ĐẠI HỌC QUỐC GIA THÀNH PHỐ HỒ CHÍ MINH

**TRƯỜNG ĐẠI HỌC CÔNG NGHỆ THÔNG TIN**

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

**BÁO CÁO**

ĐỒ ÁN MÔN HỌC LTTQ

# MỤC LỤC

[MỤC LỤC 1](#_Toc483658963)

[A. THÔNG TIN CHUNG 2](#_Toc483658964)

[A1. Tên đồ án 2](#_Toc483658965)

[A2. Thời gian thực hiện 2](#_Toc483658966)

[A3. Nhân lực tham gia 2](#_Toc483658967)

[B. MÔ TẢ ĐỒ ÁN 2](#_Toc483658968)

[B1. Giới thiệu đề tài 2](#_Toc483658969)

[B2. Tổng quan về chương trình 3](#_Toc483658970)

[**I.** **Xây dựng chương trình** 3](#_Toc483658971)

[**II.** **Chương trình gồm các Panel** 3](#_Toc483658972)

[**1.** **Panel Listen** 4](#_Toc483658973)

[**a.** **Back:** để trở về màn hình trước đó. 5](#_Toc483658974)

[**b.** **Now playing:** xuất ra danh bài hát hiện đang được phát thông qua trình phát nhạc Window Media Player của hệ điều hành. 5](#_Toc483658975)

[**c.** **Open:** mở một hoặc nhiều bài hát từ thư mục trong máy. 6](#_Toc483658976)

[**d.** **Home:** trở về màn hình chính của chương trình. 7](#_Toc483658977)

[**e.** **Menu chuột phải Now Playing:** 7](#_Toc483658978)

[**2.** **Panel Playlist** 10](#_Toc483658979)

[**a.** **Back, Nowplaying, Home** tương tự như **panel Listien.** 10](#_Toc483658980)

[**b.** **Tạo Playlist:** tạo mới 1 playlist. 10](#_Toc483658981)

[**c.** **Menu chuột phải Playlist:** 12](#_Toc483658982)

[**d.** **Menu chuột phải My Music:** tương tự như **panel Listen.** 13](#_Toc483658983)

[**3.**  **Panel Search** 14](#_Toc483658984)

[**a.** **Back, Nowplaying, Home** tương tự như **panel Listien.** 14](#_Toc483658985)

[**b.** **Location:** chọn đường dẫn thư mục chứa file nhạc. 14](#_Toc483658986)

[**c.** **Menu chuột phải Location:** 16](#_Toc483658987)

[**d.** **Menu chuột phải Sample Music:** tương tự như **panel Listen.** 16](#_Toc483658988)

[**4.** **Panel Karaoke** 17](#_Toc483658989)

[**a.** **Karaoke:** xuất ra video karaoke thông qua https://www.youtube.com. 18](#_Toc483658990)

[**b.** **Favorite:** danh sách yêu thích. 19](#_Toc483658991)

[**c.** **Menu chuột phải Karaoke:** 21](#_Toc483658992)

[**d.** **Menu chuột phải Favorite:** 22](#_Toc483658993)

[**5.** **Panel Power** 22](#_Toc483658994)

[**a.** **Close:** hẹn thời gian tắt chương trình. 23](#_Toc483658995)

[**b.** **Sleep:** hẹn thời gian cho máy ngủ. 23](#_Toc483658996)

[**c.** **Shuwdown:** hẹn thời gian tắt máy tính. 23](#_Toc483658997)

[**d.** **Pause:** tạm dừng lệnh đang thực thi. 24](#_Toc483658998)

[**e.** **Resume:** tiếp tục thục thi. 25](#_Toc483658999)

[**f.** **Cancel:** hủy lệnh đang thực thi. 25](#_Toc483659000)

[**e.** **Nowplaying, Home** tương tự như **panel Listien.** 25](#_Toc483659001)

[**III.** **Tài liệu tham khảo** 26](#_Toc483659002)

1. THÔNG TIN CHUNG

## A1. Tên đồ án

* Tên tiếng Việt: Nhạc Hay
* Tên tiếng Anh: Nice Player

## A2. Thời gian thực hiện

4tháng

## A3. Nhân lực tham gia

*(Ghi các thành viên trong nhóm)*

|  |  |  |
| --- | --- | --- |
| **TT** | **MSSV** | **Họ tên** |
| 1 | 15520679 | Nguyễn Trung Quân |
| 2 | 14520799 | Phan Hoàng Tâm |
| 3 | 15520629 | Bùi Huy Phúc |

