

TRƯỜNG ĐẠI HỌC SÀI GÒN
KHOA CÔNG NGHỆ THÔNG TIN



PHÁT TRIỂN PHẦN MỀM MÃ NGUỒN MỞ

Xây dựng phần mềm

App CHILL N FREE trên Python

GVHD: Từ Lăng Phiêu
SV: Lý Văn Công - 3121410094
Lê Ngọc Bích Thảo - 3121410459
Nguyễn Quốc Thạnh - 3121410458
Email: lengocbichthao1809@gmail.com

TP. HỒ CHÍ MINH, THÁNG 5/2024

Mục lục

1 PHẦN GIỚI THIỆU	2
1.1 Lý do chọn đề tài	2
1.2 Giới thiệu về ứng dụng	2
1.3 Các thư viện	3
1.3.1 pygame	3
1.3.2 PyQt6	3
1.4 Phân công	4
1.5 Cơ sở dữ liệu	4
2 CÁC CHỨC NĂNG	5
2.1 Admin	5
2.1.1 Music List	5
2.1.1.a Hiển thị danh sách bài hát	5
2.1.1.b Hiển thị chi tiết bài hát	6
2.1.1.c Thêm bài hát mới	7
2.1.1.d Sửa bài hát	10
2.1.1.e Xóa bài hát	12
2.1.1.f Tìm kiếm	14
2.1.1.g Làm mới	15
2.1.1.h Phát thử nhạc	17
2.1.2 Cấp quyền	18
2.1.2.a Khởi động sever	18
2.1.2.b Tắt sever	21
2.2 User	22
2.2.1 Normal Playlist	22
2.2.1.a Tìm kiếm	23
2.2.1.b Chọn bài hát	28
2.2.1.c Phát/Dừng bài hát	31
2.2.1.d Phát bài hát trước đó	33
2.2.1.e Phát bài hát sau đó	34
2.2.1.f Tăng / giảm âm lượng	35
3 KẾT LUẬN	37
3.1 Ưu điểm	37
3.2 Nhược điểm	37



1 PHẦN GIỚI THIỆU

1.1 Lý do chọn đề tài

Phát Triển Kỹ Năng Lập Trình: Xây dựng ứng dụng nghe nhạc sử dụng Python giúp nâng cao kỹ năng lập trình, đặc biệt là với các thư viện như pygame và PyQt6.

Xu Hướng Công Nghệ: Streaming là xu hướng công nghệ phổ biến hiện nay, việc nghiên cứu và phát triển ứng dụng streaming giúp bắt kịp và hiểu rõ hơn về công nghệ này.

Tính Ứng Dụng Cao: Ứng dụng nghe nhạc trực tuyến có tính ứng dụng cao, phục vụ nhu cầu giải trí hàng ngày của người dùng. Khả Năng Tùy Biến: Sử dụng Python cho phép dễ dàng tùy biến và mở rộng tính năng, tạo ra một ứng dụng độc đáo và phù hợp với sở thích cá nhân.

Học Hỏi Về Streaming Media: Dự án cung cấp cơ hội tìm hiểu sâu về các giao thức và kỹ thuật streaming, như HLS, RTMP, và WebRTC.

Trải Nghiệm Da Nền Tảng: Sử dụng Python giúp dễ dàng triển khai ứng dụng trên nhiều nền tảng khác nhau như Windows, macOS, và Linux.

Thị Trường Tiềm Năng: Thị trường âm nhạc trực tuyến đang phát triển mạnh mẽ, mở ra nhiều cơ hội kinh doanh và phát triển sản phẩm.

Thúc Đẩy Sáng Tạo: Làm ứng dụng nghe nhạc thúc đẩy sự sáng tạo trong thiết kế giao diện người dùng và trải nghiệm người dùng (UX/UI).

Đóng Góp Cộng Đồng: Xây dựng ứng dụng mã nguồn mở có thể đóng góp cho cộng đồng, giúp đỡ các lập trình viên khác học hỏi và phát triển.

1.2 Giới thiệu về ứng dụng

Ứng dụng nghe nhạc trực tuyến bằng Python là một giải pháp hiện đại và tiện lợi, đáp ứng nhu cầu nghe nhạc mọi lúc, mọi nơi của người dùng. Được phát triển bằng ngôn ngữ Python, ứng dụng này kết hợp các công nghệ tiên tiến để mang đến trải nghiệm nghe nhạc chất lượng cao và mượt mà.

Tính Năng Chính:

+ Phát Nhạc Trực Tuyến (Streaming): Ứng dụng cho phép người dùng nghe nhạc trực tuyến từ các nguồn phát khác nhau, đảm bảo chất lượng âm thanh cao và không bị gián đoạn.

+ Quản Lý Thư Viện Nhạc Cá Nhân: Người dùng có thể tạo, quản lý danh sách phát (playlist), và tổ chức các bài hát yêu thích theo các danh mục riêng.

+ Tìm Kiếm Nhanh Chóng: Tích hợp chức năng tìm kiếm mạnh mẽ, giúp người dùng dễ dàng tìm kiếm bài hát, nghệ sĩ, hoặc album yêu thích.

+ Giao Diện Thân Thiện và Tùy Biến: Giao diện người dùng được thiết kế thân thiện, dễ sử dụng và có khả năng tùy biến cao, phù hợp với nhiều phong cách và sở thích cá nhân.

Công Nghệ Sử Dụng:

+ Python: Ngôn ngữ lập trình chính, được sử dụng cho cả phần backend và một phần frontend của ứng dụng.

+ Pygame: Thư viện hỗ trợ phát nhạc và xử lý âm thanh.

+ PyQt6: Thư viện tạo giao diện người dùng, giúp tạo ra giao diện trực quan và dễ sử dụng.

+ SQLite/MySQL: Hệ quản trị cơ sở dữ liệu, dùng để lưu trữ thông tin người dùng, danh sách phát, và các dữ liệu liên quan khác.

Ứng dụng nghe nhạc trực tuyến bằng Python không chỉ là một công cụ giải trí mà còn là một nền tảng học tập và phát triển kỹ năng lập trình, sáng tạo của các lập trình viên.



1.3 Các thư viện

1.3.1 pygame

Thư viện Pygame là một bộ công cụ phổ biến trong Python, được sử dụng để phát triển các trò chơi điện tử và các ứng dụng đa phương tiện. Dưới đây là một số tính năng và ứng dụng chính của Pygame:

Đồ họa 2D: Pygame cung cấp các công cụ để tạo và xử lý đồ họa 2D. Người dùng có thể vẽ các hình cơ bản như hình chữ nhật, hình tròn, và các hình dạng tùy chỉnh, cũng như tải và hiển thị hình ảnh.

Âm thanh: Thư viện này hỗ trợ việc phát âm thanh và nhạc nền, cho phép người dùng tạo hiệu ứng âm thanh và nhạc nền cho trò chơi hoặc ứng dụng của mình.

Xử lý sự kiện: Pygame có hệ thống xử lý sự kiện mạnh mẽ, giúp dễ dàng theo dõi và phản hồi các sự kiện người dùng như nhấn phím, di chuyển chuột, và tương tác với các thiết bị điều khiển khác.

Chuyển động và hoạt họa: Pygame cung cấp các công cụ để di chuyển các đối tượng trên màn hình và tạo các hoạt họa mượt mà, giúp cho việc phát triển các trò chơi trở nên đơn giản và trực quan hơn.

Quản lý thời gian: Thư viện này cho phép quản lý thời gian và tốc độ khung hình, điều này rất quan trọng trong việc đảm bảo trò chơi chạy mượt mà và đồng nhất trên các hệ thống khác nhau.

Thư viện phụ trợ: Pygame cũng đi kèm với nhiều thư viện phụ trợ giúp đơn giản hóa các tác vụ phức tạp như xử lý hình ảnh, tạo các hiệu ứng đặc biệt, và nhiều hơn nữa.

1.3.2 PyQt6

Thư viện PyQt6 là một bộ công cụ mạnh mẽ và linh hoạt cho phát triển ứng dụng giao diện đồ họa (GUI) trong Python. PyQt6 là phiên bản mới nhất của PyQt, dựa trên bộ công cụ Qt6 của Qt Company. Dưới đây là các tính năng và ứng dụng chính của PyQt6:

Tính năng chính của PyQt6 Widgets và Controls phong phú: PyQt6 cung cấp một bộ sưu tập rộng lớn các widgets và controls (như nút, hộp thoại, thanh trượt, và bảng) để xây dựng các giao diện người dùng phức tạp.

Tích hợp mạnh mẽ với Python: PyQt6 tận dụng sức mạnh của Python, cho phép lập trình viên sử dụng các tính năng của ngôn ngữ này như quản lý bộ nhớ tự động, cú pháp đơn giản và khả năng tích hợp với các thư viện Python khác.

Hỗ trợ đa nền tảng: Các ứng dụng phát triển với PyQt6 có thể chạy trên nhiều hệ điều hành khác nhau như Windows, macOS, và Linux mà không cần phải chỉnh sửa mã nguồn.

Hỗ trợ quốc tế hóa (i18n): PyQt6 cung cấp các công cụ để dịch và địa phương hóa ứng dụng, giúp dễ dàng phát triển các phần mềm có khả năng phục vụ người dùng trên toàn thế giới.

Tính năng đồ họa và đa phương tiện: Thư viện hỗ trợ việc tạo các ứng dụng có đồ họa và đa phương tiện phức tạp, bao gồm cả đồ họa 2D và 3D, video, và âm thanh.

Mô hình sự kiện mạnh mẽ: PyQt6 sử dụng một hệ thống sự kiện mạnh mẽ và linh hoạt, cho phép lập trình viên xử lý các sự kiện người dùng một cách hiệu quả.



1.4 Phân công

1.5 Cơ sở dữ liệu

			id	song_name	singer_name	genre	mp3	img
<input type="checkbox"/>		Bấm vào mũi tên chỉ xuống để bật/tắt tính khả kiến của cột.	70	Nhắn Nhủ	Ronboogz	Việt	Nhắn Nhủ(Ronboogz).mp3	NhanNhu.jpg
<input type="checkbox"/>		Sửa	Chép	Xóa bỏ	Moonlight	Dhruv	Âu Mỹ	Moolight(Dhruv).mp3
<input type="checkbox"/>		Sửa	Chép	Xóa bỏ	Round And Round	Heize	Hàn	Round and round(Heize).mp3
<input type="checkbox"/>		Sửa	Chép	Xóa bỏ	Stay With Me	Chanyeol,Punch	Hàn	Stay with me(Chanyeol,Punch).mp3
<input type="checkbox"/>		Sửa	Chép	Xóa bỏ	Double Take	Dhruv	Âu Mỹ	double take(Dhruv).mp3

