show("Thoát khỏi form này?", "Xác nhận",
                MessageBoxButtons.YesNo, MessageBoxIcon.Question);
                if (ch == DialogResult.Yes)
                {
                    Application.Exit();
                }
            }
        }
    }
}
```

4.6. Sử dụng công cụ trong Toolbox (đối với một số thư viện có hỗ trợ công cụ)

* Chỉ có thư viện WMPLib và OpenAL là có hỗ trợ công cụ trong ToolBox

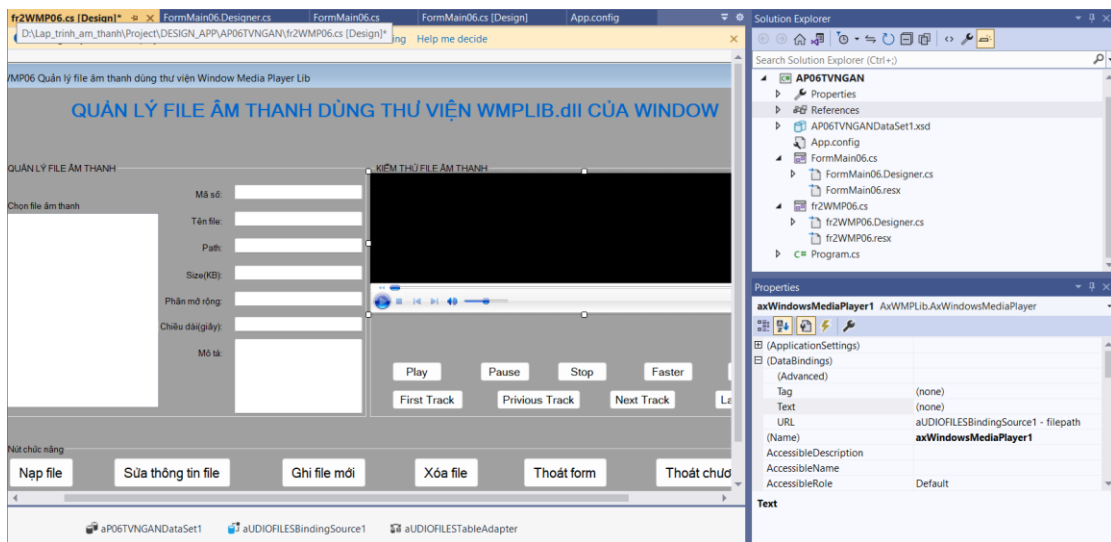
irrKlang chủ yếu lập trình bằng mã lệnh (codes), không hỗ trợ Tool

* Ví dụ: Sử dụng Windows Media Player vào GroupBox: **groupBoxAudioTest**

+ Kéo thả (Drag and Drop) axWindowsMediaPlayer1 vào groupBoxAudioTest

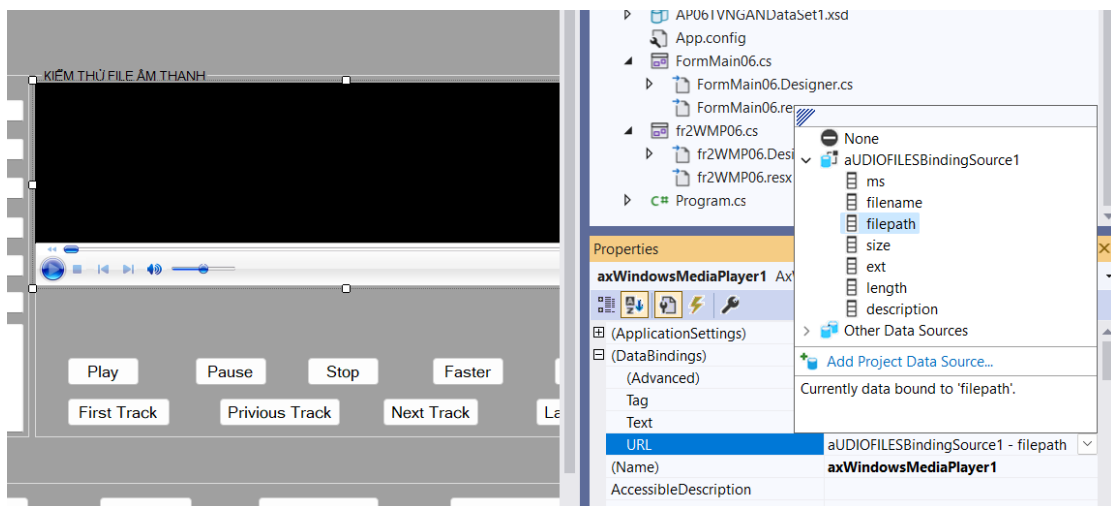
+ Gán Âm thanh vào WMP: Binding từ DB (của ListBox) vào WMP

Properties[WMP]=>Data: DataBinding=>Advanced... như Hình 4-11



Hình 4-11 Binding file âm thanh vào WMP

Chọn Binding: filepath cho Property: URL như Hình 4-12



Hình 4-12 Binding path của file âm thanh vào URL của WMP

4.7. Lập trình (Codes) XỬ LÝ âm thanh sử dụng thư viện

* Với thư viện WMPLib

Bước 1: Khai báo các NameSpace (đường dẫn thư viện) sử dụng Codes

```
using WMPLib;  
using AxWMPLib;
```

Bước 2: Lập trình (Codes) thao tác xử lý âm thanh (WMP)

