

**HỌC VIỆN KỸ THUẬT MẬT MÃ**  
**KHOA CÔNG NGHỆ THÔNG TIN**



**ĐỒ ÁN MÔN HỌC**  
**THỰC TẬP SƠ SỞ**

**Đề tài:**  
**XÂY DỰNG ỨNG DỤNG NGHE NHẠC ONLINE**  
**TRÊN ANDROID**

**Sinh viên thực hiện:** Nguyễn Văn Tú AT160452  
Nguyễn Đức Long AT160429  
Nguyễn Hải Dương AT160706

**Giảng viên hướng dẫn** ThS. Cao Thanh Vinh

*Hà Nội, tháng 4 năm 2022*

## NHẬN XÉT CỦA GIẢNG VIÊN

Điểm

Hà Nội, ngày    tháng    năm 2022

**Người nhận xét**

(Ký và ghi rõ họ tên)

# MỤC LỤC

<b>MỤC LỤC .....</b>	<b>i</b>
<b>DANH MỤC HÌNH ẢNH .....</b>	<b>iii</b>
<b>DANH MỤC BẢNG .....</b>	<b>v</b>
<b>CHƯƠNG I. CƠ SỞ LÝ THUYẾT .....</b>	<b>1</b>
1.1 GIỚI THIỆU CHUNG HỆ ĐIỀU HÀNH ANDROID .....	1
1.2 CÔNG CỤ SỬ DỤNG .....	2
1.2.1 Tổng quan về ngôn ngữ lập trình Java .....	2
1.2.2 Tổng quan về Android Studio .....	5
1.2.3 Tổng quan về Framework Laravel .....	6
1.2.4 Tổng quan về PhpMyAdmin .....	7
<b>CHƯƠNG II. PHÂN TÍCH, THIẾT KẾ HỆ THỐNG .....</b>	<b>9</b>
2.1 PHÂN TÍCH HỆ THỐNG VỀ CHỨC NĂNG .....	9
2.1.1 Bài toán tổng quát và khảo sát thực trạng .....	9
2.1.2 Các tính năng cơ bản của ứng dụng .....	11
2.2. PHÂN TÍCH HỆ THỐNG VỀ DỮ LIỆU .....	11
2.2.1 Biểu đồ use case, đặt tả use case .....	11
2.2.2 Biểu đồ Use Case .....	12
2.2.4 Xây dựng biểu đồ lớp (Class diagram) .....	19
2.2.5 Biểu đồ tuần tự .....	20
2.2.5 Biểu đồ hoạt động của người dùng.....	24
2.3. THIẾT KẾ VỀ CƠ SỞ DỮ LIỆU .....	25
2.3.1 Các bảng của cơ sở dữ liệu .....	25
2.3.2 Mối quan hệ giữa các bảng trong cơ sở dữ liệu.....	31
<b>CHƯƠNG III. KẾT QUẢ THỰC NGHIỆM .....</b>	<b>32</b>
3.1. GIAO DIỆN NGƯỜI DÙNG .....	32
3.1.1 Giao diện màn hình đăng nhập .....	32
3.1.2 Giao diện màn hình đăng ký .....	33
3.1.3 Giao diện màn hình chính ứng dụng .....	34

3.1.4 Giao diện màn hình tìm kiếm .....	36
3.1.5 Giao diện màn hình thông tin người dùng .....	37
3.1.6 Giao diện màn hình danh sách bài hát .....	38
3.1.7 Giao diện màn hình phát nhạc .....	38
3.1.8 Giao diện màn hình đăng xuất .....	39
3.1.8 Giao diện màn hình chia sẻ bài hát .....	39
3.2. GIAO DIỆN CHO ADMIN .....	40
3.2.1 Giao diện màn hình quản lý bài hát .....	40
3.2.2 Giao diện màn hình quảng trị chủ đề .....	40
3.2.3 Giao diện màn hình trang quản trị tài khoản người dùng .....	41
3.2.4 Giao diện màn hình trang quản trị nghệ sĩ .....	41
3.3. KẾT LUẬN CHƯƠNG III .....	42
1. Những điểm ứng dụng làm được .....	42
2. Nhược điểm của ứng dụng.....	42
3. Hướng phát triển của ứng dụng .....	42
4. Những kiến thức kỹ năng đạt được sau khi thực hiện đồ án .....	42
<b>TÀI LIỆU THAM KHẢO .....</b>	<b>43</b>