1. MÔ TẢ ĐỒ ÁN

## B1. Giới thiệu đề tài

* Chương trình nghe nhạc (**Nice Player**). Lấy ý tưởng từ chương trình nghe nhạc nổi tiếng Windows Media Player trên HDH Windows
* Sử dụng ngôn ngữ C# để xây dựng chượng trình. Sử dụng thư viện HtmlAgilityPack
* Các chức năng cơ bản:
* Chơi nhạc offline trên windowns
* Load file nhạc từ thư viện có sẵn
* Tạo mới playlist, lưu bài hát vào danh sách yêu thích
* Trộn bài (shuttle), lặp lại (repeat) một hoặc nhiều bài hát
* Tìm kiếm nhạc karaoke online trên trang:  [https://www.youtube.com](%20https://www.youtube.com%20)
* Tìm kiếm lời bài hát online trên trang: http://chiasenhac.vn
* Tìm kiếm nhạc trên file nhạc được chọn
* Các playlist(danh sách bài hát yêu thích) được quản lý riêng rất tiện lợi,kèm theo 1 số chức năng tiện lợi như : tìm đến thư mục chứa bài hát, xem thông số và lời bài hát, chuyển đổi qua lại giữa các bài hát trong playlist, …
* Chức năng hẹn giờ tắt máy, rất tiện lợi cho người dùng có thói quen nghe nhạc thường xuyên hoặc chuẩn bị đi ngủ



**GIAO DIỆN CHƯƠNG TRÌNH NICE PLAYER**

## B2. Tổng quan về chương trình

1. **Xây dựng chương trình**

* Để có thể vận hành và chơi 1 file media trong 1 ứng dụng **.NET** thì bắt buộc phải sử dụng 1 thư viện bên ngoài hoặc lời gọi hệ thống**.** Do đó vấn đề trước tiên đối với 1 chương trình nghe nhạc, đó là lựa chọn một thư viện để có thể giải mã và thực thi 1 file media. Hiện nay, có khá nhiều thư viện hỗ trợ cho công việc này như **thư viện wmm.dll, thư viện Quarzt, thư viện wmp.dll, thư viện MCI, …**
* Chương trình nghe nhạc của chúng em sử dụng thư viện wmp.dll do HDH Windows hỗ trợ. Lí do chúng em chọn thư viện wmp.dll cho chương trình của mình là vì thư viện này hỗ trợ các định dạng file media rất đa dạng bao gồm cả audio và video, thao tác nhanh và tiện lợi, không bị ảnh hưởng bởi các file có dấu… Một lí do quan trọng khác nữa là phần mềm nghe nhạc nổi tiếng Windows Media Player hiện cũng đang sử dụng thư viện này. Qua đó cho thấy những tính năng ,ưu điểm của nó đối với việc giải mã và thực thi 1 file media
* Chương trình nghe nhạc **“Nice Player”** được xây dựng trên 1 namespace duy nhất là **“player”,** bao gồm các thành phần cơ bản:
* **Player** là form chứa giao diện chính, thực thi các chức năng của chương trình, được thiết kế theo từng Panel riêng với các chức năng riêng biệt
* Form **Playlist** được thiết kế để đặt tên cho playlist
* Class **Media** thực hiện tất cả chức năng của chương trình

1. **Chương trình gồm các Panel**

****

Power

Karaoke

Search

Playlist

Listen

* Listen: nghe nhạc từ những bài hát được load từ file bài hát
* Playlist: tạo danh sách những bài hát yêu thích
* Search: tìm kiếm bài hát trong file bài hát được chọn
* Karaoke: tìm kiếm bài hát trực tuyến và hỗ trợ hát karaoke thông qua trang wed http://www.youtube.com
* Power: hẹn giờ tắt máy
* Sau khi cài đặt thư viện wmp.dll, ta thao tác với các file media thông qua đối tượng axWindowsMediaPlayer1

1. **Panel Listen**

****

1. **Back:** để trở về màn hình trước đó.

pbBack.Click += (sender, args) =>

{

if (rtbLyric.Visible == true) // Nếu đang hiển thị lyric => Ẩn rtbLyric & Hiện lvPlaying

{

rtbLyric.Visible = false;

rtbLyric.Text = "Please wait a moment ...";

}

else

pListen.SendToBack(); // Trở về giao diện chính

};

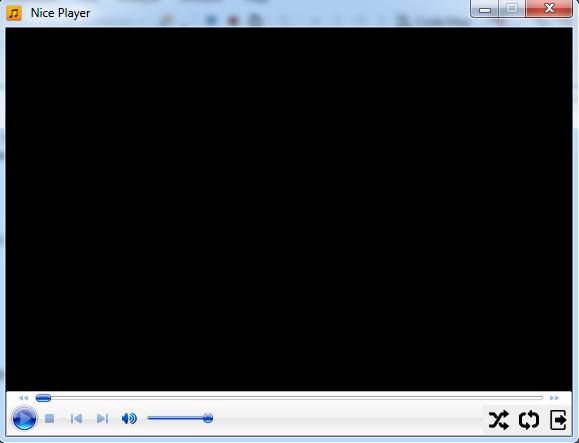
1. **Now playing:** xuất ra danh bài hát hiện đang được phát thông qua trình phát nhạc Window Media Player của hệ điều hành.