```
/// <summary>  
/// LẬP TRÌNH THAO TÁC XỬ LÝ ÂM THANH  
/// </summary>  
  
//Play = CHẠY FILE AUDIO FILES  
private void button1_Click(object sender, EventArgs e)  
{  
    axWindowsMediaPlayer1.Ctlcontrols.play();  
}  
//Pause = TẠM DỪNG AUDIO FILES ĐANG CHẠY  
private void button2_Click(object sender, EventArgs e)  
{  
    axWindowsMediaPlayer1.Ctlcontrols.pause();  
}  
//Stop = DỪNG AUDIO FILES ĐANG CHẠY  
private void button3_Click(object sender, EventArgs e)  
{  
    axWindowsMediaPlayer1.Ctlcontrols.stop();  
}  
//Faster = CHẠY AUDIO FILES NHANH HƠN (do tìm)  
private void button4_Click(object sender, EventArgs e)  
{  
    axWindowsMediaPlayer1.Ctlcontrols.fastForward();  
}  
// chạy chậm lại  
private void button5_Click(object sender, EventArgs e)  
{  
    if (axWindowsMediaPlayer1.settings.rate == 1)  
        axWindowsMediaPlayer1.settings.rate = 0.75;  
    if (axWindowsMediaPlayer1.settings.rate == 0.75)  
        axWindowsMediaPlayer1.settings.rate = 0.5;  
    if (axWindowsMediaPlayer1.settings.rate == 0.5)  
        axWindowsMediaPlayer1.settings.rate = 1;  
    // axWindowsMediaPlayer1.Ctlcontrols.  
}  
//CHẠY TRACK KẾ TIẾP; NẾU TRACK1 CUỐI => VỀ TRACK ĐẦU  
private void button8_Click(object sender, EventArgs e)  
{  
    if (listBox1.SelectedIndex < listBox1.Items.Count - 1)  
        listBox1.SelectedIndex++; //Nếu chưa phải là Track cuối ( < vị trí cuối: count -1)  
        //thì tăng vị trí Track lên 1(++) đến Track kế tiếp  
    else listBox1.SelectedIndex = 0; //nếu đang là Track cuối (count - 1) thì  
        // chuyển về Track đầu tiên(vị trí 0)  
}  
//CHẠY TRACK KẾ TRƯỚC; NẾU TRACK ĐẦU TIÊN => VỀ TRACK CUỐI  
private void button7_Click(object sender, EventArgs e)  
{  
    if (listBox1.SelectedIndex > 0) listBox1.SelectedIndex--;  
    //Nếu chưa phải là Track ĐẦU ( > 0) thì GIẢM vị trí Track XUỐNG 1 (--) đến Track kế TRƯỚC  
    else listBox1.SelectedIndex = listBox1.Items.Count - 1; //nếu đang là Track ĐẦU  
    //(0) thì chuyển XUỐNG Track CUỐI CÙNG(vị trí vị trí cuối: count - 1)  
}  
//VỀ TRACK ĐẦU TIÊN  
private void button6_Click(object sender, EventArgs e)  
{  
    listBox1.SelectedIndex = 0;  
}  
//XUỐNG TRACK ĐẦU CUỐI CÙNG  
private void button9_Click(object sender, EventArgs e)  
{  
    listBox1.SelectedIndex = listBox1.Items.Count - 1;  
}
```

4.8. Lập trình (Codes) QUẢN LÝ âm thanh sử dụng thư viện

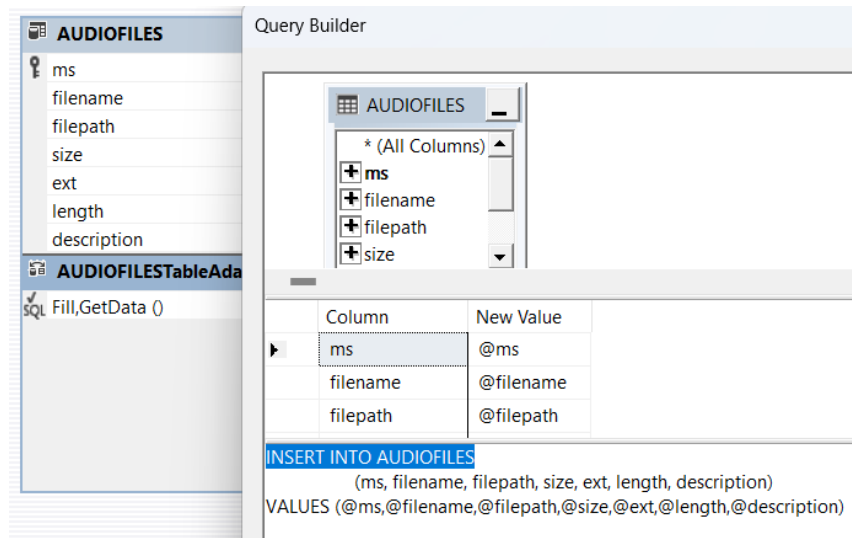
4.8.1. Giới thiệu

- * **Quản lý files âm thanh** bao gồm các thủ tục, như:
 - + Nạp [Load] thêm file âm thanh vào danh sách (ListBox) và lưu vào DataBase
 - + Sửa [Modify] chứa thông tin của file âm thanh trong Database SQL (đã lưu)
 - + Xóa [Delete] file âm thanh khỏi danh sách (Listbox)
 - + Ghi âm [Record] file âm thanh mới từ microphone sẽ lưu DataBase.
- * Với thư viện WMPLib (đã nạp như phần trên), và khai báo các NameSpace trước lập trình (Codes)
Khai báo các NameSpace (như trên)
- * Với mỗi thao tác quản lý files âm thanh (như nêu trên): Load, Modify, Delete, Record thông thường phải thực hiện qua 2 bước cơ bản:
 - + Lập câu lệnh SQL tương ứng để cập nhật thông tin files âm thanh vào DataBase SQL
 - Load, Record = Insert....Into...
 - Modify = Update....Set....Where....
 - Delete = Delete from....
 - + Lập trình (C#.NET) tương ứng các nút lệnh và gọi sử dụng các câu lệnh SQL tương ứng như nêu trên: Insert, Update, Delete.
Trong mã lệnh lập trình của mỗi thủ tục: Load, Modify, Record, Delete gồm 2 bước:
 - . Gọi câu lệnh SQL tương ứng = Insert, Upadate, Delete
 - . Tải dữ liệu lên Form (ListBox, . . .) sau khi xử lý xong = ...Fill(...) như ..Form_Load()

4.8.2. Nạp thêm file âm thanh (Load)

Bước 1: Viết câu lệnh SQL: Insert(...), như Hình 4-13