## DANH MỤC HÌNH ẢNH

Hình 1.1. Giao diện ứng dụng “Zing Mp3” .....	10
Hình 1.2. Giao diện ứng dụng “Nhạc của tui” .....	10
Hình 2.1. Các tính năng cơ bản của ứng dụng .....	11
Hình 2.3. Biểu đồ Use case tổng quát .....	11
Hình 2.4. Use case đăng nhập quản trị: .....	12
Hình 2.5. Use case nghe nhạc .....	12
Hình 2.6. Biểu đồ lớp .....	20
Hình 2.7. Biểu đồ tuần tự của chức năng đăng nhập .....	21
Hình 2.8. Biểu đồ tuần tự của chức năng thêm bài hát .....	22
Hình 2.9. Biểu đồ tuần tự của chức năng tìm kiếm bài hát .....	23
Hình 2.10. Biểu đồ hoạt động của người dùng .....	24
Hình 2.11. Bảng người dùng .....	25
Hình 2.12. Bảng Bài hát .....	25
Hình 2.13. Bảng chủ đề .....	26
Hình 2.14. Bảng xếp hạng .....	27
Hình 2.15. Bảng Bài hát thư viện Playlist .....	27
Hình 2.16. Bảng Nghệ sĩ .....	28
Hình 2.17. Bảng phổ biến .....	28
Hình 2.18. Bảng Người dùng yêu thích .....	29
Hình 2.19. Bảng Playlist .....	29
Hình 2.20. Bảng Thịnh hành .....	30
Hình 2.21. Bảng Thư viện Playlist .....	30
Hình 2.22. Mối quan hệ giữa các bảng trong cơ sở dữ liệu .....	31
Hình 3.1. Giao diện màn hình đăng nhập .....	32
Hình 3.2. Giao diện màn hình đăng ký .....	33
Hình 3.3. Khi đăng nhập với tư cách “Khách” .....	34
Hình 3.4. Khi đăng nhập với tư cách “Người dùng” .....	35
Hình 3.5. Giao diện màn hình tìm kiếm .....	36

Hình 3.6. Giao diện màn hình thư viện của tôi .....	36
Hình 3.7. Giao diện màn hình thông tin người dùng .....	37
Hình 3.8. Giao diện màn hình danh sách bài hát .....	38
Hình 3.9. Giao diện màn hình phát nhạc .....	38
Hình 3.10. Giao diện màn hình đăng xuất .....	39
Hình 3.11 Giao diện chia sẻ bài hát .....	39
Hình 3.12. Giao diện màn hình quản lý bài hát .....	40
Hình 3.13. Giao diện màn hình quảng trị chủ đề .....	40
Hình 3.14. Giao diện màn hình trang quản trị tài khoản người dùng .....	41
Hình 3.15. Giao diện màn hình trang quản trị nghệ sĩ .....	41

## DANH MỤC BẢNG

Bảng 2.1. Đặc tả Use Case Đăng ký tài khoản .....	13
Bảng 2.2. Đặc tả Use Case Đăng nhập tài khoản .....	14
Bảng 2.3. Đặc tả Use Case Tìm kiếm bài hát .....	14
Bảng 2.4. Đặc tả Use Case nghe nhạc .....	15
Bảng 2.5. Đặc tả Use Case chọn Album và bài hát yêu thích .....	16
Bảng 2.6. Đặc tả Use Case xem các quảng cáo bài hát chủ đề .....	17
Bảng 2.7. Đặc tả Use Case Quên mật khẩu .....	18
Bảng 2.8. Đặc tả Use Case quản lý CSDL .....	19

# CHƯƠNG I. CƠ SỞ LÝ THUYẾT

## 1.1 GIỚI THIỆU CHUNG HỆ ĐIỀU HÀNH ANDROID

### Android là gì?

Android là một hệ điều hành dựa trên nền tảng Linux được thiết kế dành cho các thiết bị di động có màn hình cảm ứng như điện thoại thông minh và máy tính bảng. Ban đầu, Android được phát triển bởi Android Inc. với sự hỗ trợ tài chính từ Google và sau này được chính Google mua lại vào năm 2005.

Android ra mắt vào năm 2007 cùng với tuyên bố thành lập Liên minh thiết bị cầm tay mở: một hiệp hội gồm các công ty phần cứng, phần mềm, và viễn thông với mục tiêu đẩy mạnh các tiêu chuẩn mở cho các thiết bị di động. Chiếc điện thoại đầu tiên chạy Android được bán vào năm 2008.