pbPlayer.Click += (sender, args) =>

{

wmp.BringToFront(); // Hiển thị Player

};

****

Các button trên **panel Nowplaying:**

|  |  |
| --- | --- |
|  | Có chức năng tắt âm thanh |
|  | Play 1 file media |
|  | Tạm dừng 1 file media |
|  | Tắt 1 file media đang mở |
| C:\Users\Trung Quan\AppData\Local\Microsoft\Windows\INetCache\Content.Word\Shuffle.png | Xáo trộn thứ tự chơi file media ban đầu |
| C:\Users\Trung Quan\AppData\Local\Microsoft\Windows\INetCache\Content.Word\Repeat.png | Lặp lại danh sách nhạc đang chơi |
|  | Tới bài kế tiếp bài đang chơi trong danh sách |
|  | Quay lại bài trước bài đang chơi trong danh sách |
| C:\Users\Trung Quan\AppData\Local\Microsoft\Windows\INetCache\Content.Word\Return.png | Trở về màn hình giao diện trước đó |

* Các button sử dụng các lời gọi hệ thống do thư viên wmp.dll cung cấp, thông qua biến axWindowsMediaPlayer.
* Button C:\Users\Trung Quan\AppData\Local\Microsoft\Windows\INetCache\Content.Word\Shuffle.png xây dựng trên ý tưởng dùng 1 biến kiểu Random để cho ra vị trị bất kỳ của 1 file media tiếp theo file media đang phát. Tiếp theo thực hiện xáo trộn vị trí các file trong danh sách đang chơi với vị trí mà hàm Random tạo ra.

1. **Open:** mở một hoặc nhiều bài hát từ thư mục trong máy.

pbOpen.Click += (sender, args) =>

{

OpenFileDialog ofd = new OpenFileDialog();

ofd.Filter = "Media Files | \*.mp3; \*.wav; \*.flac; \*mpg; \*.mp4; \*.mkv; \*.vob";

ofd.Multiselect = true;

if (ofd.ShowDialog() == DialogResult.OK)

{

lvPlaying.Clear();

gbPlaying.Text = "Now Playing";

rtbLyric.Visible = false;

rtbLyric.Text = "Please wait a moment ...";

playlist.clear();

dirMedia.Clear();

foreach (string item in ofd.FileNames)

{

lvPlaying.Items.Add(Path.GetFileName(item));

WMPLib.IWMPMedia media = wmp.newMedia(item);

playlist.appendItem(media);

dirMedia.Add(item);

}

wmp.currentPlaylist = playlist;

}

};

1. **Home:** trở về màn hình chính của chương trình.

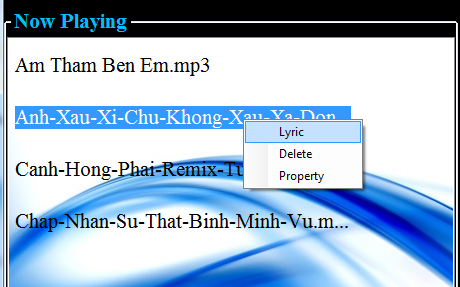
pbHome.Click += (sender, args) =>

{

pListen.SendToBack();

rtbLyric.Visible = false;

};

1. **Menu chuột phải Now Playing:**

* Lyric: xem lời bài hát: thông qua nguồn <http://search.chiasenhac.vn>
* Lấy tên bài hát được chọn và xử lý chuỗi:

string index = listView.FocusedItem.Text.Substring(0,listView.FocusedItem.Text.Length - 4);

index.Replace("-", " ").Replace(" ", "+");

HtmlWeb web = new HtmlWeb();

HtmlAgilityPack.HtmlDocument doc = new HtmlAgilityPack.HtmlDocument();

// Load HTML chứa các link phù hợp với tên media

doc = web.Load("http://search.chiasenhac.vn/search.php?s=" + index);

* Lấy link bài hát được chọn:

string link = string.Empty;

foreach (HtmlNode node in doc.DocumentNode.SelectNodes("//a[@href]"))

{

If (node.Attributes["href"].Value.StartsWith("http://m1"))

{

link = node.Attributes["href"].Value;

break;

}

}

* Load và chỉnh sửa html:

string lyric = string.Empty;

string[] pattern = new string[] { @"<span[^>]\*>[\s\S]\*?</span>", @"&quot;" };

Regex regex = new Regex(string.Join("|", pattern));

string HTML = doc.DocumentNode.InnerHtml;