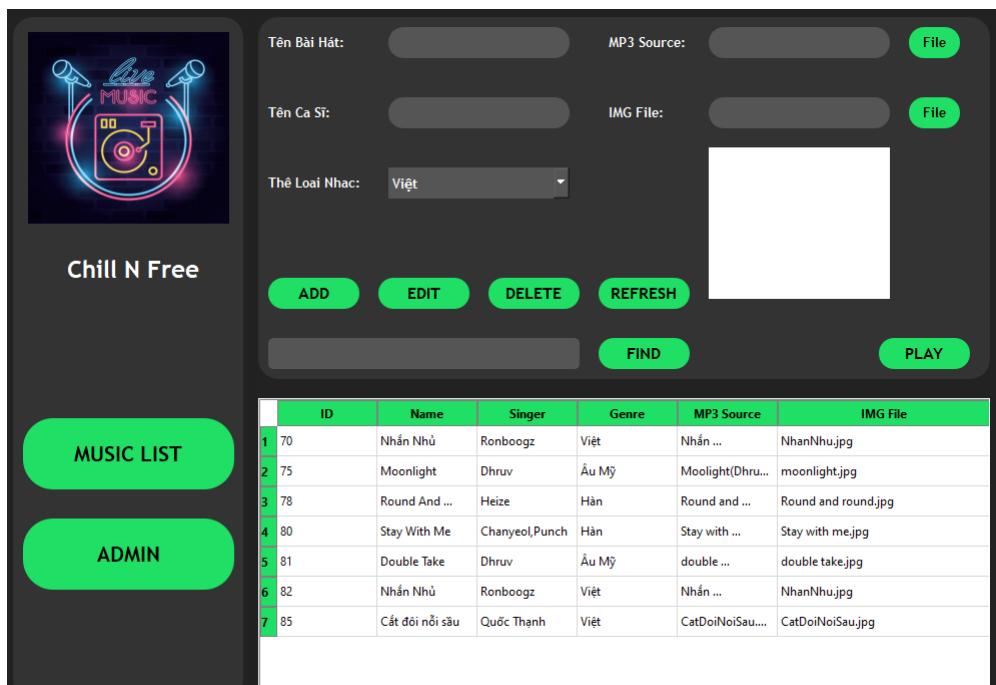
2 CÁC CHỨC NĂNG

2.1 Admin

2.1.1 Music List

2.1.1.a Hiển thị danh sách bài hát

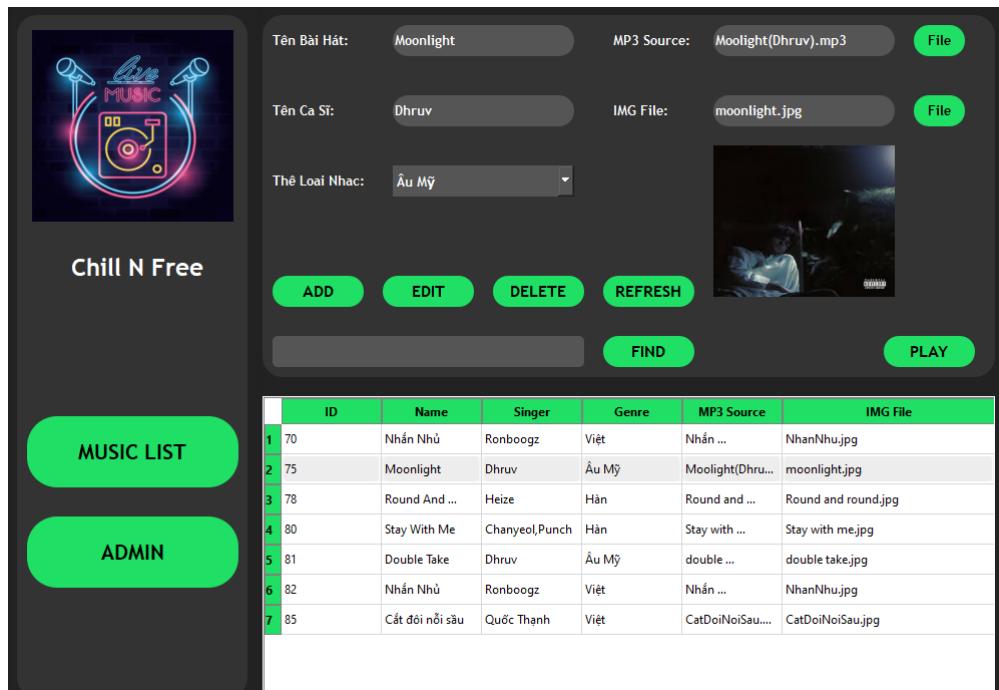
Nhấn vào dòng có tên bài hát để chọn bài hát



Hình 1: Hiển thị danh sách bài hát



2.1.1.b Hiển thị chi tiết bài hát



Hình ảnh chi tiết của bài hát

CODE:

```
def on_table_selection_changed(self, selected):
    indexes = selected.indexes()
    if indexes:
        # Lấy chỉ số hàng được chọn
        selected_row = indexes[0].row()
        # Lấy dữ liệu từ model tại hàng được chọn
        model = self.tb_music.model()
        id = model.index(selected_row, column=0).data()
        name = model.index(selected_row, column=1).data()
        singer = model.index(selected_row, column=2).data()
        genre = model.index(selected_row, column=3).data()
        mp3 = model.index(selected_row, column=4).data()
        img = model.index(selected_row, column=5).data()

        # Đường dẫn đến thư mục chứa ảnh
        img_directory = r"C:\MNM\MusicPlayer (3)\MusicPlayer\Admin\img"

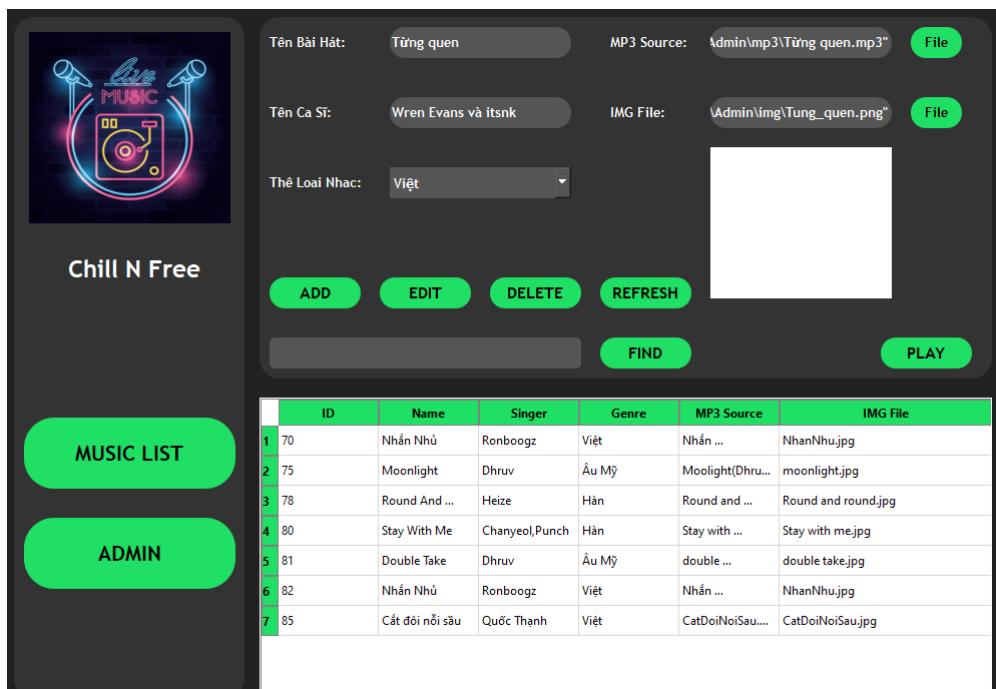
        # Cập nhật đường dẫn đầy đủ của tệp ảnh
        img_path = os.path.join(img_directory, img)

        # Đổ dữ liệu lên các QLineEdit và QComboBox tương ứng
        self.tf_name.setText(name)
        self.tf_singer.setText(singer)
        self.cb_genre.setCurrentText(genre)
        self.tf_mp3.setText(mp3)
        self.tf_img.setText(img)
        pixmap = QPixmap(img_path)
        self.lb_imgfile.setPixmap(pixmap)
        self.lb_imgfile.setScaledContents(True)
```



2.1.1.c Thêm bài hát mới

Để thêm bài hát nhập từng thông tin bài hát (Tên bài hát, Tên ca sĩ, Thể loại nhạc, MP3 source, IMG File), sau đó chọn ADD



Hình ảnh khi nhập thông tin bài hát



Chill N Free

Tên Bài Hát: _____ MP3 Source: _____ File

Tên Ca Sĩ: _____ IMG File: _____ File

Thể Loại Nhạc: Việt

MUSIC LIST

ADMIN

ID	Name	Singer	Genre	MP3 Source	IMG File
1	70	Nhắn Nhủ	Ronboogz	Việt	Nhan ... NhanNhu.jpg
2	75	Moonlight	Dhruv	Âu Mỹ	Moolight(Dhru... moonlight.jpg
3	78	Round And ...	Heize	Hàn	Round and ... Round and round.jpg
4	80	Stay With Me	Chanyeol,Punch	Hàn	Stay with ... Stay with me.jpg
5	81	Double Take	Dhruv	Âu Mỹ	double ... double take.jpg
6	82	Nhắn Nhủ	Ronboogz	Việt	Nhan ... NhanNhu.jpg
7	85	Cắt đồi nỗi sâu	Quốc Thanh	Việt	CatDoiNoiSau... CatDoiNoiSau.jpg
8	89	Tùng quen	Wren Evans và ...	Việt	Tùng quen.mp3 Tung_quen.png

Sau khi thêm bài hát



CODE :

```
def add_music(self):
    # Kiểm tra các trường nhập liệu trống
    if not self.validate_textfields():
        return

    # Nếu không có trường nào trống, tiến hành thêm bản ghi
    name = self.tf_name.text().strip()
    singer = self.tf_singer.text().strip()
    genre = self.cb_genre.currentText().strip()
    mp3_path = self.tf_mp3.text().strip()
    img_path = self.tf_img.text().strip()

    # Thực hiện thêm bản ghi qua phương thức add_music của MusicBus
    mp3_file_name = os.path.basename(mp3_path)
    img_file_name = os.path.basename(img_path)
    self.music_bus.add_music(name, singer, genre, mp3_file_name, img_file_name, sel
```



2.1.1.d Sửa bài hát

Để sửa bài hát, nhập thông tin cần sửa sau đó chọn EDIT

The screenshot shows a music application interface. At the top, there are fields for 'Tên Bài Hát' (Song Name) containing 'Cắt đồi nỗi sâu', 'MP3 Source' (MP3 Source) containing 'CatDoiNoiSau.mp3', and a 'File' button. Below that, 'Tên Ca Sĩ' (Artist Name) is 'Tăng Duy Tân', 'IMG File' (Image File) is 'CatDoiNoiSau.jpg', and another 'File' button is present. A dropdown for 'Thể Loại Nhạc' (Genre) is set to 'Việt'. On the right, a thumbnail image of the song cover is shown with the title 'CẮT ĐỒI NỖI SÂU' and the subtitle 'BÀI HẠT TỐT NHẤT TRÊN KHO MUSIC CỦA YOUTUBE'. Below these controls are buttons for 'ADD', 'EDIT', 'DELETE', 'REFRESH', 'FIND', and 'Play'. A large green button labeled 'MUSIC LIST' is on the left, and a smaller green button labeled 'ADMIN' is below it. At the bottom is a table showing a list of songs:

ID	Name	Singer	Genre	MP3 Source	IMG File
1	70	Nhán Nhủ	Ronboogz	Việt	NhanNhu.jpg
2	75	Moonlight	Dhruv	Âu Mỹ	Moolight(Dhru... moonlight.jpg
3	78	Round And ...	Heize	Hàn	Round and ... Round and round.jpg
4	80	Stay With Me	Chanyeol,Punch	Hàn	Stay with ... Stay with me.jpg
5	81	Double Take	Dhruv	Âu Mỹ	double ... double take.jpg
6	82	Nhán Nhủ	Ronboogz	Việt	NhanNhu.jpg
7	85	Cắt đồi nỗi sâu	Quốc Thanh	Việt	CatDoiNoiSau.... CatDoiNoiSau.jpg
8	89	Tùng quen	Wren Evans và ...	Việt	Tung quen.mp3 Tung_quen.png