Android có mã nguồn mở và Google phát hành mã nguồn theo Giấy phép Apache. Chính mã nguồn mở cùng với một giấy phép không có nhiều ràng buộc đã cho phép các nhà phát triển thiết bị, mạng di động và các lập trình viên nhiệt huyết được điều chỉnh và phân phối Android một cách tự do. Ngoài ra, Android còn có một cộng đồng lập trình viên đông đảo chuyên viết các ứng dụng để mở rộng chức năng của thiết bị, bằng một loại ngôn ngữ lập trình Java có sửa đổi. Tháng 10 năm 2012, có khoảng 700.000 ứng dụng trên Android, và số lượt tải ứng dụng từ Google Play, cửa hàng ứng dụng chính của Android, ước tính khoảng 25 tỷ lượt.

### Android có gì khác biệt?

Android đã thu hút được sự chú ý của giới công nghệ khắp toàn cầu khi đứa con của Google sẽ sử dụng giấy phép mã nguồn mở. Đó là một sản phẩm kết tinh từ ý tưởng của Khôi Liên minh thiết bị cầm tay mở do Google dẫn đầu, gồm 34 thành viên với các công ty hàng đầu về công nghệ và di động toàn cầu như Qualcomm, Intel, Motorola, Texas Instruments và LG Electronics, các nhà mạng như T-Mobile, Sprint Nextel, NTT DoCoMo và China Mobile.

Các nhà phát triển có thể sử dụng miễn phí bộ Android Software Development Kit (SDK) để xây dựng các ứng dụng của mình.



Android sẽ hỗ trợ:

- Màn hình cảm ứng (đa điểm)
- 3G
- Wifi
- Trình duyệt dựa trên webkit
- Tin nhắn (SMS) theo luồng
- Định dạng MPEG-4, H.264, MP3, AAC
- Bộ tăng tốc đồ họa 3D

Android chiếm 87,7% thị phần điện thoại thông minh trên toàn thế giới vào thời điểm quý 2 năm 2017, với tổng cộng 2 tỷ thiết bị đã được kích hoạt và 1,3 triệu lượt kích hoạt mỗi ngày. Sự thành công của hệ điều hành cũng khiến nó trở thành mục tiêu trong các vụ kiện liên quan đến bằng phát minh, góp mặt trong cái gọi là "cuộc chiến điện thoại thông minh" giữa các công ty công nghệ.

## **1.2 CÔNG CỤ SỬ DỤNG**

### ***1.2.1 Tổng quan về ngôn ngữ lập trình Java***

#### *Ngôn ngữ lập trình Java là gì?*

Java là một ngôn ngữ lập trình hiện đại, bậc cao, bảo mật và mạnh mẽ. Nó được sử dụng trong phát triển phần mềm, trang web, game hay các ứng dụng trên thiết bị di động.

Java được phát hành năm 1994, đến năm 2010 được Oracle mua lại từ Sun Microsystems. Ban đầu Java được tạo ra nhằm mục đích viết phần mềm cho các sản phẩm gia dụng, và có tên là Oak.

Điều khác biệt nhất giữa Java và các ngôn ngữ lập trình khác chính là Java có thể thực hiện nhiều tác vụ mà bạn chỉ cần thiết lập chúng chạy một lần duy nhất. Với tiêu chí: “Viết một lần, thực thi khắp nơi” (Write Once, Run Anywhere – WORA). Chương trình phần mềm viết bằng Java có thể chạy trên mọi nền tảng (platform) khác nhau thông qua một môi trường thực thi với điều kiện có môi trường thực thi thích hợp hỗ trợ nền tảng đó.

#### *Đặc điểm của ngôn ngữ lập trình Java*

*Tương tự C++, hướng đối tượng hoàn toàn*

Trong quá trình tạo ra một ngôn ngữ mới phục vụ cho mục đích chạy được trên nhiều nền tảng, các kỹ sư của Sun Microsystems muốn tạo ra một ngôn ngữ dễ học và quen thuộc với đa số người lập trình. Vì vậy họ đã sử dụng lại các cú pháp của C và C++.

Tuy nhiên, trong Java thao tác với con trỏ bị lược bỏ nhằm đảm bảo tính an toàn và dễ sử dụng hơn. Các thao tác overload, goto hay các cấu trúc như struct và union cũng được loại bỏ khỏi Java.