```
INSERT INTO dbo.AUDIOFILES (ms, filename, filepath, size, ext, length, description)
VALUES (@ms,@filename,@filepath,@size,@ext,@length,@description)
```

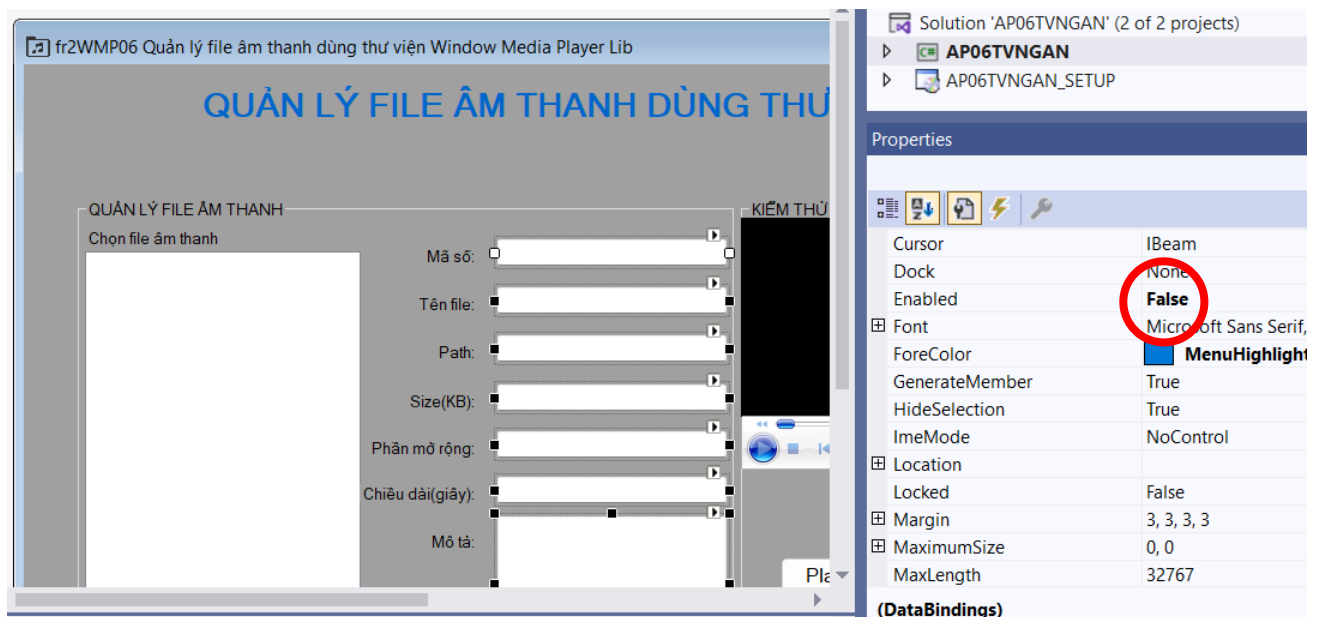


Hình 4-13 Câu lệnh SQL: Insert = nạp thêm files âm thanh vào App

Bước 2: Lập trình (Codes) thao tác Nạp File âm thanh (Load)

2.1. Sử dụng công cụ trong ToolBox=>Dialogs: OpenFileDialog: **openFileDialog1**

2.2. Trạng thái ban đầu các ô TextBox thông tin Files âm thanh phải .Enable = False, như Hình 4-14, tức là NSD không được phép sửa thông tin files âm thanh.



Hình 4-14 Gán .Enable = False các TextBox thông tin các Files âm thanh

2.3. Viết mã lệnh (Codes) cho thủ tục nạp file âm thanh: btnLoad_Click(...)

+ Thủ tục này gồm các công việc (bước) như sau:

B1: Đóng || Mở = Cho || cấm các textbox thông tin & nút lệnh các khác:

....Enabled = !....Enabled;

Ngoại trừ các textbox tự động lấy từ file âm thanh chọn nạp từ OpenFileDialog.

B2: Nút lệnh “Nạp thêm files”/btnLoad có 2 trạng thái

- Bắt đầu nạp file âm thanh = cho NSD

> chọn file âm thanh từ máy tính cá nhân nhờ vào OpenFileDialog... (if...) &

> nhập các thông tin cần thiết về file âm thanh vào các textbox tương ứng[ms và desc]

```
if (button10.Text == "Nạp file") { //Mở OpenFileDialog lên cho NSD chọn file âm thanh / PC
    dongYNapFile = openFileDialog1.ShowDialog();
    //biến toàn cục ch đã khai báo phía trên, giữ lại nút lệnh (Y|N) mà NSD đã chọn để xử
    lý trong ...else..

    if(dongYNapFile == DialogResult.OK)
    {
        textBox1.Text = "Quý vị phải nhập mã số file vào đây";
        textBox5.Text = ""; // Xóa mô tả để NSD nhập mô tả mới
        button10.Text = "Lưu"; //Đổi nhãn nút này thành "Lưu...": tự nhập

        textBox1.Enabled = !textBox1.Enabled; // được phép nhập mã số nhưng không được nhập
        textBox5.Enabled = !textBox5.Enabled;

        string tenfile = System.IO.Path.GetFileName(openFileDialog1.FileName);
        //tên file âm thanh mà NSD đã chọn
        textBox2.Text = tenfile;
        textBox3.Text = audiopath + tenfile;
        FileInfo info = new FileInfo(audiopath + tenfile);
        textBox4.Text = (Math.Round((double)(info.Length) / 1024)).ToString();
        textBox6.Text = info.Extension;

        var player = new WindowsMediaPlayer();
        var clip = player.newMedia(audiopath + tenfile);
        textBox7.Text = clip.duration.ToString();
    }
}
```

- Khi chọn file và nhập thông xong, NSD sẽ bấm nút “Lưu file âm thanh”

cũng chính là nút lệnh này => vì vậy phải đổi nhãn (.Text) thông qua lệnh if... else....

```
if (button10.Text == "Nạp thêm files")
{ //Mở OpenFileDialog lên cho NSD chọn file âm thanh / PC
    Code như trên
}
else //Sau khi NSD chọn file & nhập thông tin xong
{
    Thực hiện code nạp file
    Button10.Text = "Nạp thêm files"; //Trả lại nhãn ban đầu ["Nạp thêm files" copy từ Design sang, KHÔNG tự nhập]
}
}
```

B3: Nạp file âm thanh = “Lưu file âm thanh” trong ...else.. (nêu trên)

string tenfile = System.IO.Path.GetFileName(openFileDialog1.FileName); //tên file âm thanh mà NSD đã chọn

GỒM 2 BƯỚC CƠ BẢN:

+ COPY FILE ÂM THANH VÀO THƯ MỤC CỦA chứa data của APP

```
try
{
    if (!Directory.Exists(dataDirectoryPath))
    {
        Directory.CreateDirectory(dataDirectoryPath);
    }

    System.IO.File.Copy(filePath, Path.Combine(dataDirectoryPath, fileName), true);
}
catch (System.Exception ex) { MessageBox.Show("Có lỗi copy file âm thanh:" + ex.Message); }
```

+ Update THÔNG TIN FILE ÂM THANH VÀO DB

```
//[2] NẠP THÔNG TIN CỦA FILE ÂM THANH ĐÃ CHỌN VÀO DATABASE SQL
if (textBox1.Text != "" && textBox1.Text != "Quý vị phải nhập mã số file vào đây")
//NSD bắt buộc nhập mã số file âm thanh thì mới lưu được.
{
    try
    {
        aUDIOFILESTableAdapter.Insert(textBox1.Text.Trim(), fileName, Path.Combine(dataDirectoryPath, fileName), fileSize, fileExtension,
fileDuration, textBox5.Text);

        //NẠP THÔNG TIN FILE ÂM THANH VÀO DB [1 TRONG 2 CÂU LỆNH QUANG TRỌNG]

    }
    catch (System.Exception ex)
    { MessageBox.Show("Có lỗi khi nạp thông tin file âm thanh vào DB: " + ex.Message); }
}
```

B4: Tải file âm thanh vừa nạp lên App bằng việc Copy từ thủ tục ...Load...(…) ở trên