	ID	Name	Singer	Genre	MP3 Source	IMG File
1	70	Nhán Nhủ	Ronboogz	Việt	NhanNhu.jpg	
2	75	Moonlight	Dhruv	Âu Mỹ	Moolight(Dhru... moonlight.jpg	
3	78	Round And ...	Heize	Hàn	Round and ... Round and round.jpg	
4	80	Stay With Me	Chanyeol,Punch	Hàn	Stay with ... Stay with me.jpg	
5	81	Double Take	Dhruv	Âu Mỹ	double ... double take.jpg	
6	82	Nhán Nhủ	Ronboogz	Việt	NhanNhu.jpg	
7	85	Cắt đồi nỗi sâu	Tăng Duy Tân	Việt	CatDoiNoiSau.... CatDoiNoiSau.jpg	
8	89	Tùng quen	Wren Evans và ...	Việt	Tung quen.mp3 Tung_quen.png	

Sau khi sửa bài hát



```
def update_music(self):
    if not self.validate_textfields():
        return

    try:
        # Lấy chỉ số hàng đang được chọn
        selected_indexes = self.tb_music.selectionModel().selectedIndexes()
        if selected_indexes:
            selected_row = selected_indexes[0].row()
            model = self.tb_music.model()

            # Lấy thông tin từ các trường dữ liệu
            music_id = model.index(selected_row, column=0).data()
            name = self.tf_name.text().strip()
            singer = self.tf_singer.text().strip()
            genre = self.cb_genre.currentText().strip()
            mp3_path = self.tf_mp3.text().strip()
            img_path = self.tf_img.text().strip()

            # Cập nhật dữ liệu thông qua MusicBus
            self.music_bus.update_music(music_id, name, singer, genre, mp3_path, im

            # Cập nhật lại bảng và làm mới trang
            self.show_table()
            self.refresh_page()

            print("Cập nhật thành công")
    except Exception as e:
        logging.error(f"Lỗi khi cập nhật dữ liệu: {str(e)}")
```



2.1.1.e Xóa bài hát

Để xóa bài hát, chọn bài hát muốn xóa sau đó nhấn DELETE

The screenshot shows a music management application interface. At the top, there are input fields for 'Tên Bài Hát' (Song Name) set to 'Tùng quen', 'MP3 Source' set to 'Tùng quen.mp3', and a 'File' button. Below that, 'Tên Ca Sĩ' (Artist) is 'Wren Evans và itsnk', 'IMG File' is 'Tung_quen.png', and another 'File' button. The 'Thể Loại Nhạc' (Genre) dropdown is set to 'Việt'. To the right, a thumbnail image of a video player showing a person at a piano with the text 'TÙNG QUEN' is displayed. Below these controls are buttons for 'ADD', 'EDIT', 'DELETE', 'REFRESH', 'FIND', and 'PLAY'. A large green button labeled 'MUSIC LIST' is on the left, and a green button labeled 'ADMIN' is below it. At the bottom is a table titled 'MUSIC LIST' with columns: ID, Name, Singer, Genre, MP3 Source, and IMG File. The table contains 8 rows of data.

ID	Name	Singer	Genre	MP3 Source	IMG File
1	70	Nhắn Nhủ	Ronboogz	Việt	NhanNhu.jpg
2	75	Moonlight	Dhruv	Âu Mỹ	Moolight(Dhru... moonlight.jpg
3	78	Round And ...	Heize	Hàn	Round and ... Round and round.jpg
4	80	Stay With Me	Chanyeol,Punch	Hàn	Stay with ... Stay with me.jpg
5	81	Double Take	Dhruv	Âu Mỹ	double ... double take.jpg
6	82	Nhắn Nhủ	Ronboogz	Việt	NhanNhu.jpg
7	85	Cát đồi nỗi sầu	Quốc Thanh	Việt	CatDoiNoiSau.... CatDoiNoiSau.jpg
8	88	Tùng quen	Wren Evans và ...	Việt	Tùng quen.mp3* Tung_quen.png



```
def delete_music(self):
    try:
        selected_indexes = self.tb_music.selectionModel().selectedIndexes()
        if selected_indexes: # Kiểm tra xem có hàng nào được chọn không
            # Lấy chỉ số hàng đầu tiên được chọn
            selected_row = selected_indexes[0].row()
            model = self.tb_music.model()
            music_id = model.index(selected_row, column=0).data()
            print(music_id)
            if music_id is not None: # Kiểm tra xem music_id có giá trị không
                # Gọi phương thức xóa bản ghi từ MusicBus
                self.music_bus.delete_music_by_id(music_id)
                self.refresh_page()
                print("Xóa thành công")
            else:
                print("Không có ID âm nhạc tương ứng.")
        else:
            print("Không có hàng nào được chọn.")
    except Exception as e:
        logging.error(f"An error occurred: {str(e)}")
        print("lỗi tại đây")
```



The screenshot shows a dark-themed application window titled "Chill N Free". On the left, there are two green rounded rectangular buttons labeled "MUSIC LIST" and "ADMIN". The main area contains several input fields: "Tên Bài Hát:" with a placeholder "Nhạc của bạn", "MP3 Source:", and a "File" button; "Tên Ca Sĩ:" with a placeholder "Ca sĩ của bạn", "IMG File:", and a "File" button; and a dropdown menu "Thể Loại Nhạc:" set to "Việt". Below these are four green buttons: "ADD", "EDIT", "DELETE", and "REFRESH". A large white rectangular area is positioned to the right of the input fields. At the bottom, there are three green buttons: "FIND", "PLAY", and another "PLAY" button. To the right of the "FIND" button is a table with the following data:

ID	Name	Singer	Genre	MP3 Source	IMG File
1	70	Nhắn Nhủ	Ronboogz	Việt	NhanNhu.jpg
2	75	Moonlight	Dhruv	Âu Mỹ	Moolight(Dhru... moonlight.jpg
3	78	Round And ...	Heize	Hàn	Round and ... Round and round.jpg
4	80	Stay With Me	Chanyeol,Punch	Hàn	Stay with ... Stay with me.jpg
5	81	Double Take	Dhruv	Âu Mỹ	double ... double take.jpg
6	82	Nhắn Nhủ	Ronboogz	Việt	NhanNhu.jpg
7	85	Cắt đồi nỗi sầu	Quốc Thanh	Việt	CatDoiNoiSau... CatDoiNoiSau.jpg

Sau khi xóa bài hát

2.1.1.f Tìm kiếm

Nhập tên bài hát, sau đó chọn FIND

The screenshot shows the same application interface as before, but now the "FIND" button is highlighted. In the search bar above the table, the text "Moonlight" is entered. The table below shows one result row:

ID	Song Name	Singer Name	Genre	MP3	IMG
1	75	Moonlight	Dhruv	Âu Mỹ	Moolight(Dhru... moonlight.jpg



```
def search_and_display_music(self):
    # Lấy nội dung từ ô nhập liệu
    search_text = self.tf_find.text()

    # Thực hiện tìm kiếm âm nhạc
    if search_text:
        # Gọi hàm search_music của đối tượng MusicDao để thực hiện tìm kiếm trong cơ sở dữ liệu
        search_result = music_dao.search_music(search_text)

        # Xóa dữ liệu hiện có trên bảng tb_music
        self.tb_music.setModel(None)
        self.tb_music.setModel(QtGui.QStandardItemModel())

        # Hiển thị kết quả tìm kiếm trên bảng
        if search_result:
            model = QtGui.QStandardItemModel(len(search_result), 6)
            model.setHorizontalHeaderLabels(["ID", "Song Name", "Singer Name", "Genre", "Duration", "Lyrics"])
            for row, record in enumerate(search_result):
                for column, value in enumerate(record):
                    item = QtGui.QStandardItem(str(value))
                    model.setItem(row, column, item)
            self.tb_music.setModel(model)

            # Gọi phương thức on_table_selection_changed để hiển thị thông tin chi tiết khi có sự thay đổi
            self.tb_music.selectionModel().selectionChanged.connect(self.on_table_selection_changed)
        else:
            print("Không tìm thấy kết quả phù hợp.")
    else:
        print("Vui lòng nhập từ khóa tìm kiếm.")
```

2.1.1.g Làm mới

Sau khi nhấn refresh thì giao diện làm mới lại ban đầu



ID	Name	Singer	Genre	MP3 Source	IMG File
1	Nhán Nhủ	Ronboogz	Việt	Nhan ...	NhanNhu.jpg
2	Moonlight	Dhruv	Âu Mỹ	Moonlight(Dhru...)	moonlight.jpg
3	Round And ...	Heize	Hàn	Round and ...	Round and round.jpg
4	Stay With Me	Chanyeol,Punch	Hàn	Stay with ...	Stay with me.jpg
5	Double Take	Dhruv	Âu Mỹ	double ...	double take.jpg
6	Nhán Nhủ	Ronboogz	Việt	Nhan ...	NhanNhu.jpg
7	Cát đồi nỗi sâu	Quốc Thành	Việt	CatDoiNoiSau....	CatDoiNoiSau.jpg

```
def refresh_page(self):
    # Gọi phương thức show_table() để làm mới dữ liệu trong bảng
    self.show_table()

    # Làm mới dữ liệu trong các trường nhập liệu
    self.tf_name.clear()
    self.tf_singer.clear()
    self.cb_genre.setCurrentIndex(0)
    self.tf_mp3.clear()
    self.tf_img.clear()
    self.lb_imgfile.clear()

    self.error_name.clear()
    self.error_name.hide()
    self.error_singer.clear()
    self.error_singer.hide()
    self.error_mp3.clear()
    self.error_mp3.hide()
    self.error_img.clear()
    self.error_img.hide()
```



2.1.1.h Phát thử nhạc

Nhấn nút Play để phát thử nhạc

The screenshot shows the 'Chill N Free' application interface. On the left, there are two green rounded rectangular buttons labeled 'MUSIC LIST' and 'ADMIN'. The main area has three input fields: 'Tên Bài Hát' (Từng quen), 'MP3 Source' (Từng quen.mp3), and 'Tên Ca Sĩ' (Wren Evans và itsnk). Below these are dropdown menus for 'Thể Loại Nhạc' (Việt) and 'IMG File' (Tung_quen.png). A preview video thumbnail for 'TÙNG QUEN' is shown on the right. At the bottom are buttons for 'ADD', 'EDIT', 'DELETE', 'REFRESH', 'FIND', and 'PLAY'. To the right of the preview is a table of music tracks:

ID	Name	Singer	Genre	MP3 Source	IMG File
1	70	Nhán Nhú	Ronboogz	Việt	NhanNhu.jpg
2	75	Moonlight	Dhruv	Âu Mỹ	Moolight(Dhru... moonlight.jpg
3	78	Round And ...	Heize	Hàn	Round and ... Round and round.jpg
4	80	Stay With Me	Chanyeol,Punch	Hàn	Stay with ... Stay with me.jpg
5	81	Double Take	Dhruv	Âu Mỹ	double ... double take.jpg
6	82	Nhán Nhú	Ronboogz	Việt	NhanNhu.jpg
7	85	Cát đồi nỗi sầu	Quốc Thanh	Việt	CatDoiNoiSau.... CatDoiNoiSau.jpg
8	88	Tùng quen	Wren Evans và ...	Việt	Tùng quen.mp3* Tung_quen.png