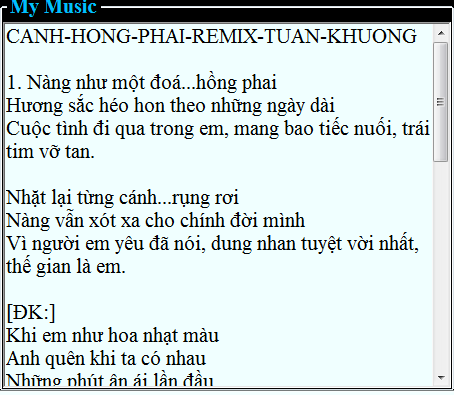
HTML = regex.Replace(HTML, "\n");

doc.LoadHtml(HTML);

* Lấy lời bài hát:

HtmlNode node= doc.DocumentNode.SelectSingleNode("//p[@class='genmed']");

lyric = node.InnerText;

* Hiển thị lời bài hát:
* Delete: xóa bài hát đó trong danh sách nhạc
* Xóa trên listview:

listView.Items.Remove(listView.FocusedItem);

* Xóa khỏi danh sách nhạc đang phát:

foreach (string item in dirMedia)

{

if (Path.GetFileName(item) == listView.FocusedItem.Text)

{

dirMedia.Remove(item);

break;

}

}

* Add To: thêm bài hát vào danh sách bài hát yêu thích
* Duyệt xem media sẽ thêm đã tồn tại trong danh sách đã chọn chưa:

string line = string.Empty;

FileStream stream = new FileStream(@"Playlist\" + args.ClickedItem.Text + ".txt", FileMode.Open);

StreamReader reader = new StreamReader(stream, Encoding.UTF8);

// Duyệt xem media sẽ thêm có tồn tại trong playlist không

while ((line = reader.ReadLine()) != null)

{

if (Path.GetFileName(line) == listView.FocusedItem.Text)

{

available = true; // Media đã tồn tại

break;

}

}

* Lưu vào danh sách đã chọn media đó nếu media chưa tồn tại trong danh sách:

if (!available)

{

// Lưu đường dẫn của media đó vào playlist

foreach (string item in dirMedia)

{

if (Path.GetFileName(item) == listView.FocusedItem.Text)

{

File.AppendAllText(@"Playlist\" + args.ClickedItem.Text + ".txt", item + Environment.NewLine);

break;

}

}

}

* Property: hiển thị thông số về bài hát được chọn
* Tên media:

string title = file.Name.Substring(0, file.Name.Length - 4);

* Thời lượng:

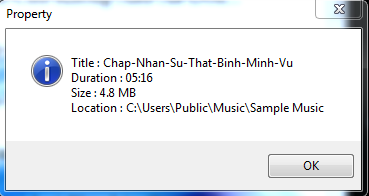
string duration = wmp.newMedia(item).durationString;

* Vị trí:

string directory = file.DirectoryName;

* Kích thước:

float size = (float)file.Length / (1024 \* 1024);



1. **Panel Playlist**
2. **Back, Nowplaying, Home** tương tự như **panel Listien.**
3. **Tạo Playlist:** tạo mới 1 playlist.

****

pbNewPlaylist.Click += (sender, args) =>

{

OpenFileDialog ofd = new OpenFileDialog();

ofd.Filter = "Media Files | \*.mp3; \*.wav; \*.flac; \*mpg; \*.mp4; \*.mkv; \*.vob";

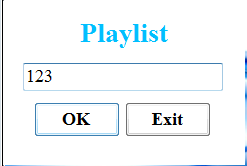
ofd.Multiselect = true;

if (ofd.ShowDialog() == DialogResult.OK)

{

Retry: // Hiển thị lại form rename nếu tên playlist đã tồn tại

Playlist playlistName = new Playlist(); // Mở form điền tên playlist

****

if (playlistName.ShowDialog() == DialogResult.OK)

{

try

{

FileStream stream = new FileStream(@"Playlist\" + playlistName.getName() + ".txt", FileMode.CreateNew);

StreamWriter writer = new StreamWriter(stream, Encoding.UTF8);

foreach (string item in ofd.FileNames)

{

writer.WriteLine(item); // Lưu danh sách bài hát vào file .txt

}

writer.Close();

stream.Close();

lvPlaylist.Items.Add(playlistName.getName()); // Hiển thị playlist mới lên khung lvPlaylist

MessageBox.Show("Done", string.Empty, MessageBoxButtons.OK, MessageBoxIcon.Information); // Thông báo tạo mới thành công

}

catch

{

// Thông báo tên playlist đã tồn tại => Nhập một tên khác

DialogResult result = MessageBox.Show("The playlist has already existed", string.Empty, MessageBoxButtons.RetryCancel, MessageBoxIcon.Error);

if(result == DialogResult.Retry)

goto Retry; // Mở form điền tên playlist khác

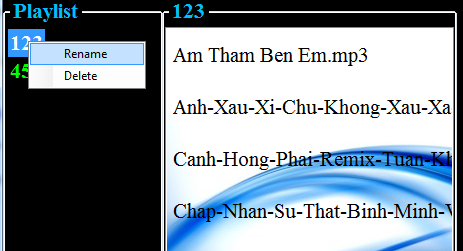
}

}

}

};