### *Độc lập phần cứng và hệ điều hành*

Một chương trình viết bằng ngôn ngữ Java có thể chạy tốt ở nhiều môi trường khác nhau. Gọi là khả năng “cross-platform”. Khả năng độc lập phần cứng và hệ điều hành được thể hiện ở 2 cấp độ là cấp độ mã nguồn và cấp độ nhị phân.

Ở cấp độ mã nguồn: Kiểu dữ liệu trong Java nhất quán cho tất cả các hệ điều hành và phần cứng khác nhau. Java có riêng một bộ thư viện để hỗ trợ vấn đề này. Chương trình viết bằng ngôn ngữ Java có thể biên dịch trên nhiều loại máy khác nhau mà không gặp lỗi.

Ở cấp độ nhị phân: Một mã biên dịch có thể chạy trên nhiều nền tảng khác nhau mà không cần dịch lại mã nguồn. Tuy nhiên cần có Java Virtual Machine để thông dịch đoạn mã này.

### *Ngôn ngữ thông dịch*

Ngôn ngữ lập trình thường được chia ra làm 2 loại (tùy theo các hiện thực hóa ngôn ngữ đó) là ngôn ngữ thông dịch và ngôn ngữ biên dịch.

Thông dịch (Interpreter) : Nó dịch từng lệnh rồi chạy từng lệnh, lần sau muốn chạy lại thì phải dịch lại.

Biên dịch (Compiler): Code sau khi được biên dịch sẽ tạo ra 1 file thường là .exe, và file .exe này có thể đem sử dụng lại không cần biên dịch nữa.

Ngôn ngữ lập trình Java thuộc loại ngôn ngữ thông dịch. Chính xác hơn, Java là loại ngôn ngữ vừa biên dịch vừa thông dịch.

### *Cơ chế thu gom rác tự động*

Khi tạo ra các đối tượng trong Java, JRE sẽ tự động cấp phát không gian bộ nhớ cho các đối tượng ở trên heap.

Với ngôn ngữ như C \ C++, bạn sẽ phải yêu cầu hủy vùng nhớ mà bạn đã cấp phát, để tránh việc thất thoát vùng nhớ. Tuy nhiên vì một lý do nào đó, bạn không hủy một vài vùng nhớ, dẫn đến việc thất thoát và làm giảm hiệu năng chương trình.

Ngôn ngữ lập trình Java hỗ trợ cho bạn điều đó, nghĩa là bạn không phải tự gọi hủy các vùng nhớ. Bộ thu dọn rác của Java sẽ theo vết các tài nguyên đã được cấp. Khi không có tham chiếu nào đến vùng nhớ, bộ thu dọn rác sẽ tiến hành thu hồi vùng nhớ đã được cấp phát.

### *Đa luồng*

Java hỗ trợ lập trình đa tiến trình (multithread) để thực thi các công việc đồng thời. Đồng thời cũng cung cấp giải pháp đồng bộ giữa các tiến trình (giải pháp sử dụng priority...).

### *Tính an toàn và bảo mật*

#### Tính an toàn

- Ngôn ngữ lập trình Java yêu cầu chặt chẽ về kiểu dữ liệu.
- Dữ liệu phải được khai báo tường minh.
- Không sử dụng con trỏ và các phép toán với con trỏ.
- Java kiểm soát chặt chẽ việc truy nhập đến mảng, chuỗi. Không cho phép sử dụng các kỹ thuật tràn. Do đó các truy nhập sẽ không vượt quá kích thước của mảng hoặc chuỗi.
- Quá trình cấp phát và giải phóng bộ nhớ được thực hiện tự động.
- Cơ chế xử lý lỗi giúp việc xử lý và phục hồi lỗi dễ dàng hơn.

#### Tính bảo mật

Java cung cấp một môi trường quản lý chương trình với nhiều mức khác nhau.

Mức 1 : Chỉ có thể truy xuất dữ liệu cũng như phương thức thông qua giao diện mà lớp cung cấp.

Mức 2 : Trình biên dịch kiểm soát các đoạn mã sao cho tuân thủ các quy tắc của ngôn ngữ lập trình Java trước khi thông dịch.

Mức 3 : Trình thông dịch sẽ kiểm tra mã byte code xem các đoạn mã này có đảm bảo được các quy định, quy tắc trước khi thực thi.