```
try
{
    this.aUDIOFILESTableAdapter.Fill(this.aP06TVNGANDDataSet1.AUDIOFILES);
    //nạp DB files âm thanh
}
catch (System.Exception ex)
{ MessageBox.Show("Có lỗi tải các files âm thanh lên danh sách! " + ex.Message); }
```

ĐẦY ĐỦ MÃ LỆNH CỦA THỦ TỤC NẠP FILE ÂM THANH VÀO APP (FULL CODES)

```
if (button10.Text == "Nạp file") { //Mở OpenFileDialog lên cho NSD chọn file âm thanh / PC
    dongYNapFile = openFileDialog1.ShowDialog();

    if(dongYNapFile == DialogResult.OK)
    {
        textBox1.Text = "Quý vị phải nhập mã số file vào đây";
        textBox5.Text = ""; // Xóa mô tả để NSD nhập mô tả mới
        button10.Text = "Lưu"; //Đổi nhãn nút này thành "Lưu...": tự nhập

        textBox1.Enabled = !textBox1.Enabled; // được phép nhập mã số nhưng không được nhập trung mã số
        textBox5.Enabled = !textBox5.Enabled;

        // khởi tạo giá trị cho các biến lưu thông tin file audio
        filePath = openFileDialog1.FileName;
        fileName = System.IO.Path.GetFileName(filePath);

        fileInfo = new FileInfo(filePath);
        fileSize = (Math.Round((double)(fileInfo.Length) / 1024));
        fileExtension = fileInfo.Extension;
        player = new WindowsMediaPlayer();
        audio = player.newMedia(filePath);
        fileDuration = (int)Math.Round(audio.duration);

        textBox2.Text = fileName;
        textBox3.Text = filePath;
        textBox4.Text = fileSize.ToString();
        textBox6.Text = fileExtension;
        textBox7.Text = fileDuration.ToString();

        Console.WriteLine("Dung dan file vua chon la: " + filePath);
    }
}
else if(button10.Text == "Lưu")
{
    if (dongYNapFile == DialogResult.OK) //NSD đã đồng ý nạp file âm thanh đã chọn từ OpenFileDialog
    {
        Console.WriteLine(filePath + " " + dataDirectoryPath);

        try
        {
            if (!Directory.Exists(dataDirectoryPath))
```

```

    {
        Directory.CreateDirectory(dataDirectoryPath);
    }

    System.IO.File.Copy(filePath, Path.Combine(dataDirectoryPath, fileName), true);
}
catch (System.Exception ex) { MessageBox.Show("Có lỗi copy file âm thanh:" + ex.Message); }

//[2] NẠP THÔNG TIN CỦA FILE ÂM THANH ĐÃ CHỌN VÀO DATABASE SQL
if (textBox1.Text != "" && textBox1.Text != "Quý vị phải nhập mã số file vào đây")
//NSD bắt buộc nhập mã số file âm thanh thì mới lưu được.
{
    try
    {
        AUDIOFILESTableAdapter.Insert(textBox1.Text.Trim(), fileName,
        Path.Combine(dataDirectoryPath, fileName), fileSize, fileExtension, fileDuration, textBox5.Text);
        //NẠP THÔNG TIN FILE ÂM THANH VÀO DB [1 TRONG 2 CÂU LỆNH QUANG TRỌNG]
    }
    catch (System.Exception ex)
    { MessageBox.Show("Có lỗi khi nạp thông tin file âm thanh vào DB: " + ex.Message); }
}
else MessageBox.Show("Không được để trống mã file");

try
{
    this.aUDIOFILESTableAdapter.Fill(this.aP06TVNGANDDataSet1.AUDIOFILES);
}
catch (System.Exception ex)
{ MessageBox.Show("Có lỗi tải các files âm thanh lên danh sách! " + ex.Message); }

button10.Text = "Nạp file";
} //NSD OK
} //Lưu file

```

4.8.3. Sửa thông tin file âm thanh (Modify)

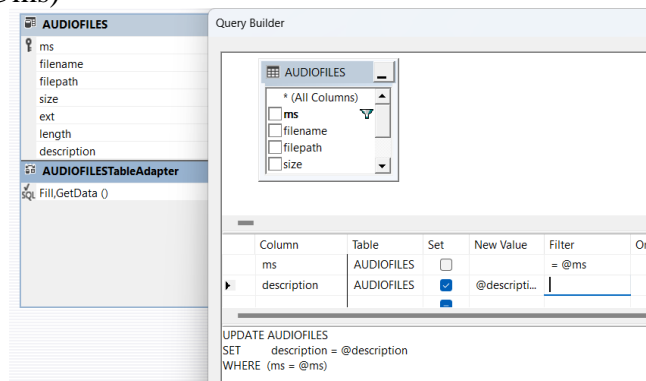
* Chỉ sửa được Mô tả.

* Bước 1: Viết câu lệnh SQL: Update(...)

```

UPDATE    AUDIOFILES
SET        description = @description
WHERE (ms = @ms)

```



Hình 4-15 Câu lệnh SQL: Update = sửa thông files âm thanh đã có trong App

* Bước 2: Lập trình (Codes) thao tác Sửa thông tin File âm thanh (Modify):

ĐẦY ĐỦ MÃ LỆNH CỦA THỦ TỤC SỬA THÔNG TIN FILE ÂM THANH ĐÃ CÓ TRONG APP (FULL CODES)

```

//SỬA THÔNG TIN FILE ÂM THANH ĐÃ CÓ TRONG APP
private void button11_Click(object sender, EventArgs e) // SỬA THÔNG TIN FILE ÂM THANH
{
    //B1: Cho || cấm các nút lệnh các khác và textbox mô tả tin file âm thanh (không sửa được các thông khác)
    textBox1.Enabled = !textBox1.Enabled; // được phép nhập mã số nhưng không được nhập trung mã số đã có
    textBox5.Enabled = !textBox5.Enabled;
}

```