1. **Menu chuột phải Playlist:**



* Rename: đổi tên Playlist
* Cho phép đổi tên:

bool loop = false; // Giải quyết lỗi AfterLabelEdit

listView.LabelEdit = true; // Cho phép rename

listView.FocusedItem.BeginEdit(); // Bắt đầu rename

* Sau khi đổi tên kiểm tra tính hợp lệ:

if (!loop) // Nếu loop = false <=> Chưa có lỗi loop

{

// Nếu tên mới rỗng

if (evt.Label == string.Empty)

{

MessageBox.Show("Please enter a name !", "Warning", MessageBoxButtons.OK, MessageBoxIcon.Warning);

evt.CancelEdit = true;

}

// Nếu tên mới đã có sẵn

else if (evt.Label != null && !Directory.GetFiles(@"Playlist", "\*.txt").Contains(@"Playlist/" + evt.Label + ".txt"))

{

try // Rename

{

File.Move(@"Playlist/" + listView.FocusedItem.Text + ".txt", @"Playlist/" + evt.Label + ".txt");

}

catch

{

// Thông báo lỗi nếu không thể rename

MessageBox.Show("The name already existed !", "Error", MessageBoxButtons.OK, MessageBoxIcon.Error);

evt.CancelEdit = true;

}

}

}

loop = true;

* Delete: xóa Playlist
* Xóa trên listview:

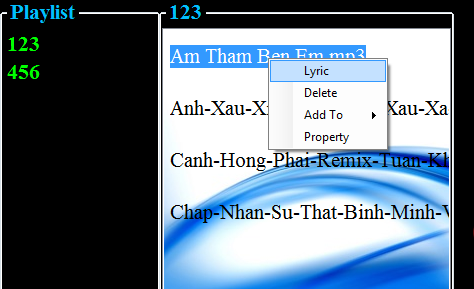
listView.Items.Remove(listView.FocusedItem);

* Xóa file text:

// Xóa file .txt của playlist

File.Delete(@"Playlist\" + listView.FocusedItem.Text + ".txt");

1. **Menu chuột phải My Music:** tương tự như **panel Listen.**



1.  **Panel Search**
2. **Back, Nowplaying, Home** tương tự như **panel Listien.**
3. **Location:** chọn đường dẫn thư mục chứa file nhạc.



pbLocation.Click += (sender, args) =>

{

bool avaiable = false;

FolderBrowserDialog fbd = new FolderBrowserDialog();

try // try-catch => Tránh lỗi thoát Dialog mà không chọn path

{

if (fbd.ShowDialog() == DialogResult.OK)

{

foreach (string item in dirLocation) // dirLocation chứa tất cả thư mục media

{

if (item == fbd.SelectedPath) // Nếu thư mục đã tồn tại => Cảnh báo

{

MessageBox.Show("The path has already existed", string.Empty, MessageBoxButtons.OK, MessageBoxIcon.Error);

avaiable = true;

break;

}

}

// Nếu thư mục đang chọn chưa tồn tại

if (!avaiable)

{

// Kiểm tra thư mục đang chọn có chứa media không

foreach (string item in Directory.GetFiles(fbd.SelectedPath))

{

if (item.EndsWith(".mp3") || item.EndsWith(".wav") || item.EndsWith(".flac") || item.EndsWith(".mpg") || item.EndsWith(".mp4") || item.EndsWith(".mkv") || item.EndsWith(".vob"))

{

avaiable = false;

break;

}

else

avaiable = true;

}

// Nếu đường dẫn đã chọn có chứa media => Lưu mới

if (!avaiable)

{

lvLocation.Items.Add(fbd.SelectedPath);

File.AppendAllText(@"Location.txt", fbd.SelectedPath + Environment.NewLine);

dirLocation.Add(fbd.SelectedPath);

}

else // Thông báo thư mục không có media

MessageBox.Show("The path doesn't contain media files", string.Empty, MessageBoxButtons.OK, MessageBoxIcon.Error);

}

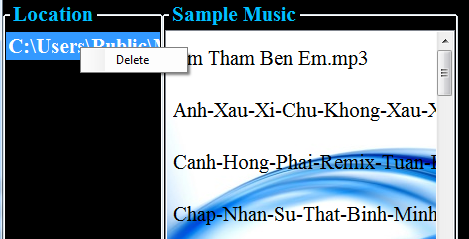
}

catch

{

}

};

1. **Menu chuột phải Location:**

* Delete: xóa Location
* Xóa trên listview:

listView.Items.Remove(listView.FocusedItem);