Mức 4: Java kiểm soát việc nạp các lớp vào bộ nhớ để giám sát việc vi phạm giới hạn truy xuất trước khi nạp vào hệ thống.

### ***1.2.2 Tổng quan về Android Studio***

#### **Android Studio là gì?**

Android Studio là IDE chính thức được sử dụng trong phát triển ứng dụng Android dựa trên IntelliJ IDEA.

Chức năng chính của Android Studio là cung cấp các giao diện giúp người dùng có thể tạo các ứng dụng và xử lý các công cụ file phức tạp sau hậu trường. Ngôn ngữ lập trình được sử dụng trong Android Studio là Java và nó sẽ được cài đặt sẵn trên thiết bị của bạn.

Khi sử dụng Android Studio thì bạn chỉ cần viết, chỉnh sửa và lưu trữ chúng trên các dự án của mình và các file nằm trong dự án đó. Đồng thời, Android Studio còn cung cấp quyền truy cập vào Android SDK.

Ngoài ra, bạn có thể xem Android Studio là đuôi cho code Java cho phép nó chạy trơn tru trên các thiết bị Android rồi tận dụng được lợi thế của phần cứng gốc. Bạn chỉ cần sử dụng ngôn ngữ lập trình Java để có thể viết chương trình, khi đó Android SDK sẽ có nhiệm vụ kết nối các phần này lại với nhau.

Khi đó, Android Studio sẽ kích hoạt để thực hiện chạy code và thông qua trình giả lập hoặc dựa vào bất kỳ phần cứng giúp kết nối với thiết bị. Sau đó, bạn có thể gỡ rối cho chương trình của mình ngay khi nó chạy và nhận phản hồi giúp giải thích các sự cố,...

Cho đến hiện nay, Google đã và đang rất nỗ lực để giúp cho Android Studio sẽ trở nên mạnh mẽ và hữu ích hơn. Khi bạn gõ code, nó sẽ giúp bạn cung cấp danh sách gợi ý hoàn thành để giúp người dùng có thể hoàn thiện được dòng code đó. Đây là một trong những chức năng rất hữu ích để phòng trường hợp người dùng không nhớ chính xác cú pháp giúp bạn tiết kiệm thời gian hiệu quả hơn.

### ***1.2.3 Tổng quan về Framework Laravel***

Laravel là một PHP framework mã nguồn mở và miễn phí, được phát triển bởi Taylor Otwell và nhắm vào mục tiêu hỗ trợ phát triển các ứng dụng website theo kiến trúc model-view-controller (MVC). Những tính năng nổi bật của Laravel bao gồm cú pháp dễ hiểu – rõ ràng, một hệ thống đóng gói modular và quản lý gói phụ thuộc, nhiều cách khác nhau để truy cập vào các cơ sở dữ liệu quan hệ, nhiều tiện ích khác nhau hỗ trợ việc triển khai vào bảo trì ứng dụng.

#### ***Những ưu điểm khi sử dụng laravel:***

##### ***Bắt đầu dễ dàng***

Lý do đầu tiên phải kể đến chính việc Laravel có thể sử dụng cực dễ dàng cho người mới bắt đầu, kể cả khi bạn chỉ biết sơ về PHP, bạn vẫn có khả năng phát triển 1 website với 5 trang trong vòng vài giờ.

##### ***Có mã nguồn mở***

Framework Laravel với mã nguồn mở miễn phí, cho phép bạn xây dựng ứng dụng web lớn và phức tạp một cách nhanh chóng, dễ dàng.

Tất cả điều bạn cần làm ở đây là cài đặt PHP, cộng với một trình soạn thảo văn bản để bắt đầu.

##### ***Hỗ trợ cộng đồng***

Trường hợp bạn mắc kẹt với những bug khó nhằn, nhưng có người đã trải qua và hướng dẫn bạn thì không còn gì tuyệt hơn.

Laravel có một hệ thống thư viện hỗ trợ vô cùng lớn và mạnh hơn hẳn các Framework khác.

Nếu bạn report lỗi hoặc vi phạm bảo mật trong Framework, phản hồi của cộng đồng sẽ rất nhanh chóng.

##### ***Theo dõi MVC***

Cấu trúc MVC và lập trình hướng đối tượng OOP vẫn được giữ lại trong Framework Laravel, giúp cung cấp tài liệu tốt hơn, và tăng hiệu suất hơn.