```

button1.Enabled = !button1.Enabled; // button play
button2.Enabled = !button2.Enabled; // button pause
button3.Enabled = !button3.Enabled; // button stop
button4.Enabled = !button4.Enabled; // button faster
button5.Enabled = !button5.Enabled; // button slow
button6.Enabled = !button6.Enabled; // button first track
button7.Enabled = !button7.Enabled; //
button8.Enabled = !button8.Enabled;
button9.Enabled = !button9.Enabled;
button10.Enabled = !button10.Enabled;
button12.Enabled = !button12.Enabled;
button13.Enabled = !button13.Enabled;
//B2: ĐỔI NHÃN (.TEXT) CỦA NÚT LỆNH: "Sửa..." <=> "Lưu..."
if (button11.Text == "Sửa thông tin file")
//bắt đầu Sửa thông tin file âm thanh trong các TextBox ["Sửa thông tin files" copy từ
Design sang, KHÔNG tự nhập]
{ //Thông báo nhắc NSD cách sửa thông tin
    MessageBox.Show("Quý vị chỉ được sửa mô tả file"); //thông báo hướng dẫn NSD cách sửa
    thông tin file

    button11.Text = "Lưu"; //Đổi nhãn (.Text) thành2 "Lưu...": tự nhập
}
else //Sau khi NSD sửa thông tin xong => Lưu thông tin file âm thanh sau sửa vào DB
{ //B3: SỬA THÔNG TIN FILE ÂM THANH + Update THÔNG TIN SỬA VÀO DB
    try
    {
        aAUDIOFILESTableAdapter.Update(textBox5.Text, textBox1.Text.Trim());
        //Lưu THÔNG TIN FILE ÂM THANH sau Sửa VÀO DB [CÂU LỆNH QUANG TRỌNG]
    }
    catch (System.Exception ex)
    { MessageBox.Show("Có lỗi khi SỬA thông tin file âm thanh: " + ex.Message); }

    //B4: Tải file mới vừa nạp vào ListBox và nghe thử / axWMP = COPY xuống TỪ trên
    FrWMPManagement_Load(..)
    try
    {
        this.aAUDIOFILESTableAdapter.Fill(this.aP06TVNGANDDataSet1.AUDIOFILES);
        // nạp DB files âm thanh
    }
    catch (System.Exception ex)
    { MessageBox.Show("Có lỗi tải các files âm thanh lên danh sách! " + ex.Message); }
    //B5: Đổi nhãn thành ban đầu
    button11.Text = "Sửa thông tin files";
    //Trả lại nhãn ban đầu ["Sửa thông tin files" copy từ Design sang, KHÔNG tự nhập]
} //else Lưu thông tin file vào DB
} //button11// SỬA THÔNG TIN FILE ÂM THANH

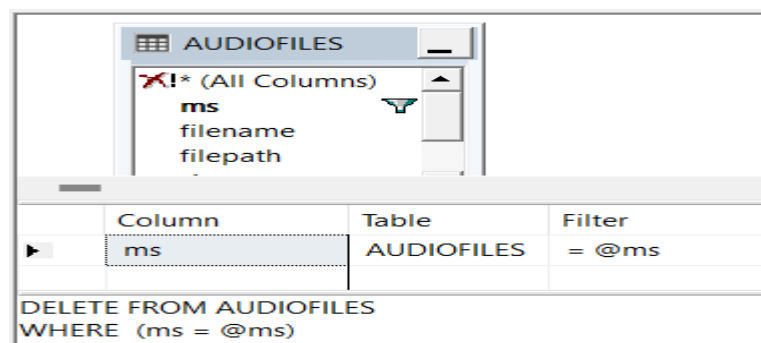
```

4.8.4. Ghi âm file âm thanh (Record)

4.8.5. Xóa file âm thanh (Delete)

* Bước 1: Viết câu lệnh SQL: Delete(...)

DELETE FROM AUDIOFILES WHERE (ms = @ms)



Hình 4-16

* Bước 2: Lập trình (Codes) thao tác Xóa File âm thanh (Delete):

ĐẦY ĐỦ MÃ LỆNH CỦA THỦ TỤC XÓA FILE ÂM THANH CÓ TRONG APP (FULL CODES)

```
private void Button13_Click(object sender, EventArgs e)
{
    axWindowsMediaPlayer1.Ctlcontrols.pause();
    //B1: HỎI XÁC NHẬN
    DialogResult ch = MessageBox.Show("Bạn có muốn xóa file: " +
    textBox2.Text.Trim() + " không(Y/N)?", "Xác nhận",
    MessageBoxButtons.YesNo,
    MessageBoxIcon.Question);
    //B2: XÓA Khi NSD Y
    if (ch == DialogResult.Yes)//NSD đồng ý Xóa
    {
        try
        {
            AUDIOFILESTableAdapter.Delete(textBox1.Text.Trim()); //XÓA FILE ÂM
            THANH [CÂU LỆNH QUANG TRỌNG]
        }
        catch (System.Exception ex)
        {
            MessageBox.Show("Có lỗi khi XÓA file âm thanh: " + ex.Message);
        }
        try //B3: Tải các file sau xóa lên ListBox = COPY xuống TỪ trên
        FrWMPManagement_Load(..)
        {
            this.AUDIOFILESTableAdapter.Fill(this.aP06TVNGANDDataSet1.AUDIOFILES); // nạp DB files âm
            thanh
        }
        catch (System.Exception ex)
        {
            MessageBox.Show("Có lỗi tải các files âm thanh lên danh sách! " +
            ex.Message);
        }
        } //If NSD Y