Sau khi nhấn nút Play để nghe thì sau đó nó sẽ chuyển thành nút Pause để dừng nhạc

The screenshot shows the same 'Chill N Free' application interface after a track has been played. The 'PLAY' button has changed to a 'PAUSE' button. The rest of the interface remains the same, including the input fields, dropdowns, preview video, and the music track table.



```
def play_music(self):
    # Lấy dữ liệu từ model
    selected_indexes = self.tb_music.selectionModel().selectedIndexes()
    if selected_indexes:
        # Lấy chỉ số hàng đầu tiên được chọn
        selected_row = selected_indexes[0].row()
        # Lấy dữ liệu từ model tại hàng được chọn
        model = self.tb_music.model()
        mp3_filename = model.index(selected_row, column=4).data()

        # Xây dựng đường dẫn đầy đủ đến tệp âm thanh
        music_folder = r"C:\MNM\MusicPlayer\Admin\mp3"
        mp3_path = os.path.join(music_folder, mp3_filename)

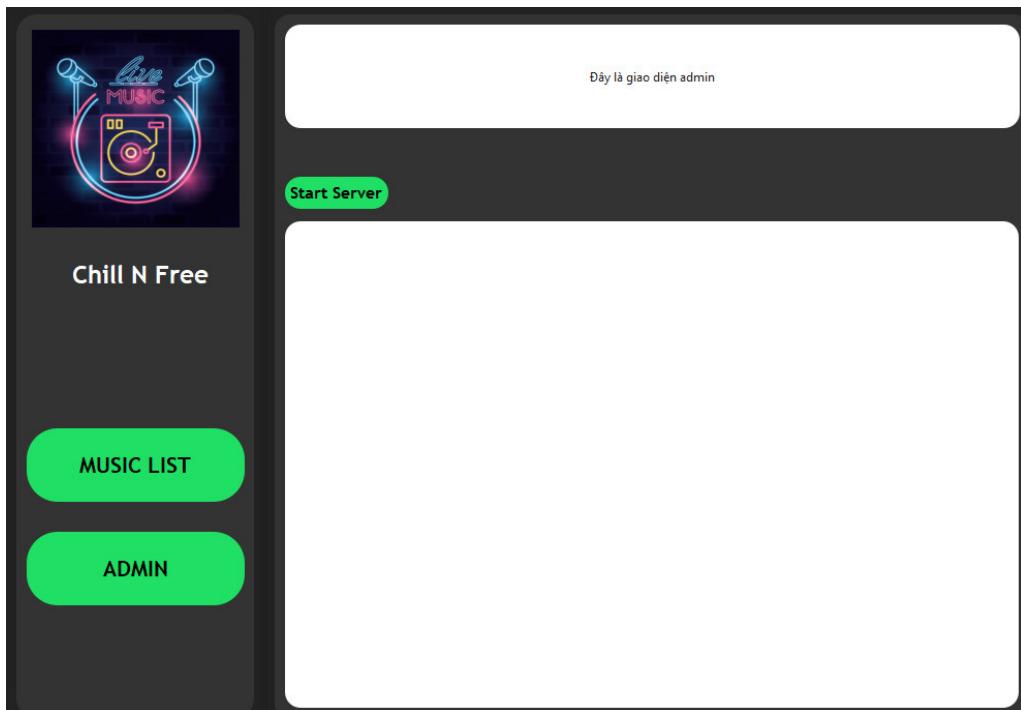
        # Kiểm tra xem thư viện pygame đã được khởi tạo chưa
        if not pygame.mixer.get_init():
            pygame.mixer.init()

        # Kiểm tra xem nhạc đã được phát hay chưa
        if pygame.mixer.music.get_busy():
            # Nếu nhạc đang được phát, tạm dừng
            pygame.mixer.music.pause()
            # Cập nhật văn bản trên nút thành "Play"
            self.btn_play.setText("Play")
        else:
            # Nếu nhạc chưa được phát, tải và phát nhạc
            pygame.mixer.music.load(mp3_path)
            pygame.mixer.music.play()
            # Cập nhật văn bản trên nút thành "Pause"
            self.btn_play.setText("Pause")
```

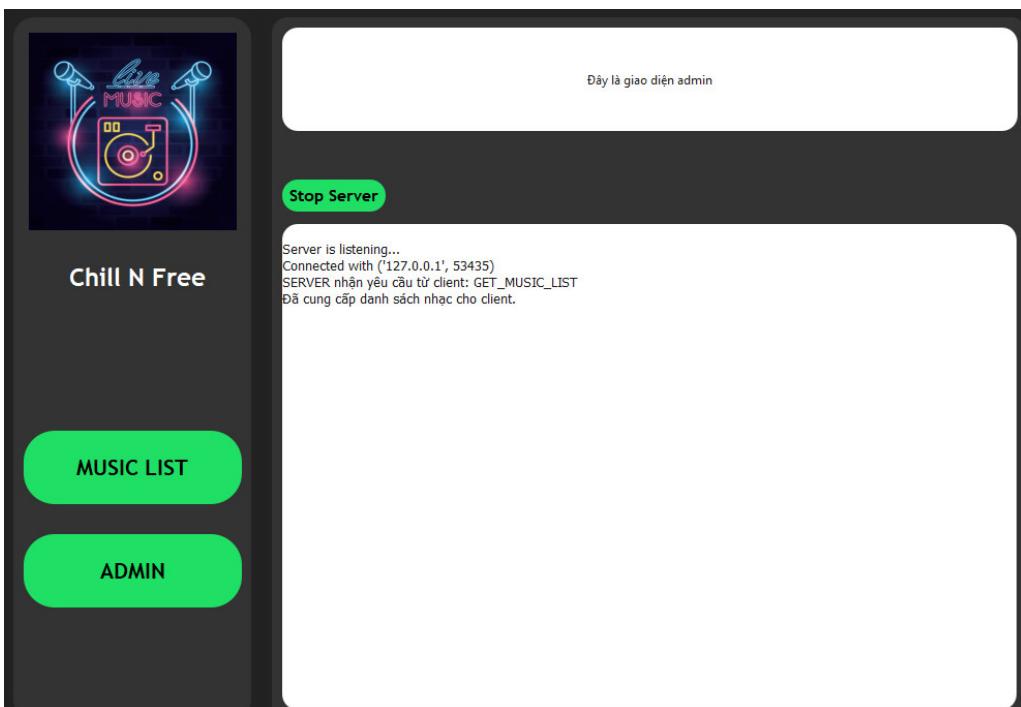
2.1.2 Cấp quyền

2.1.2.a Khởi động sever

Nhấn nút Start sever để khởi động sever



Sau khi nhấn nút Start Sever để khởi động Sever thì sau đó nó sẽ chuyển thành nút Stop sever để dừng sever





CODE

```
def start_server(self):
    if not hasattr(self, 'server_thread') or not self.server_thread.isRunning():
        # Khởi tạo và bắt đầu server nếu chưa được khởi tạo hoặc đang dừng
        self.server_thread = ServerThread(self.music_bus, self.music_dao) # Truyền các tham số cần thiết
        self.server_thread.new_connection.connect(self.update_server_lb)

        self.server_thread.start()
        self.btn_server.setText("Stop Server")

    else:
        # Dừng server nếu đang chạy
        self.server_thread.terminate()
        self.btn_server.setText("Start Server")
```

Khi bấm start sẽ tạo một đối tượng sever mới

```
def run(self):
    host = 'localhost'
    port = 3306

    server = socket.socket(socket.AF_INET, socket.SOCK_STREAM)
    server.bind((host, port))
    server.listen(5) # Số lượng kết nối đồng thời tối đa

    print("Server is listening...")
    self.new_connection.emit("Server is listening...")

    while True:
        client_socket, client_address = server.accept()
        print(f"Connected with {client_address}")

        # Cập nhật giao diện người dùng
        self.new_connection.emit(f"Connected with {client_address}")
        # Đưa tin nhắn vào hàng đợi

        # Gọi hàm update_server_lb ở đây
        # Xử lý client trong một luồng riêng biệt
        thread = threading.Thread(target=self.handle_client, args=(client_socket,))
        thread.start()
```

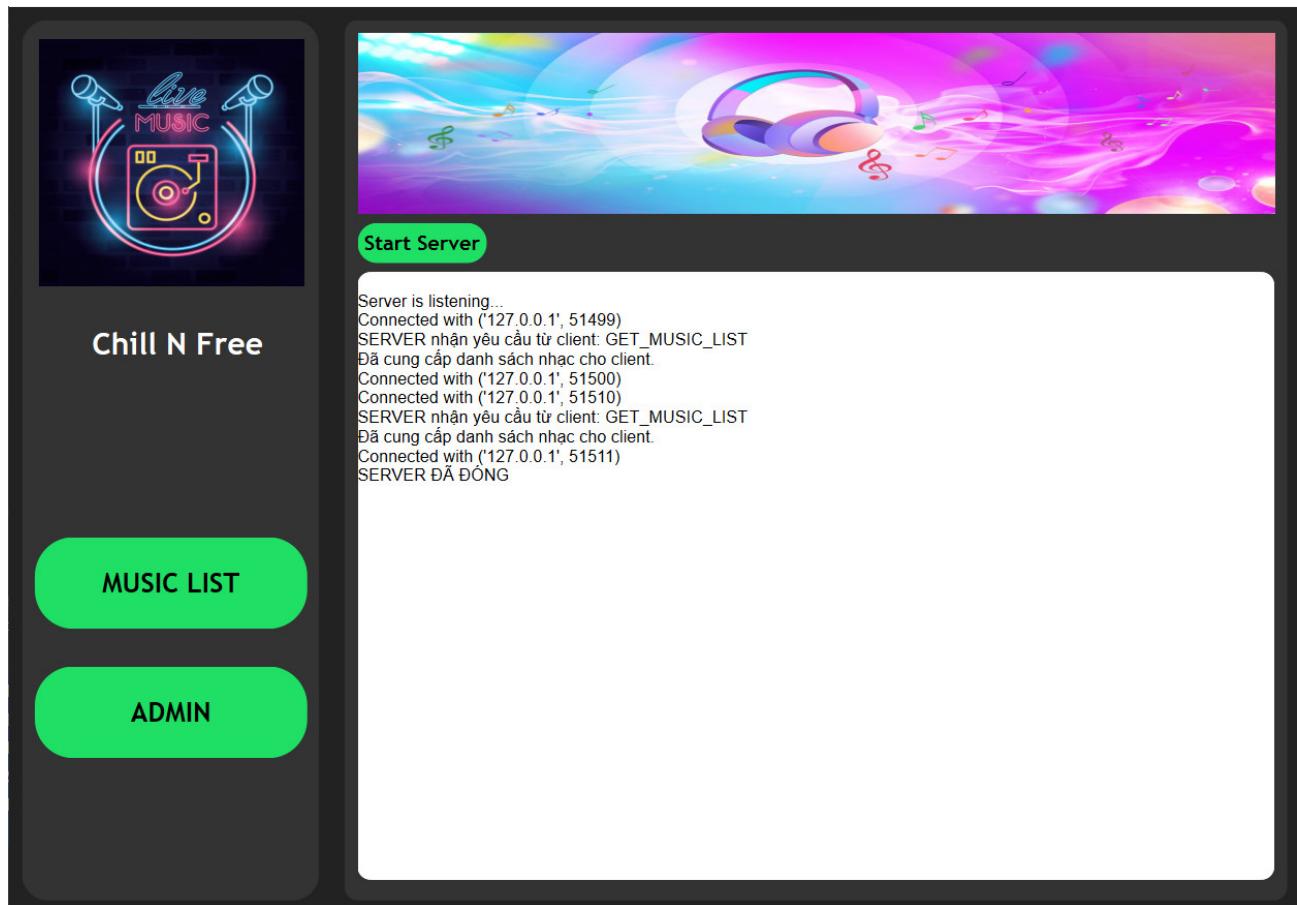
Hàm chạy Sever

```
def update_server_lb(self, message):
    print("Updating server label with message:", message)
    current_text = self.lb_server.text()
    print("Đã nhận thông điệp từ server:", message)
    new_text = f"{current_text}\n{message}"
    self.lb_server.setText(new_text)
```