##### ***Được xây dựng dựa trên những Framework hiệu quả nhất***

Vì ra đời muộn nên Laravel được thừa hưởng những ưu điểm và thế mạnh của Framework khác, khi có phần route cực mạnh. Ví dụ điển hình là Laravel sử dụng một số thành phần tốt nhất của Symfony.

### *Di chuyển Database dễ dàng*

Di chuyển Database là một trong những tính năng trọng yếu của Laravel. Nó cho phép bạn duy trì cấu trúc cơ sở dữ liệu ứng dụng mà không nhất thiết phải tạo lại.

Di chuyển Database còn cho phép bạn viết mã PHP để kiểm soát Database, thay vì sử dụng SQL. Nó còn cho phép bạn khôi phục những thay đổi gần đây nhất trong Database.

### *Tính năng bảo mật hoàn thiện*

Ứng dụng của bạn sẽ an toàn khi dùng Framework Laravel. Kỹ thuật ORM của Laravel sử dụng PDO, chống chèn SQL.

Ngoài ra, tính năng bảo vệ csrf của Laravel giúp ngăn chặn giả mạo từ yêu cầu trang chéo. Đó là cú pháp tự động thoát bất kì HTML nào đang được truyền qua các tham số

Ngoài ra để tối ưu cho việc code ta còn sử dụng thêm framework bootstrap với nhiều tính năng giúp người lập trình hoàn thiện giao diện một cách tốt nhất.

### **1.2.4 Tổng quan về PhpMyAdmin**

PhpMyAdmin là một phần mềm có mã nguồn mở được viết bằng ngôn ngữ lập trình PHP. Người sử dụng phần mềm này thường là nhà quản trị các cơ sở dữ liệu, các database administrator, thiết kế website (web designer)....

Nhờ vào việc sử dụng phần mềm này, người dùng có thể quản lý các dữ liệu MySQL thông qua giao diện web thay vì sử dụng giao diện cửa sổ dòng lệnh truyền thống.

Cài đặt Phpmyadmin trên Windows, Ubuntu, Mac giúp cho người dùng dễ dàng thực hiện được nhiều tác vụ khác nhau như cập nhật, tạo mới, sửa đổi hoặc xóa bỏ cơ sở dữ liệu, bản ghi hoặc bảng, thực hiện các báo cáo SQL, phân quyền và quản lý người dùng,...

### *Những tính năng nổi bật của PhpMyAdmin*

Tăng hiệu quả quản lý: Cài đặt Phpmyadmin trên Windows, Ubuntu, Mac giúp cho người dùng nhanh chóng đạt được hiệu quả hơn so với việc cài đặt trên cửa sổ dòng lệnh thông thường. Phần mềm này được thiết kế để các công việc

phổ biến như xem danh sách cơ sở dữ liệu trên máy chủ, xem cấu trúc bảng, chèn dữ liệu, thay đổi dữ liệu một cách nhanh chóng.

Giao diện web thân thiện: Phpmyadmin với giao diện web thân thiện sẽ giúp cho người dùng nhanh chóng nắm được thông tin là có bao nhiêu database (cơ sở dữ liệu) được tạo ra trên server. Bên cạnh đó, khi nhập dữ liệu người dùng cũng có thể biết được kiểu dữ liệu nhập đúng cho từng trường để tránh trường hợp dữ liệu không chính xác. Đây là những tính năng thuận lợi không có trong cửa sổ dòng lệnh ngày trước.

Chế độ xem trước (Design View): Không chỉ dừng lại ở các tác vụ thông thường mà Phpmyadmin cho phép người dùng có thể xem được thiết kế của cơ sở dữ liệu đã nhập vào một cách trực quan.

## CHƯƠNG II. PHÂN TÍCH, THIẾT KẾ HỆ THỐNG

### 2.1 PHÂN TÍCH HỆ THỐNG VỀ CHỨC NĂNG

#### 2.1.1 Bài toán tổng quát và khảo sát thực trạng

##### Bài toán tổng quát:

- Bối cảnh thực hiện: Tháng 5 năm 2022, thực hiện xây dựng ứng nghe nhạc online trên hệ điều hành Android bởi sinh viên Nhóm sinh viên gồm: Nguyễn Văn Tú, Nguyễn Đức Long, Nguyễn Hải Dương đang học tại Khoa An toàn thông tin – Học viện Kỹ thuật mật mã để thực hiện cho yêu cầu của môn Thực tập cơ sở.