* Xóa đường dẫn thư mục khỏi dirLocation:

dirLocation.Remove(listView.FocusedItem.Text);

* Xóa file text:

File.Delete(@"Location.txt"); // Xóa file Location.txt

* Tạo mới lại file text:

string line = string.Empty;

// Tạo mới file Location.txt

FileStream stream = new FileStream(@"Location.txt", FileMode.CreateNew); StreamWriter writer = new StreamWriter(stream, Encoding.UTF8);

foreach (string item in dirLocation)

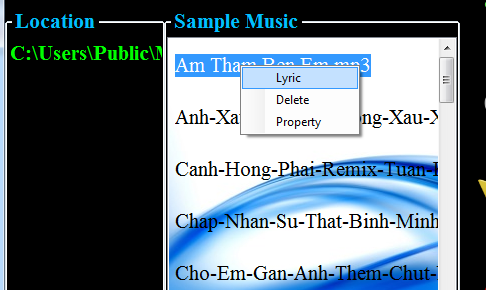
{

// Ghi các đường dẫn đã update ở trên vào file Location.txt mới

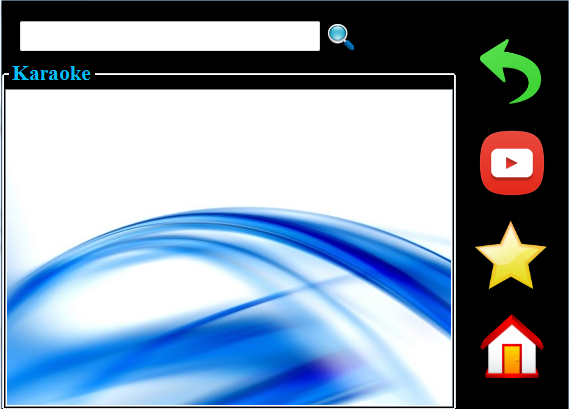
writer.WriteLine(item);

}

1. **Menu chuột phải Sample Music:** tương tự như **panel Listen.**

****

1. **Panel Karaoke**

****

1. **Karaoke:** xuất ra video karaoke thông qua <https://www.youtube.com>.

pbPlayer.Click += (sender, args) =>

{

// Chỉ hiển thị player khi đang phát

if (flash.Movie != null && flash.Movie != "Karaoke")

{

flash.Visible = true;

gbKaraoke.Text = string.Empty;

}

};



if (args.Button == MouseButtons.Left) // Click trái lên lvKaraoke => Play

{

if (gbKaraoke.Text == "Result") // Nếu đang tìm kiếm

{

wmp.Ctlcontrols.stop(); // Stop wmp

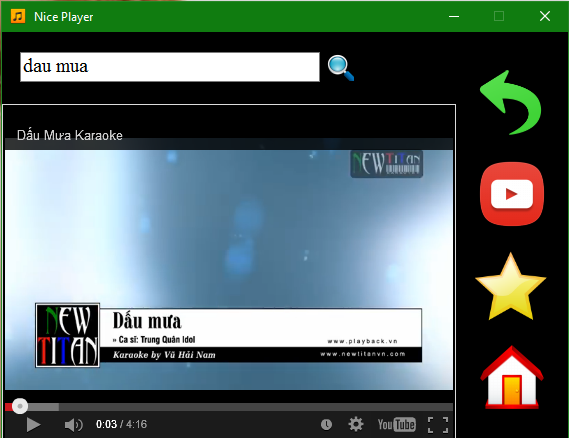
flash.Movie = urlYouTube[lvKaraoke.FocusedItem.Index]; // Phát title được chọn

flash.Visible = true;

gbKaraoke.Text = string.Empty;

}

}



1. **Favorite:** danh sách yêu thích.

pbFavorite.Click += (sender, args) =>

{

lvKaraoke.Clear();

flash.Visible = false;

gbKaraoke.Text = status = "Favorite";

int i = 0;

foreach (string item in dirFavorite)

{

if (i % 2 == 0) // i chẵn là Title + i lẻ là URL

lvKaraoke.Items.Add(item);

i++;