Hàm khi bấm start sẽ gọi tới

2.1.2.b Tắt sever

Nhấn nút Stop sever để tắt sever



Hình ảnh sau khi tắt Sever

```
def start_server(self):
    if not hasattr(self, 'server_thread') or not self.server_thread.isRunning():
        # Khởi tạo và bắt đầu server nếu chưa được khởi tạo hoặc đang dừng
        self.server_thread = ServerThread(self.music_bus, self.music_dao) # Truyền
        self.server_thread.new_connection.connect(self.update_server_lb)

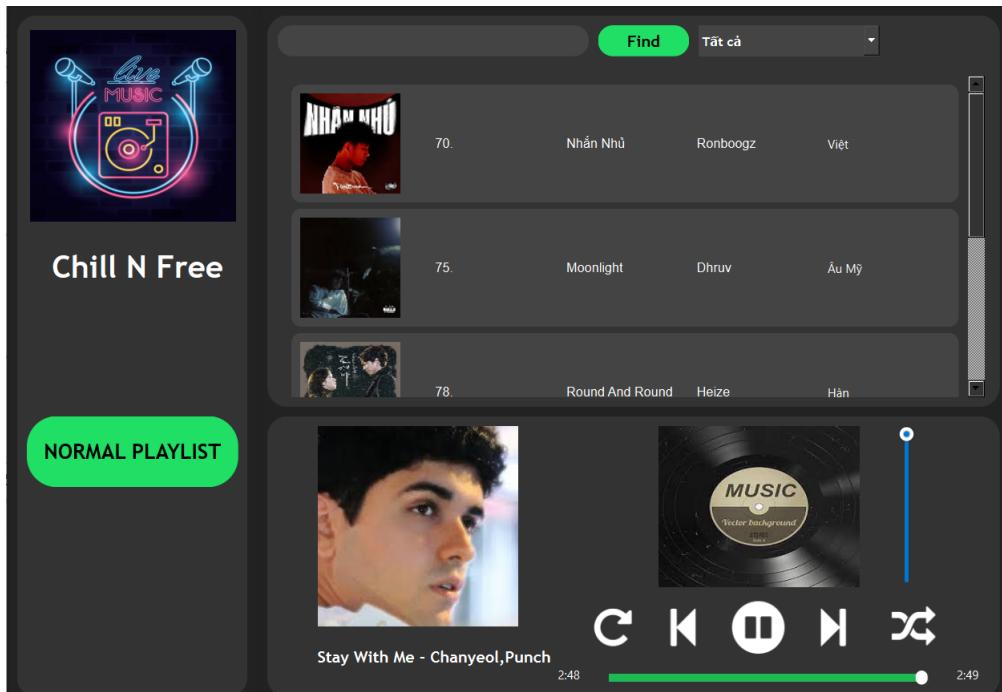
        self.server_thread.start()
        self.btn_server.setText("Stop Server")

    else:
        # Dừng server nếu đang chạy
        self.server_thread.terminate()
        self.btn_server.setText("Start Server")
```

Hàm này sẽ dừng Sever nếu đang chạy

2.2 User

2.2.1 Normal Playlist

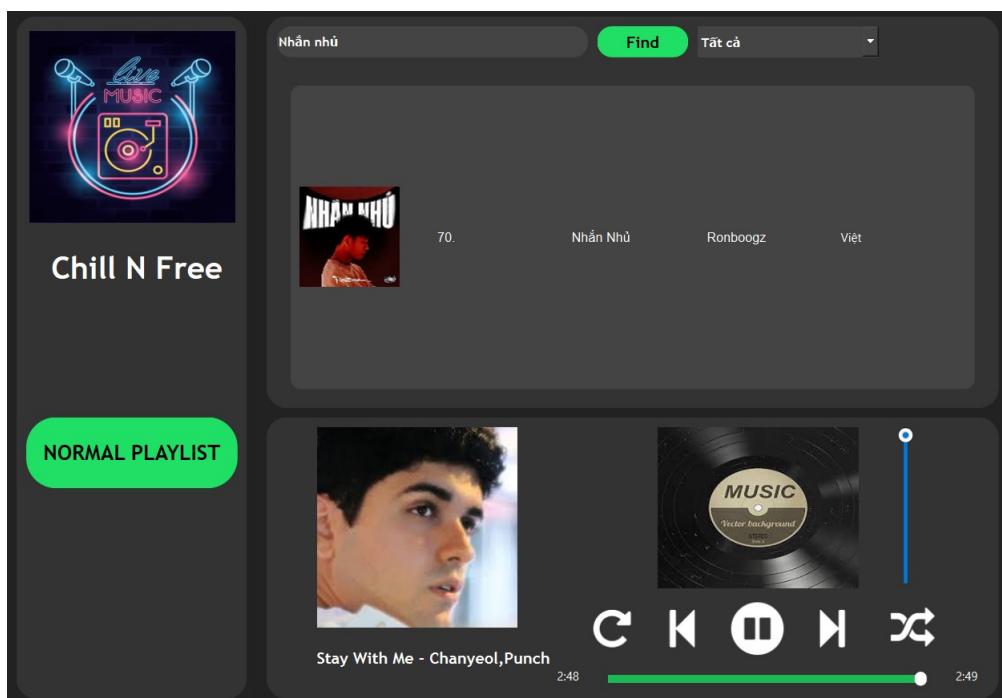


Danh sách bài hát khi nhận dữ liệu từ admin

2.2.1.a Tìm kiếm

TÌM KIẾM CÓ 2 CÁCH :

1. Nhập tên bài hát muốn tìm kiếm



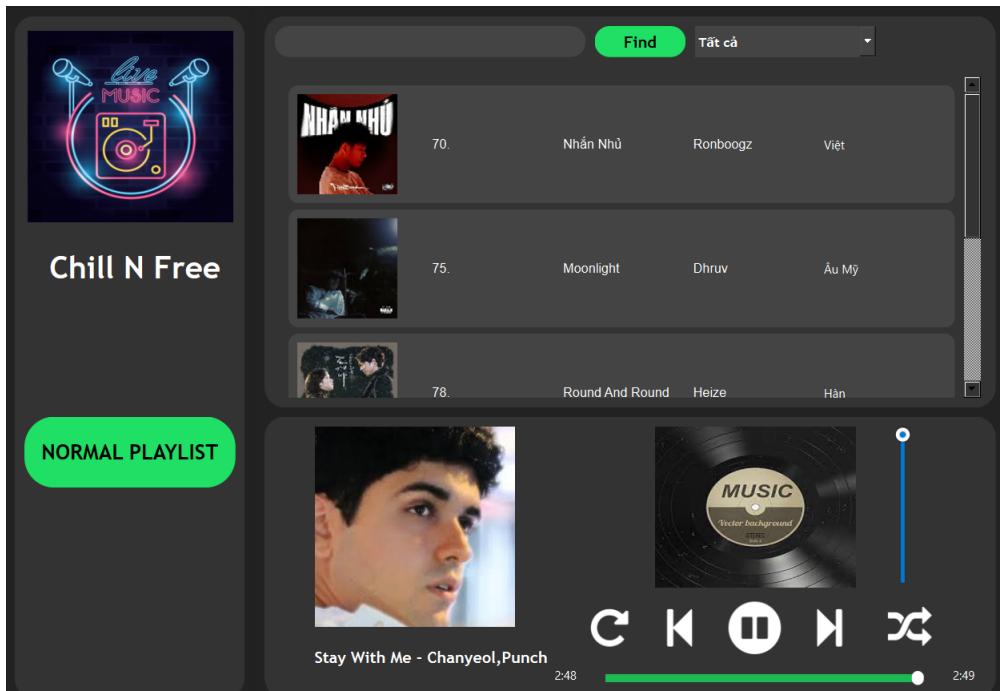
Hình ảnh sau khi tìm kiếm

CODE :

```
1 usage
def findData(self):
    arrFind = []
    for item in self.arrList:
        data = self.tf_find.text()
        if data.lower() in item['name'].lower():
            arrFind.append(item)
    self.update_music_list(arrFind)
```



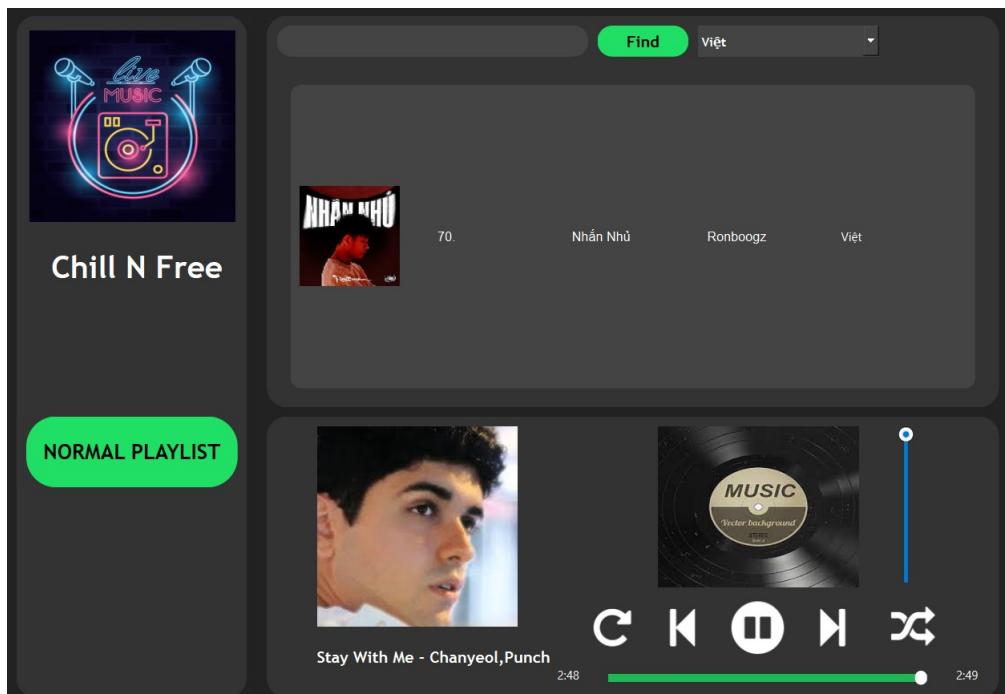
2. Tìm kiếm theo quốc gia
Tìm kiếm tất cả