        axWindowsMediaPlayer1.Ctlcontrols.play();
    } //button13// XÓA FILE ÂM THANH
```

4.9. Lập trình (Codes) XỬ LÝ âm thanh sử dụng thư viện irrKlang

Việc thiết kế và lập trình quản lý và xử lý âm thanh dùng thư viện irrKlang về nguyên tắc là tương tự như thư viện WMPLib, nên trong phần này chỉ minh họa cách dùng thư viện irrKlang.

Tên Form: Fr3irrKLang06.cs

4.9.1. Thiết kế Forms

Màn hình được thiết kế như Hình 4-17, gồm:

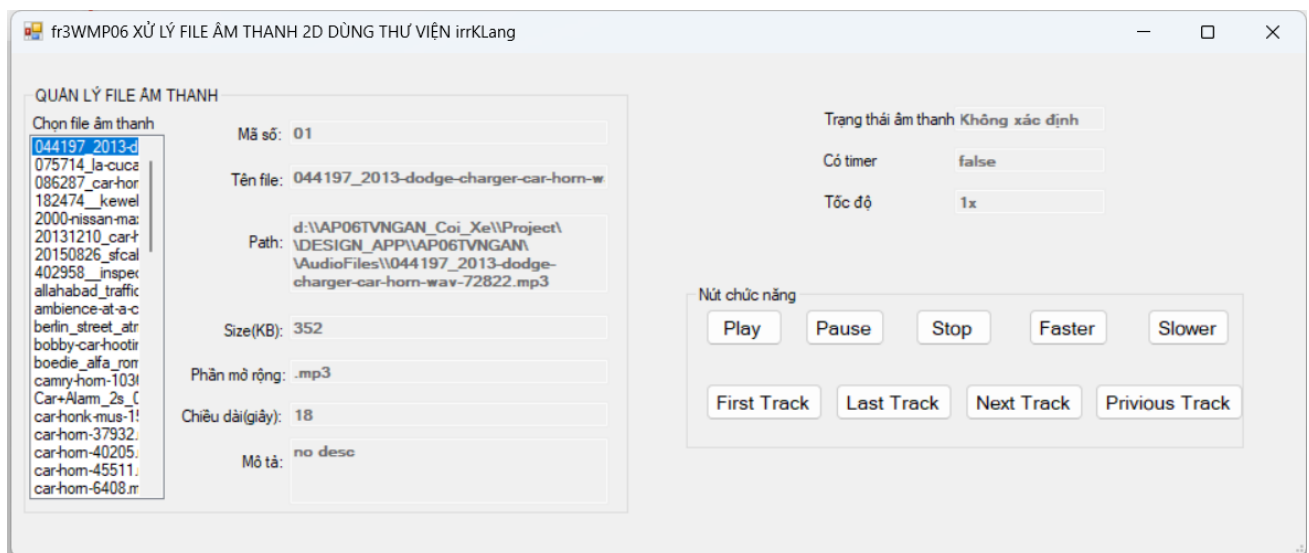
. Text = XỬ LÝ FILE ÂM THANH 2D DÙNG THƯ VIỆN irrKlang

. Gồm hai phần chính:

+) Hiện thị danh sách file âm thanh và thông tin chi tiết(ở phần Quản lý âm thanh bên trái màn hình)

+) Xử lý file âm thanh với một số chức năng cơ bản(play, pause, faster, next track... bên phải màn hình).

. Phần thiết lập thuộc tính của các nút chức năng, liên kết data tương tự như của form Fr2WMP06 (đã được trình bày ở bên trên).



Hình 4-17 Thiết kế màn hình minh họa Play Sound dùng thư viện irrKlang

4.9.2. Lập trình (Codes)

Xem trong file fr3irrKlang06.cs trong Solution winform đính kèm.

4.10. Thiết kế form đăng nhập

Form này dùng để nhập thông tin server muốn kết nối.

Cần đảm bảo rằng server kết nối có bảng **AUDIOFILES** như yêu cầu ban đầu.

Chức năng: + Đăng nhập và lưu thông tin kết nối

+ Thoát chương trình

* **Khắc phục** tình trạng server trên máy NSD có thông tin khác với thông tin server mà app yêu cầu. NSD chỉ cần đảm bảo database có bảng AUDIOFILES giống như đã được đề cập ở mục 4.2.1

Mã code xem trong project đính kèm.

Chương 5: ĐÓNG GÓI BỘ CÀI ĐẶT SETUP CỦA SẢN PHẨM ĐỀ TÀI

5.1. Giới thiệu chung

Phần App được lập trình nêu trên là phần Source Code (mã nguồn) do tác giả lưu giữ nhằm đảm bảo bản quyền và phục vụ việc nâng cấp, cải tiến và phát triển chương trình sau này. Để có thể chuyển App đến người dùng thì phải thực hiện thủ tục đóng gói và chuyển giao sản phẩm đến người dùng (Gọi là: Package and Deployment) hoặc lập bộ Setup.

Người sử dụng sẽ dùng bộ Setup (thông thường có file Setup.exe hoặc Install.exe) để cài đặt vào máy tính làm việc của mình. Sau khi cài đặt, sản phẩm phần mềm sẽ lưu trong thư mục **C:\Program Files\TVNGAN\AP06TVNGAN_SETUP** đồng thời sẽ có biểu tượng (icon) trên Desktop và Start Menu của người dùng để tiện sử dụng.

Sản phẩm sau khi cài đặt trên máy tính người dùng (tức là, trong **C:\Program Files\TVNGAN\AP06TVNGAN_SETUP**) là sản phẩm mã máy. Không thể đọc hiểu theo dạng văn bản (text: mở trên NotePad), kể cả tác giả. Về nguyên tắc là không dịch ngược mã máy về mã code lập trình (dạng văn bản).

Tóm lại: trình tự sản phẩm phần mềm từ sản xuất (lập trình) đến người dùng, thông thường phải qua 3 bước:

- Lập trình Source Codes
- Đóng thành bộ Setup.exe (nhiệm vụ chương này)
- Cài lên máy tính người dùng (**C:\Program Files\TVNGAN\AP06TVNGAN_SETUP**):
NSD dùng phần mềm

5.2. Công cụ sử dụng thiết lập bộ cài đặt Setup cho App

Bộ công cụ hỗ trợ lập bộ cài đặt (Setup/Install) của các phần mềm WinApp: **VSI_extension.exe** (Microsoft Visual Studio Installer Project): Gói dịch vụ cài đặt bổ sung vào MS. Visual Studio .NET hỗ trợ thiết lập bộ cài đặt (Setup) dạng đơn giản.

5.3. Thiết lập bộ cài đặt Setup cho App

Phải mở source winapp trên Visual Studio .NET, đảm bảo chạy được: không lỗi.

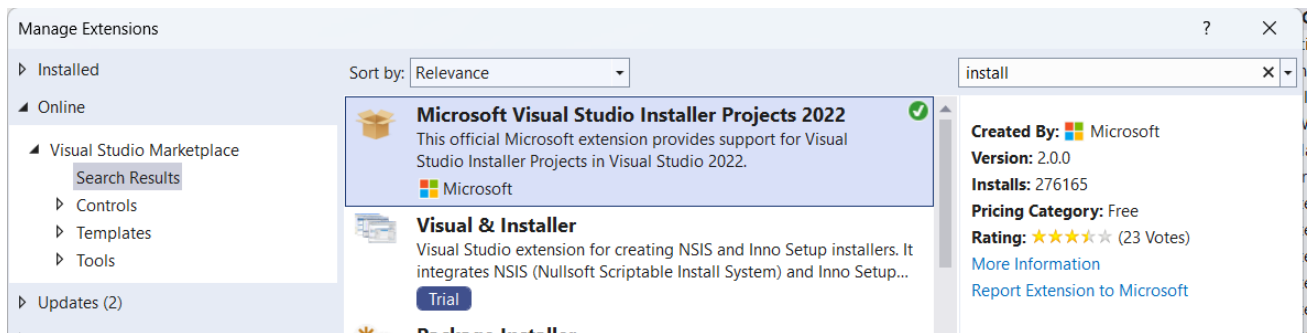