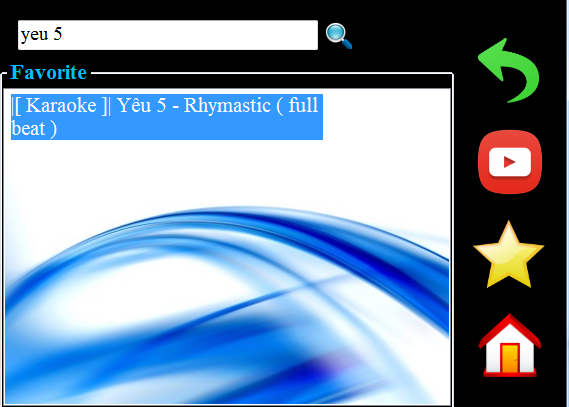
}

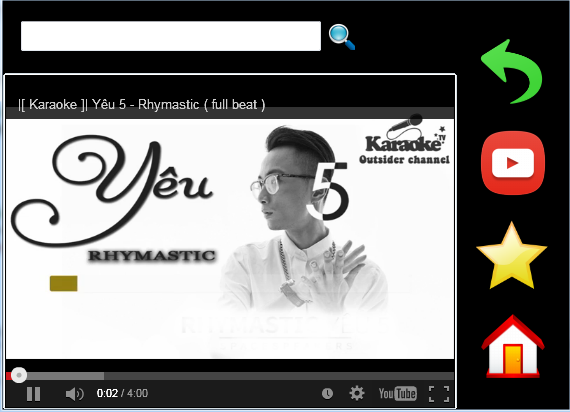
// Nếu chưa có dữ liệu Favorite

if (lvKaraoke.Items.Count == 0)

lvKaraoke.Items.Add("No data found !");

};

****



if (args.Button == MouseButtons.Left) // Click trái lên lvKaraoke => Play

{

if (gbKaraoke.Text == "Result") // Nếu đang tìm kiếm

{

wmp.Ctlcontrols.stop(); // Stop wmp

flash.Movie = urlYouTube[lvKaraoke.FocusedItem.Index]; // Phát title được chọn

flash.Visible = true;

gbKaraoke.Text = string.Empty;

}

else // Nếu đang xem Favorite

{

foreach (string item in dirFavorite)

{

if (item == lvKaraoke.FocusedItem.Text)

{

wmp.Ctlcontrols.stop(); // Stop wmp

flash.Movie = dirFavorite[dirFavorite.IndexOf(item) + 1]; // Phát title được chọn

flash.Visible = true;

gbKaraoke.Text = string.Empty;

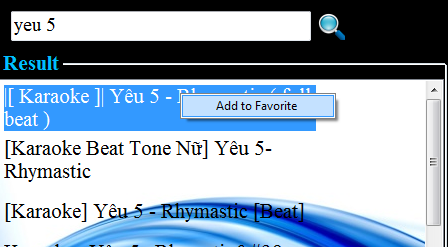
break;

}

}

}

}

1. **Menu chuột phải Karaoke:**

* Add to Favorite: thêm bài hát karaoke vào danh sách yêu thích
* Kiểm tra xem bài hát đã tồn tại trong danh sách chưa:

bool available = false; // Title cần thêm chưa tồn tại trong Favorite

// Kiểm tra tồn tại

foreach (string item in dirFavorite)

{

if (item == urlYouTube[lvKaraoke.FocusedItem.Index])

{

available = true;

break;

}

}

* Thêm bài hát vào danh sách:

if (!available) // Nếu chưa tồn tại thì thêm vào danh sách

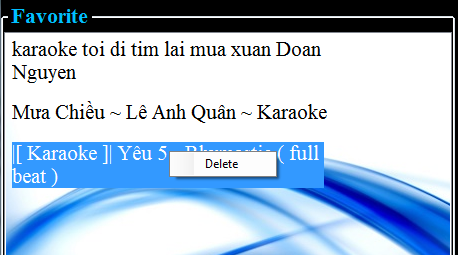
{

File.AppendAllText(@"Karaoke.txt", lvKaraoke.FocusedItem.Text + Environment.NewLine + urlYouTube[lvKaraoke.FocusedItem.Index] + Environment.NewLine);

dirFavorite.Add(lvKaraoke.FocusedItem.Text);

dirFavorite.Add(urlYouTube[lvKaraoke.FocusedItem.Index]);

}

1. **Menu chuột phải Favorite:**

* Delete: xóa bài hát đã chọn trong danh sách yêu thích
* Xóa trên listview:

lvKaraoke.Items.Remove(lvKaraoke.FocusedItem);

* Xóa khỏi danh sách yêu thích:

foreach (string item in dirFavorite)

{

if (item == lvKaraoke.FocusedItem.Text)

{

dirFavorite.RemoveRange(dirFavorite.IndexOf(item), 2);

break;

}

}

* Xóa khỏi file text:

File.Delete(@"Karaoke.txt"); // Xóa file Karaoke.txt

* Tạo mới file text:

FileStream stream = new FileStream(@"Karaoke.txt", FileMode.CreateNew);

StreamWriter writer = new StreamWriter(stream, Encoding.UTF8);