Hình ảnh sau khi tìm kiếm theo tất cả



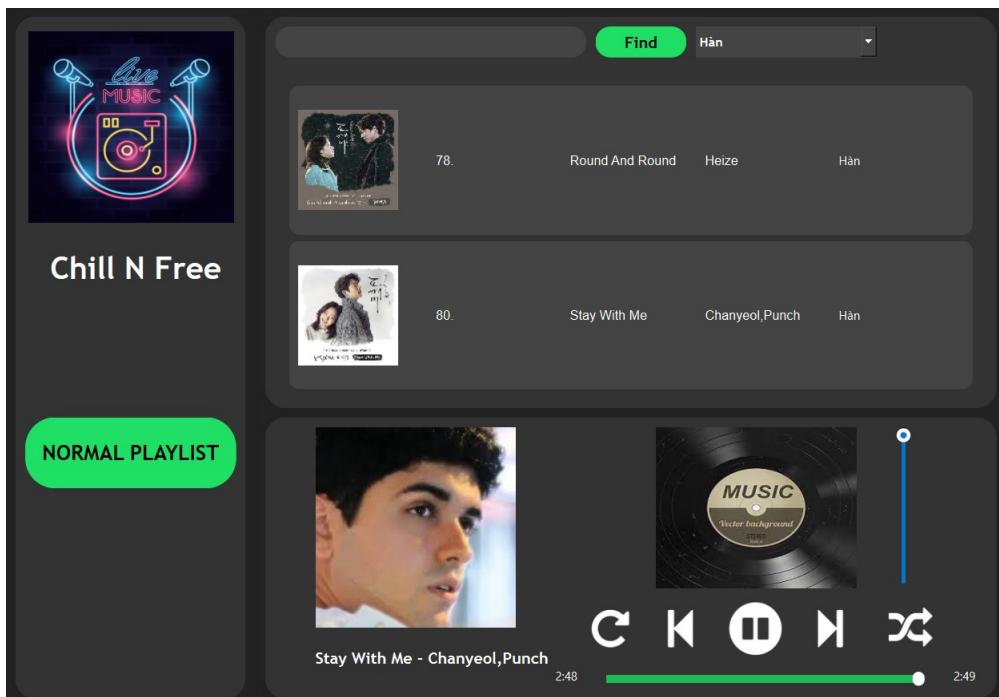
Nhấn chọn tìm kiếm theo Việt



Hình ảnh sau khi tìm kiếm theo Việt



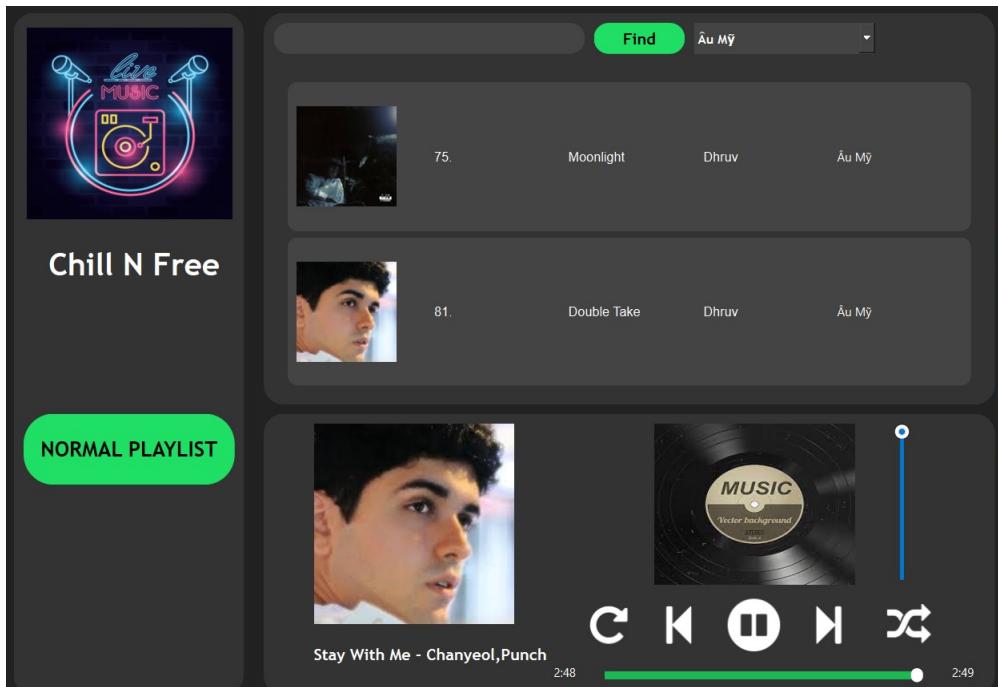
Nhấn chọn tìm kiếm theo Hàn



Hình ảnh sau khi tìm kiếm theo Hàn



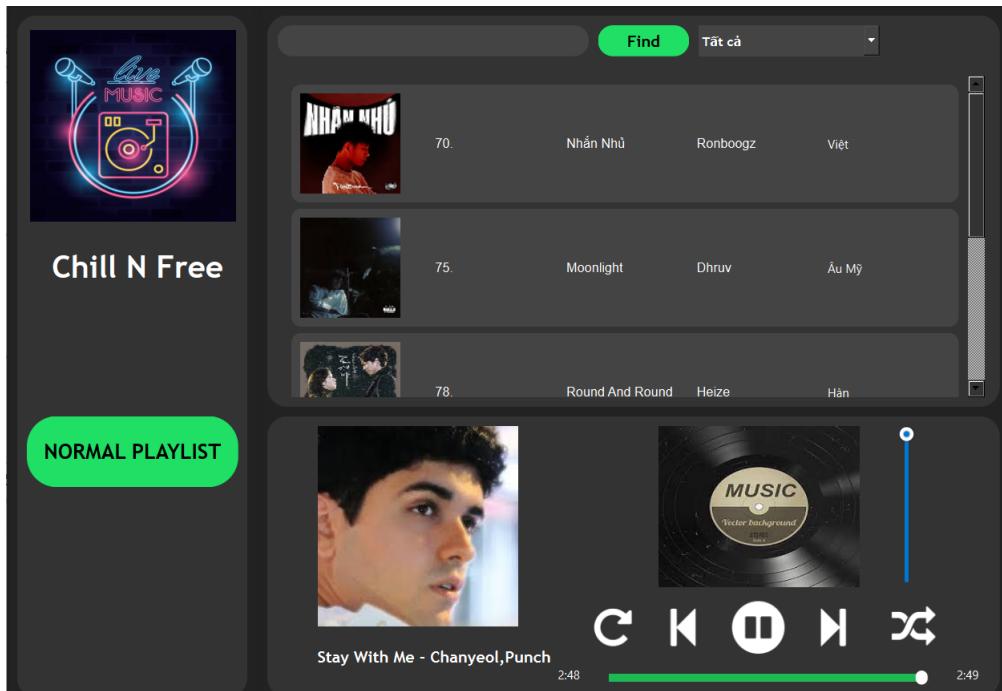
Nhấn chọn tìm kiếm theo Âu Mỹ



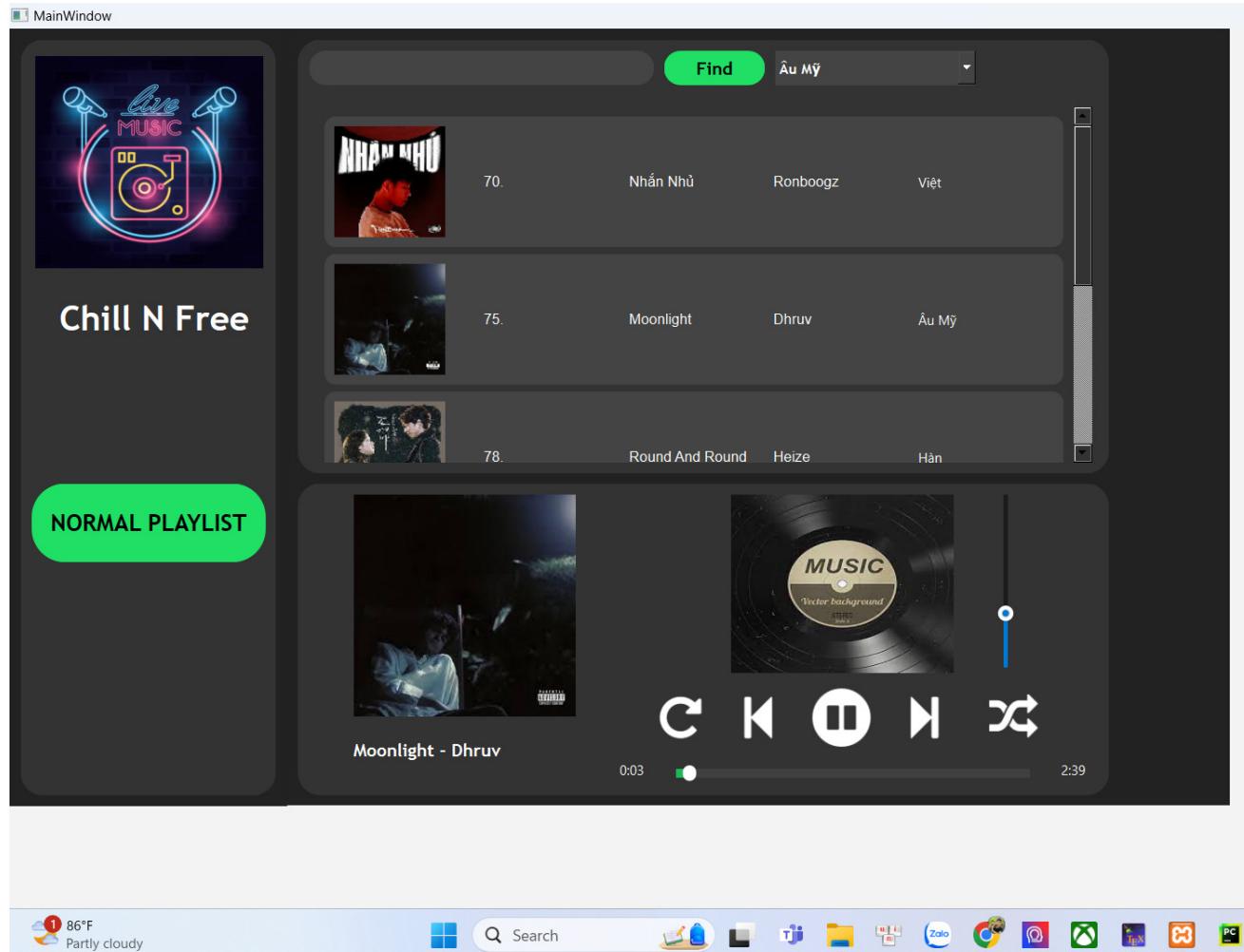
Hình ảnh sau khi tìm kiếm theo Âu Mỹ

```
1 usage
def handle_genre_selection(self):
    arrFind = []
    selected_genre = self.cb_genre.currentText()
    if selected_genre == 'Tất cả':
        self.update_music_list(self.arrList)
    else:
        for item in self.arrList:
            if item['type'] == selected_genre:
                arrFind.append(item)
        self.update_music_list(arrFind)
```

2.2.1.b Chọn bài hát



Hình ảnh trước khi chọn bài hát



Hình ảnh sau khi tìm kiếm bài hát "Moonlight"