Bước 1: THÊM PROJECT MỚI VÀO SOLUTION HIỆN TẠI, Như

+ Chuột phải vào tên Solution => Add => New Project

+ Installed: Other Project Types: Visual Studio Installer (Lập bộ cài đặt đơn giản):

+ Setup Project

Như Hình 5-1



Hình 5-1. Thiết lập các thông tin cho Project Setup

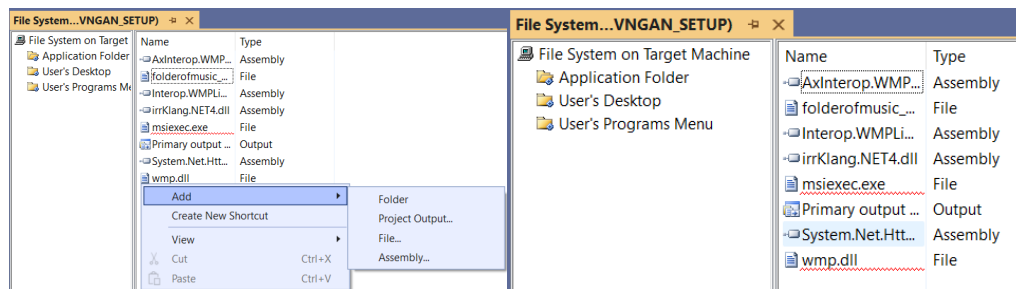
Bước 2: THIẾT LẬP CÁC THÀNH PHẦN KẾT QUẢ CÀI ĐẶT, như Error! Reference source not found.

- Application Folder: Thư mục của phần mềm trong %OS%\Program Files...
- User's Desktop: Dùng để cài đặt shortcut trên nền Desktop
- User's Program Menu: Dùng để cài đặt hiển thị app trong menu Start của Window

1. Application on Target Machine

Chuột phải [vùng trống] => Add=>Project Output: Primary Output (Active)

Tương tự đối với msixexec.exe (dùng để gỡ cài đặt ứng dụng)



2. User's Desktop:

Chuột phải [vùng trống] => Create New Shortcut => Đổi tên [Tên của app]

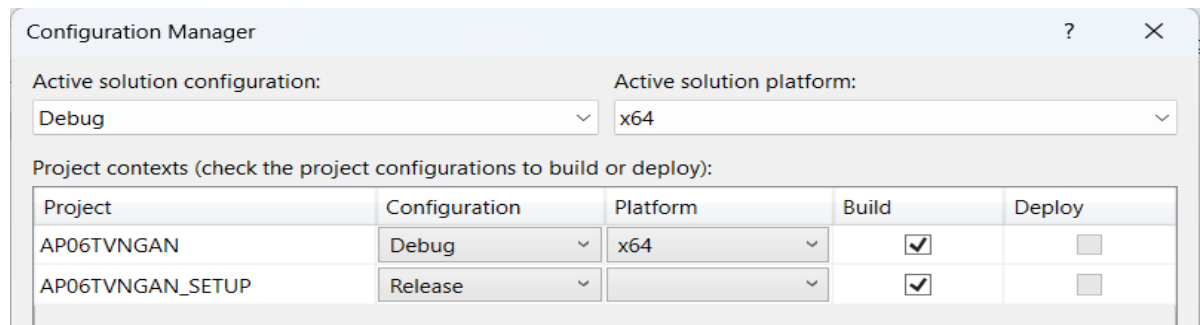
Trong properties, chỉnh sửa nội dung thuộc tính Target thành Primary output, và hai thuộc tính khác: Description, Icon.

3. User's Program Menu: Làm tương tự User's Desktop

Bước 3: CẤU HÌNH BIÊN DỊCH, như Hình 5-2

Chuột phải vào tên Solution => Configuration Manager: [x] Setup [Build]

Với Release (thương mại) || Debug (thử nghiệm)



Hình 5-2. Thiết lập biên dịch bộ Setup

Bước 4: BIÊN DỊCH BỘ CÀI ĐẶT

- Thực hiện chọn [Build]=>(Re)Build Solution

- Sản phẩm bộ cài trong thư mục:

**D:\AP06TVNGAN_Coi_Xe\Project\DESIGN_APP\AP06TVNGAN_SETUP\Release
 \AP06TVNGAN_SETUP.msi** (vì đã được chọn trong Bước 3 là Release).

- Nếu chọn Debug

**D:\AP06TVNGAN_Coi_Xe\Project\DESIGN_APP\AP06TVNGAN_SETUP\Debug\
 AP06TVNGAN_SETUP.msi**

Bước 5: THIẾT LẬP THÊM CÁC ĐẶC TÍNH KHÁC CỦA BỘ CÀI ĐẶT

Đặt chọn vào vị trí Project Setup / Solution Explorer:

Sử dụng thanh công cụ ngay trên Solution Explorer

+ File System Editor: Thiết lập các kết quả cài đặt / máy tính NSD

+ User's Interface Editor: Thiết kế các màn hình chờ trong quá trình cài đặt PM

Install: Client (NSD)

Administrative Install: Server

Start: Bắt đầu cài đặt

Progress: Trong quá trình cài đặt

End: kết thúc việc cài đặt

Có thể: thêm / xóa bớt (Thay đổi thứ tự) các cửa sổ màn hình chờ cài đặt

Thay đổi = thiết kế lại (VD: Việt hóa): sử dụng cửa sổ Properties

+ Custom Action: Thiết kế các thành phần hỗ trợ khác của bộ cài đặt ;

GHI CHÚ: CÁC THUỘC TÍNH KHÁC CỦA PM

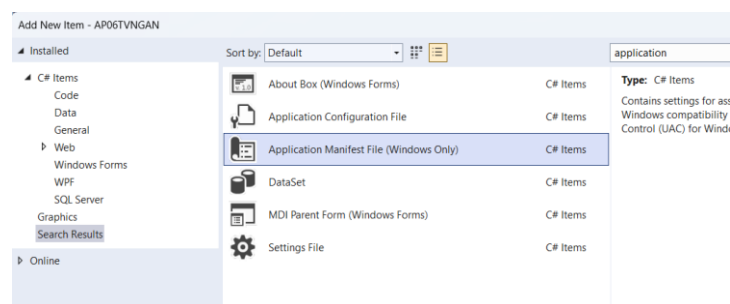
[Solution Explorer]: Properties=>AssemblyInfo.cs

+ Phải cài đặt DB riêng => khai báo lại chuỗi kết nối trong file .XML kèm theo sau khi cài đặt

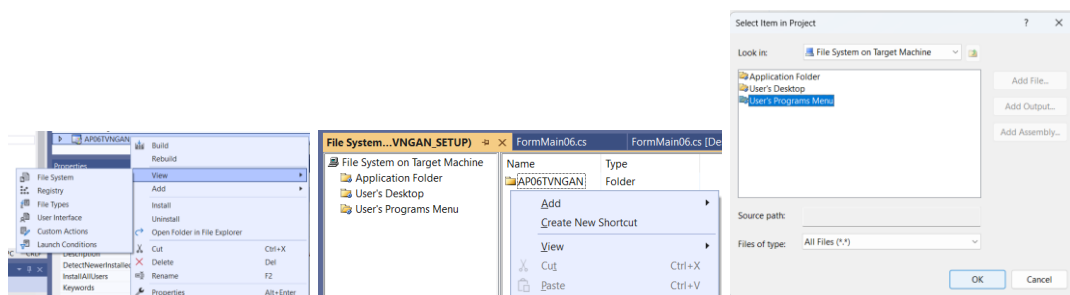
Author	Trần Văn Ngạn
BackwardCompatibleIDGeneration	False
Description	Chương trình quản lý âm thanh sử dụng thư viện Windo
DetectNewerInstalledVersion	True
InstallAllUsers	False
Keywords	
Localization	English (United States)
Manufacturer	TVNGAN
ProductName	AP06TVNGAN_SETUP
RemovePreviousVersions	True
RunPostBuildEvent	On successful build
SearchPath	
Subject	
SupportPhone	
SupportUrl	
TargetPlatform	x64
Title	AP06TVNGAN_SETUP

Bước 6: Thiết lập một số đặc tính của app

- Thiết lập app yêu cầu NSD cần chạy dưới quyền quản trị(điều này cần thiết cho một số chức năng của app)
- Chuột phải vào Tên project => Add => New item => Application Manifest File