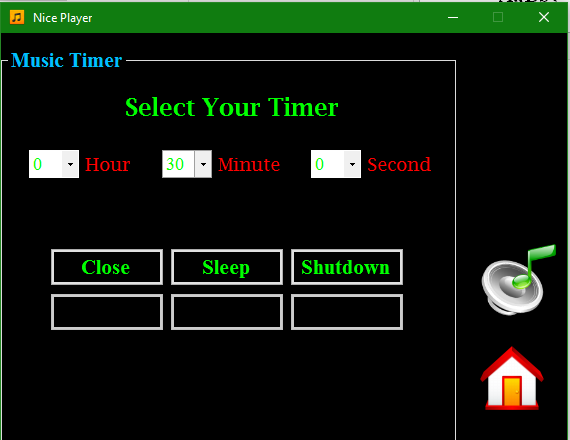
foreach (string item in dirFavorite)

{

writer.WriteLine(item); // Lưu các đường dẫn từ dirFavorite đã update

}

1. **Panel Power**



1. C:\Users\Trung Quan\AppData\Local\Microsoft\Windows\INetCache\Content.Word\close.png**Close:** hẹn thời gian tắt chương trình.

btnClose.Click += (sender, args) =>

{

time = 3600 \* int.Parse(cbHour.SelectedItem.ToString()) + 60 \* int.Parse(cbMinute.SelectedItem.ToString()) + int.Parse(cbSecond.SelectedItem.ToString());

timer.Start();

type = "Close";

};

1. C:\Users\Trung Quan\AppData\Local\Microsoft\Windows\INetCache\Content.Word\sleep.png**Sleep:** hẹn thời gian cho máy ngủ.

btnSleep.Click += (sender, args) =>

{

time = 3600 \* int.Parse(cbHour.SelectedItem.ToString()) + 60 \* int.Parse(cbMinute.SelectedItem.ToString()) + int.Parse(cbSecond.SelectedItem.ToString());

timer.Start();

type = "Sleep";

};

1. C:\Users\Trung Quan\AppData\Local\Microsoft\Windows\INetCache\Content.Word\shutdown.png**Shuwdown:** hẹn thời gian tắt máy tính.

btnShutdown.Click += (sender, args) =>

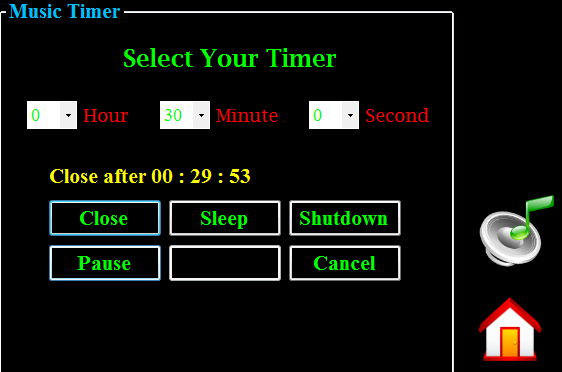
{

time = 3600 \* int.Parse(cbHour.SelectedItem.ToString()) + 60 \* int.Parse(cbMinute.SelectedItem.ToString()) + int.Parse(cbSecond.SelectedItem.ToString());

timer.Start();

type = "Shutdown";

};



1. C:\Users\Trung Quan\AppData\Local\Microsoft\Windows\INetCache\Content.Word\pause.png**Pause:** tạm dừng lệnh đang thực thi.

btnPause.Click += (sender, args) =>

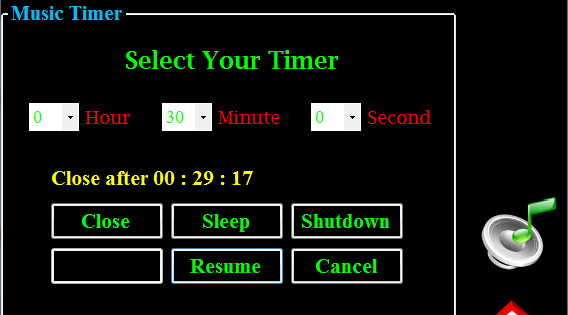
{

timer.Stop();

btnResume.Enabled = true;

btnPause.Enabled = false;

};

****

1. C:\Users\Trung Quan\AppData\Local\Microsoft\Windows\INetCache\Content.Word\resume.png**Resume:** tiếp tục thục thi.

btnResume.Click += (sender, args) =>

{

timer.Start();

btnPause.Enabled = true;

btnResume.Enabled = false;

};

1. C:\Users\Trung Quan\AppData\Local\Microsoft\Windows\INetCache\Content.Word\cancel.png**Cancel:** hủy lệnh đang thực thi.

btnCancel.Click += (sender, args) =>

{

lblRemain.Text = "Canceled !";

btnPause.Enabled = false;

btnResume.Enabled = false;

btnCancel.Enabled = false;

timer.Stop();

};

1. **Nowplaying, Home** tương tự như **panel Listien.**
2. **Tài liệu tham khảo**

**https://hoangmirs.wordpress.com/2013/06/30/su-dung-thu-vien-htmlagilitypack-de-xu-ly-van-ban-html/**