CODE :



```
def select_song(self, song_id, play_music=False):
    song_info = None
    for item in self.arrList:
        if item['id'] == song_id:
            song_info = [
                item['id'],
                item['name'],
                item['singer'],
                item['type'],
                item['image_path']
            ]

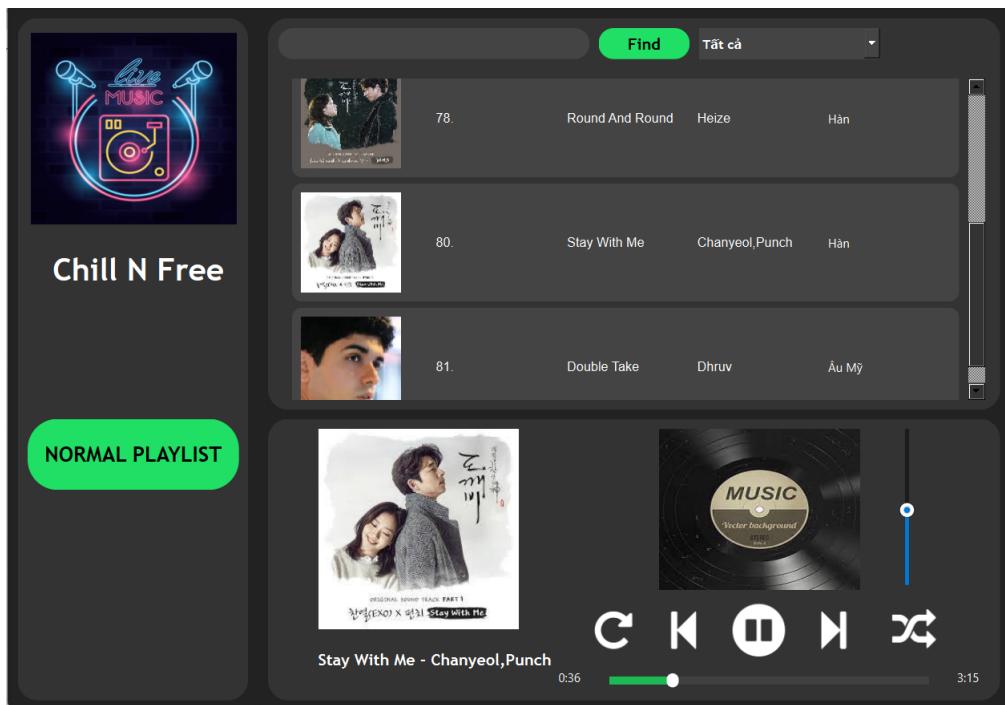
    if song_info:
        song_id = song_info[0]
        song_name = song_info[1]
        singer = song_info[2]
        self.lb_title_2.setText(f"{song_name} - {singer}")
        self.selected_song_id=song_id
        print("Id đang được chọn: ",self.selected_song_id)

    if play_music:
        self.play_selected_song(self.selected_song_id, immediate_play=True)
        self.btn_play.setIcon(icon( *names: "fa.pause-circle", color='white'))
```



2.2.1.c Phát/Dừng bài hát

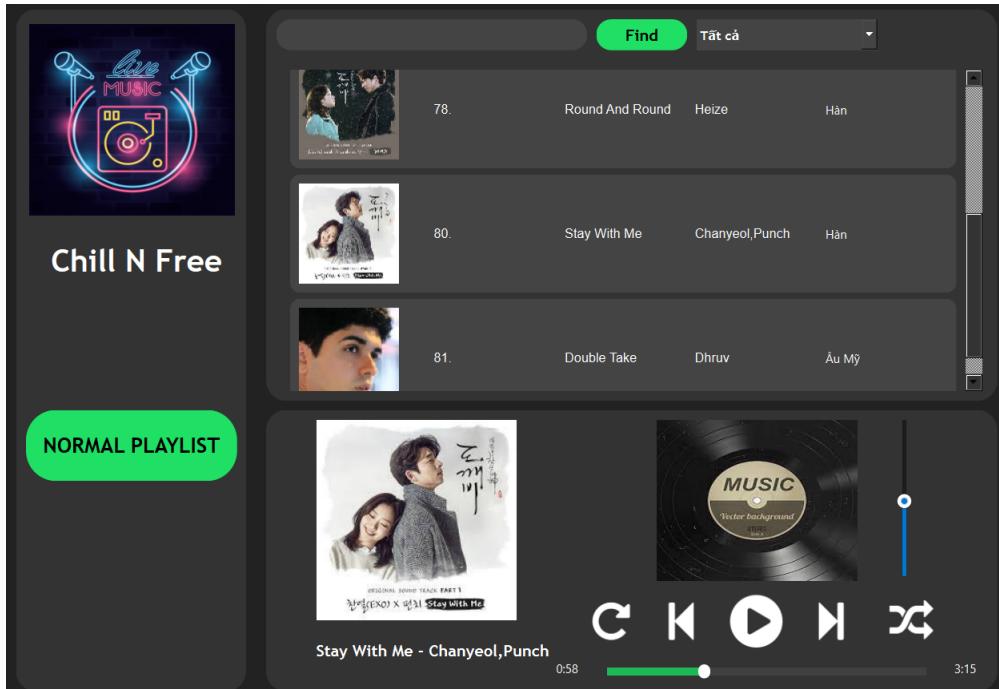
Nhấn nút Play để bắt đầu phát nhạc



Hình ảnh khi phát bài hát



Nhấn nút Pause để dừng phát nhạc



Hình ảnh khi dừng bài hát

CODE :

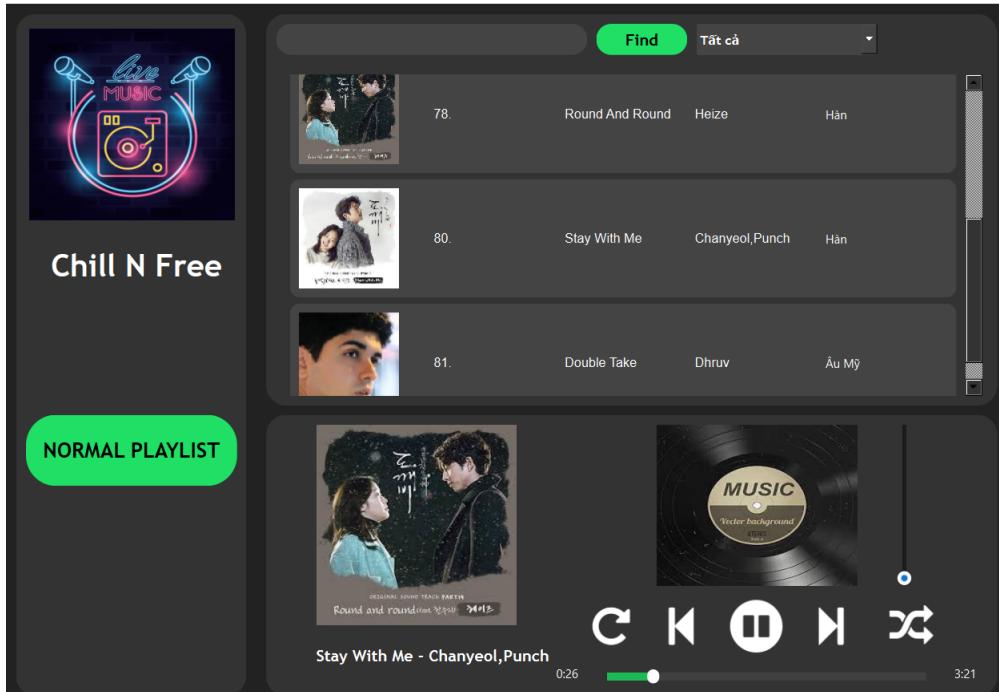
```
1 usage
def play_or_stop(self):
    if self.is_playing:
        pygame.mixer.music.pause()
        self.paused_pos = pygame.mixer.music.get_pos() / 1000.0 # Lưu trữ vị trí hiện tại khi dừng
        self.btn_play.setIcon(icon( "names: "fa.play-circle", color='white'))
        self.timer.stop()
    else:
        if self.paused_pos is not None: # Nếu đã lưu trữ vị trí khi dừng
            pygame.mixer.music.unpause() # Tiếp tục phát nhạc từ vị trí đó
            self.timer.start(1000) # Bắt đầu đồng hồ đếm thời gian
            self.btn_play.setIcon(icon( "names: "fa.pause-circle", color='white'))
            self.paused_pos = None # Đặt lại vị trí đã dừng
        else:
            if pygame.mixer.music.get_busy() and pygame.mixer.music.get_pos() > 0:
                pygame.mixer.music.unpause() # Nếu nhạc đang tạm dừng và có trong trạng thái phát, tiếp tục
            else:
                pygame.mixer.music.play() # Phát từ đầu
                self.update_time_slider() # Cập nhật thanh trượt thời gian
                self.timer.start(1000) # Bắt đầu đồng hồ đếm thời gian
                self.btn_play.setIcon(icon( "names: "fa.pause-circle", color='white'))
    self.is_playing = not self.is_playing

    threading.Thread(target=self.monitor_music_end).start()
```



2.2.1.d Phát bài hát trước đó

Nhấn vào nút mũi tên ngược để quay lại bài Round And Round



Hình ảnh khi phát bài hát trước đó

CODE :

```
def monitor_music_end(self):
    while pygame.mixer.music.get_busy():
        pass
    self.btn_play.setIcon(icon("fa.play-circle", color='white'))
    self.is_playing = False

2 usages
def get_next_song_id(self, direction='next'):
    if self.current_song_id is not None:
        for index, item in enumerate(self.arrList):
            if item['id'] == self.current_song_id:
                if direction == 'next':
                    next_index = (index + 1) % len(self.arrList)
                elif direction == 'pre':
                    next_index = (index - 1) % len(self.arrList)
                return self.arrList[next_index]['id']
    return None
```

Hình Lấy ID của bài hát

```
2 usages
def stop_music(self):
    if self.is_playing:
        pygame.mixer.music.stop()
        self.is_playing = False
    # self.timer.stop() # mới dc thêm
    print("Music stopped.")
```

Dừng bài hát

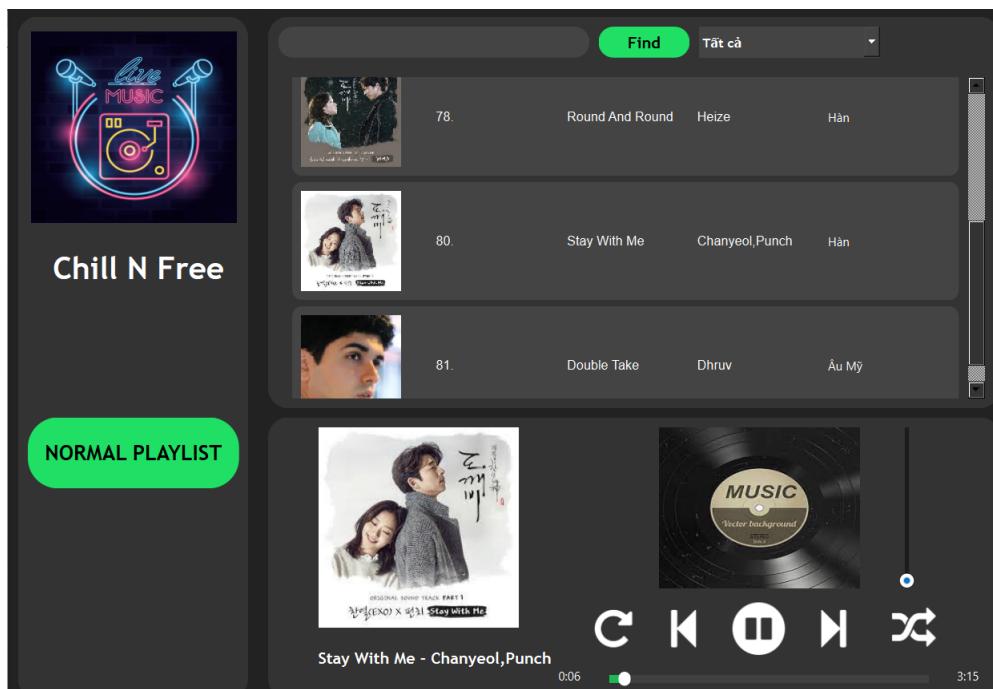


```
1 usage
def play_prev_song(self):
    prev_song_id = self.get_next_song_id(direction='pre')
    if prev_song_id:
        print("Bài hát trước đó:", prev_song_id)
        self.stop_music() # Dừng nhạc trước khi phát bài hát trước đó
        self.play_selected_song(prev_song_id, immediate_play=True) # Phát bài hát trước đó ngay
        self.selected_song_id = prev_song_id
```