- Mở file app.manifest và chỉnh sửa lại tag requestedExecutionLevel với nội dung như sau
`<requestedExecutionLevel level="requireAdministrator" uiAccess="false" />`



2.



Bước 7: CHUYỂN GIAO BỘ CÀI ĐẶT: SETUP CHO NSD

Khi cần tháo gỡ (Uninstall) phần mềm thì phải thực hiện trong:

+ Control Panel: Programs and Features

+ Không được Xóa thư mục phần mềm %OS%\Program Files...

Vì nếu “Xóa” sẽ không cài đặt lại được (khi cài đặt sẽ bị báo là đã có => chỉ có thể khắc phục bằng cách Xóa các Key trong Registry)

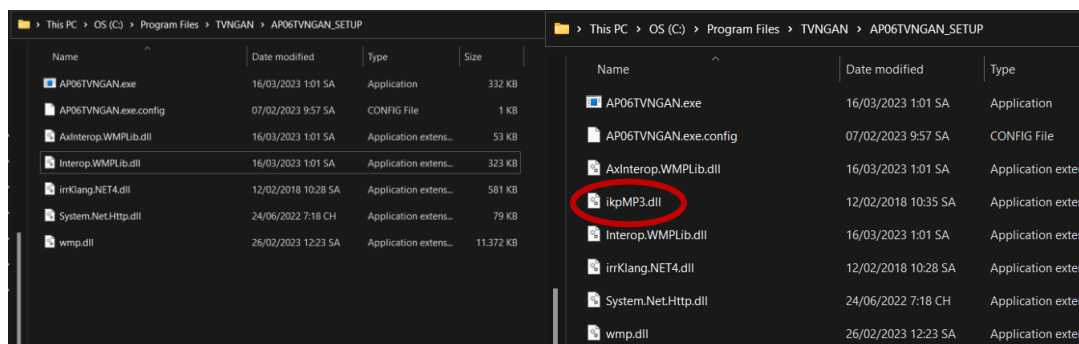
CHÚ Ý: Phần mềm mới được cài đặt sẽ phát sinh những vấn đề sau:

1. App có đường dẫn C:\Program Files\TVNGAN\AP06TVNGAN_SETUP

2. Ứng dụng mới cài đặt lần đầu sẽ không có file nhạc mẫu

NSD muốn có file nhạc mẫu để chạy thử thì cần copy thư mục **AUDIO_DATA** đính kèm file source vào thư mục có đường dẫn **C:\Program Files\TVNGAN**. Nếu không thì NSD không thể sử dụng file nhạc có sẵn trên app => phải thêm file nhạc mới. Lưu ý rằng nếu không thực hiện copy, NSD nên xóa các file nhạc không thể sử dụng đó (bằng giao diện của app) để tránh nhầm lẫn sau này.

3. Chức năng xử lý nhạc bằng thư viện irrKlang không hoạt động. Để xử lý cần làm như sau: Copy file **ikpMP3.dll** vào trong thư mục có đường dẫn **C:\Program Files\TVNGAN\AP06TVNGAN_SETUP** => Kết quả:



- **Lưu ý rằng** ikpMP3.dll cần được build bởi x64 platform (quay lại bước 3 của mục 5.3 để biết rõ hơn cách cài đặt

Chương 6: KẾT LUẬN VÀ HƯỚNG PHÁT TRIỂN CỦA ĐỀ TÀI

6.1. Kết luận

6.1.1. Những kết quả đạt được

Đề tài đã:

+ Tìm hiểu về âm thanh học và kỹ thuật âm thanh

+ Thiết kế và lập trình được sản phẩm dạng WinApp minh họa của đề tài dùng C#.NET trên công cụ Visual Studio.NET với thư viện WMPLib.dll, bao gồm:

. FormMain06.cs: Màn hình chính có dùng Menu, Toolbal, Status, NotifyIcon, ContextMenu

. Fr2WMP06.cs: Form quản lý và xử lý âm thanh dùng thư viện WMPLib, đây là form chính của sản phẩm đề tài, bao gồm:

- Xử lý âm thanh: Play, Pause, Stop, Faster, Slower
- Quản lý âm thanh: Nạp, Sửa, Xóa

. Fr3irrKLang06.cs:

+ Tập âm thanh mẫu thực nghiệm: Bao gồm các file dạng .wav, .mp3 được tham khảo từ <https://pixabay.com>

6.1.2. Hạn chế

Do thời gian hạn chế, nên App sản phẩm của đề tài còn một số hạn chế sau:

- + Phần chức năng ghi âm: chưa thực hiện
- + Phần irrKLang chưa hoàn thiện
- + Chưa dùng một số thư viện khác: như OpenAL

6.2. Hướng phát triển

6.2.1. Hướng khắc phục các hạn chế

Sẽ tiếp tục thực hiện:

- + Phần chức năng ghi âm sẽ bổ sung sau
- + Phần irrKLang sẽ hoàn thiện
- + Nghiên cứu ứng dụng thêm thư viện OpenAL

6.2.2. Hướng mở rộng đề tài

Sẽ nghiên cứu và phát triển app sản phẩm trên Python dùng các thư viện mở.

DANH MỤC TÀI LIỆU THAM KHẢO

- [1]. Võ Xuân Thê (2021), *Bài giảng học phần Lập trình âm thanh*, Học viện Công nghệ Bưu chính – Viễn thông, cơ sở TP.HCM (lưu hành nội bộ).
- [2]. <http://www.ambiera.com/irrklang>: Thư viện irrKlang
- [3]. <https://dotnet.microsoft.com/> và <https://dotnet.microsoft.com/apps/aspnet>: Công nghệ ASP.NET
- [4]. <https://visualstudio.microsoft.com/downloads>: Các bộ công cụ hỗ trợ của Microsoft
- [5]. <https://docs.microsoft.com/en-us>: Các tài liệu hướng dẫn lập trình của Microsoft (tiếng Anh)
- [6]. <https://www.ambiera.com/irrklang/docunet/index.html>