Chuyển về bài hát trước

2.2.1.e Phát bài hát sau đó

Nhấn vào nút mũi tên để chuyển tiếp tới bài hát Stay With Me



Hình ảnh khi bài hát tiếp theo được phát

CODE :

```
def monitor_music_end(self):
    while pygame.mixer.music.get_busy():
        pass
    self.btn_play.setIcon(QIcon( "names: "fa.play-circle", color='white'))
    self.is_playing = False

2 usages
def get_next_song_id(self, direction='next'):
    if self.current_song_id is not None:
        for index, item in enumerate(self.arrList):
            if item['id'] == self.current_song_id:
                if direction == 'next':
                    next_index = (index + 1) % len(self.arrList)
                elif direction == 'pre':
                    next_index = (index - 1) % len(self.arrList)
                return self.arrList[next_index]['id']
    return None
```

Hình Lấy ID của bài hát



```
2 usages
def stop_music(self):
    if self.is_playing:
        pygame.mixer.music.stop()
        self.is_playing = False
    # self.timer.stop() # mới dc thêm
    print("Music stopped.")
```

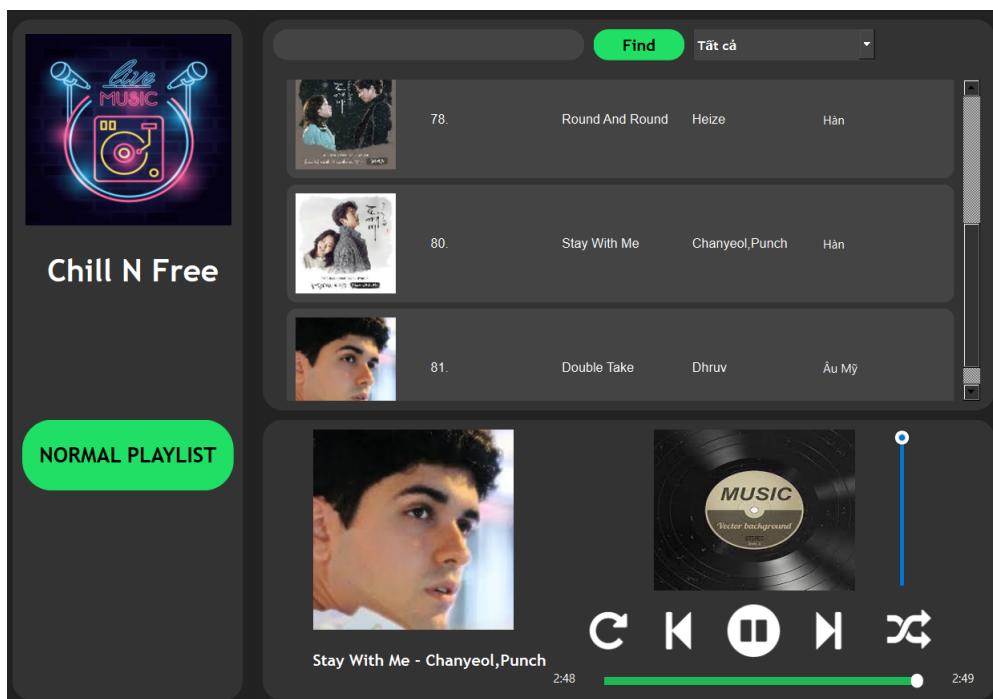
Dừng bài hát

```
def play_next_song(self):
    next_song_id = self.get_next_song_id(direction='next')
    if next_song_id:
        print("id tiếp:", next_song_id)
        self.stop_music() # Dừng nhạc trước khi phát bài hát tiếp theo
        self.play_selected_song(next_song_id, immediate_play=True) # Phát bài hát tiếp theo ngay lập tức
        self.selected_song_id = next_song_id
```

Chuyển về bài hát trước

2.2.1.f Tăng / giảm âm lượng

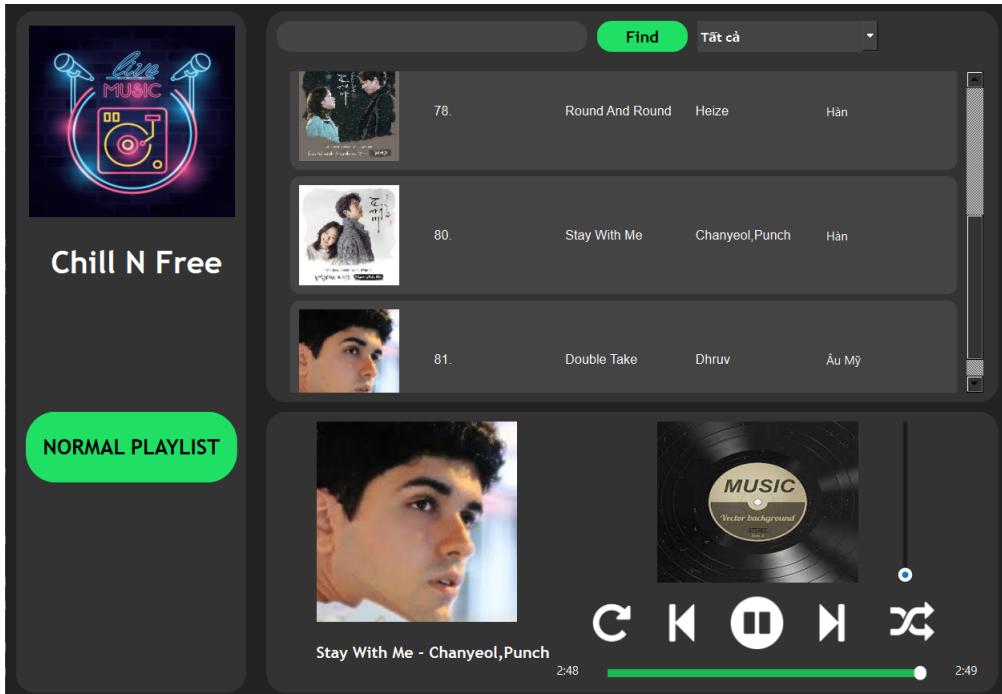
Kéo thanh âm lượng lên để tăng âm lượng



Hình ảnh khi tăng âm lượng



Kéo thanh âm lượng xuống để giảm âm lượng



Hình ảnh khi giảm âm lượng

CODE :

```
self.volume_slider = QtWidgets.QSlider(parent=self.frame_PlayMusic)
self.volume_slider.setGeometry(QtCore.QRect(630, 10, 16, 161))
self.volume_slider.setOrientation(QtCore.Qt.Orientation.Vertical)
self.volume_slider.setMinimum(0)
self.volume_slider.setMaximum(100)
self.volume_slider.setValue(50) # Giá trị mặc định là 50
self.volume_slider.valueChanged.connect(self.on_volume_slider_changed)
self.volume_slider.setObjectName("volume_slider")
```

Cài đặt giá trị Volume

```
def on_volume_slider_changed(self, value):
    # Chuyển đổi giá trị từ phần trăm thành khoảng từ 0.0 đến 1.0
    volume = value / 100.0
    pygame.mixer.music.set_volume(volume)
```

Tăng giảm volume



3 KẾT LUẬN

3.1 Ưu điểm

Phần mềm phát nhạc hiện đại mang lại nhiều lợi ích đáng kể cho người dùng. Dưới đây là một số ưu điểm chính của phần mềm phát nhạc:

Dễ dàng truy cập và quản lý thư viện nhạc:

Phần mềm phát nhạc cho phép người dùng lưu trữ, sắp xếp và truy cập hàng ngàn bài hát chỉ với vài thao tác đơn giản. Người dùng có thể tạo các danh sách phát theo sở thích cá nhân và dễ dàng tìm kiếm bài hát theo tên, nghệ sĩ hoặc album. Chất lượng âm thanh cao:

Phần mềm phát nhạc hiện đại hỗ trợ nhiều định dạng âm thanh chất lượng cao giúp người nghe trải nghiệm âm nhạc với chất lượng âm thanh tốt nhất. Tích hợp nhiều tính năng thông minh:

Các phần mềm phát nhạc thường tích hợp các tính năng như gợi ý bài hát, tạo danh sách phát tự động dựa trên sở thích của người dùng Tùy chỉnh âm thanh :

Phần mềm phát nhạc thường cung cấp các tùy chọn để tùy chỉnh âm thanh Nhữn ưu điểm này làm cho phần mềm phát nhạc trở thành một công cụ không thể thiếu cho những người yêu thích âm nhạc, mang lại trải nghiệm nghe nhạc tốt nhất và tiện lợi nhất.

3.2 Nhược điểm

Tiêu thụ dữ liệu:

Việc phát nhạc trực tuyến tiêu tốn một lượng dữ liệu đáng kể, đặc biệt là khi nghe nhạc chất lượng cao. Điều này có thể dẫn đến chi phí dữ liệu cao đối với người dùng có gói dữ liệu hạn chế. Phụ thuộc vào phần mềm:

Việc sử dụng phần mềm phát nhạc có thể khiến người dùng trở nên phụ thuộc vào các ứng dụng này để nghe nhạc. Nếu phần mềm gặp sự cố hoặc không được cập nhật, trải nghiệm nghe nhạc của người dùng có thể bị ảnh hưởng. Giới hạn về nội dung:

Mặc dù có rất nhiều bài hát có sẵn, nhưng không phải tất cả các bài hát đều có mặt trên tất cả các nền tảng phát nhạc. Điều này có thể gây khó khăn cho người dùng khi tìm kiếm các bản nhạc hiếm hoặc độc quyền. Vấn đề về bản quyền:

Các vấn đề về bản quyền có thể dẫn đến việc một số bài hát bị gỡ bỏ khỏi thư viện hoặc không có sẵn ở một số quốc gia. Điều này có thể làm hạn chế lựa chọn nhạc của người dùng. Phụ thuộc vào thiết bị:

Chất lượng âm thanh và trải nghiệm nghe nhạc có thể phụ thuộc vào thiết bị phần cứng mà người dùng sử dụng. Nếu không có thiết bị âm thanh chất lượng cao, người dùng có thể không tận hưởng được trọn vẹn chất lượng âm nhạc. Những nhược điểm này cần được cân nhắc khi lựa chọn và sử dụng phần mềm phát nhạc, để người dùng có thể tận dụng tối đa các lợi ích mà phần mềm mang lại trong khi giảm thiểu các vấn đề tiềm ẩn.