- Vấn đề cần giải quyết: Xây dựng một ứng dụng nghe nhạc trên hệ điều hành Android đáp ứng đầy đủ các nhu cầu của người dùng trong một về một ứng dụng nghe nhạc và các yêu cầu về quản lý cho người quản trị.

- Nội dung tổng quan:

- Xây dựng ứng dụng nghe nhạc bằng Android Studio, với mô hình MVC giúp việc quản lý mã nguồn được dễ dàng hơn. Ứng dụng cho phép người dùng có thể xem các bài hát mới, bài hát theo chủ đề và thể loại, các bài hát được nhiều người yêu thích được cập nhật liên tục trên ứng dụng.

- Xây dựng backend sử dụng framework Laravel dùng để quản lý cơ sở dữ liệu. Giúp thuận lợi hơn trong cập nhật dữ liệu của ứng dụng.

- Mục đích:

- Ngày càng hoàn thiện các chức năng để đáp ứng đầy đủ các yêu cầu của người dùng và tối ưu hóa ứng dụng để tăng trải nghiệm cho người dùng.

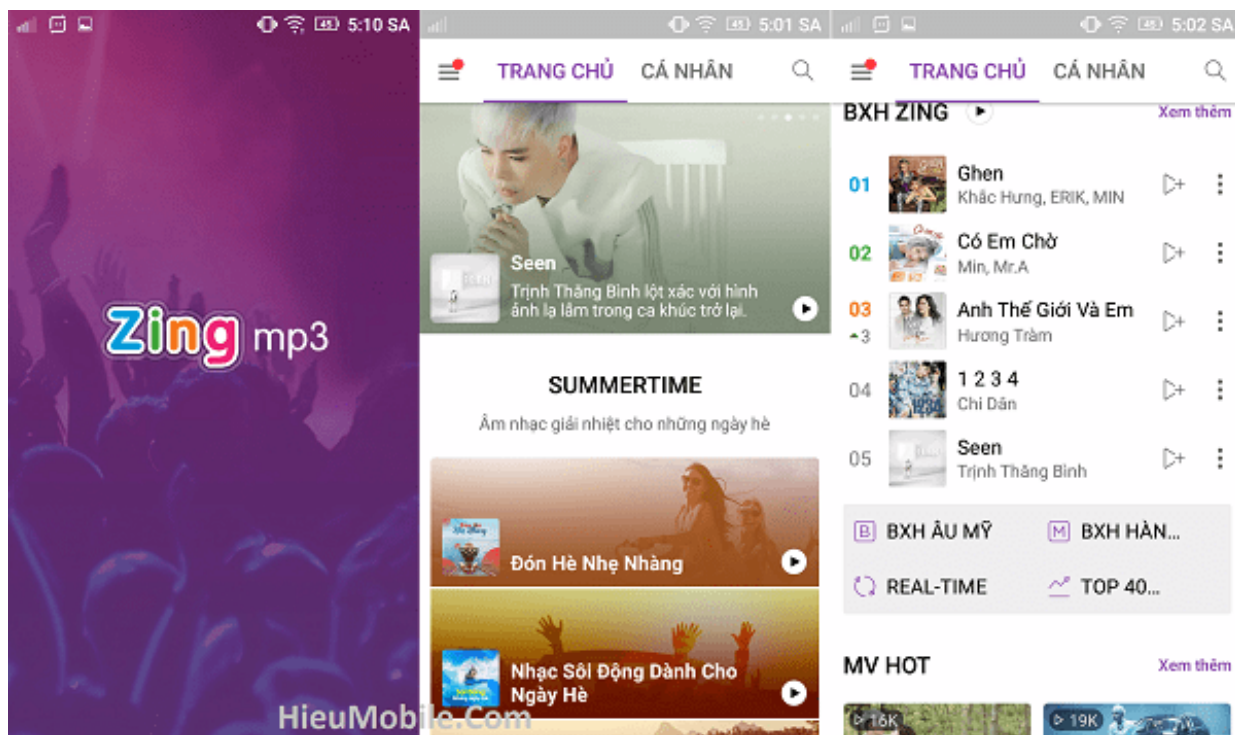
- Xây dựng một ứng dụng nghe nhạc nổi tiếng có lượt tải về cao mang lại nhiều lợi nhuận.

- Phạm vi: Hiện nay, quy mô trong nội thành Hà Nội. Tương lai hướng đến phạm vi trên cả nước.

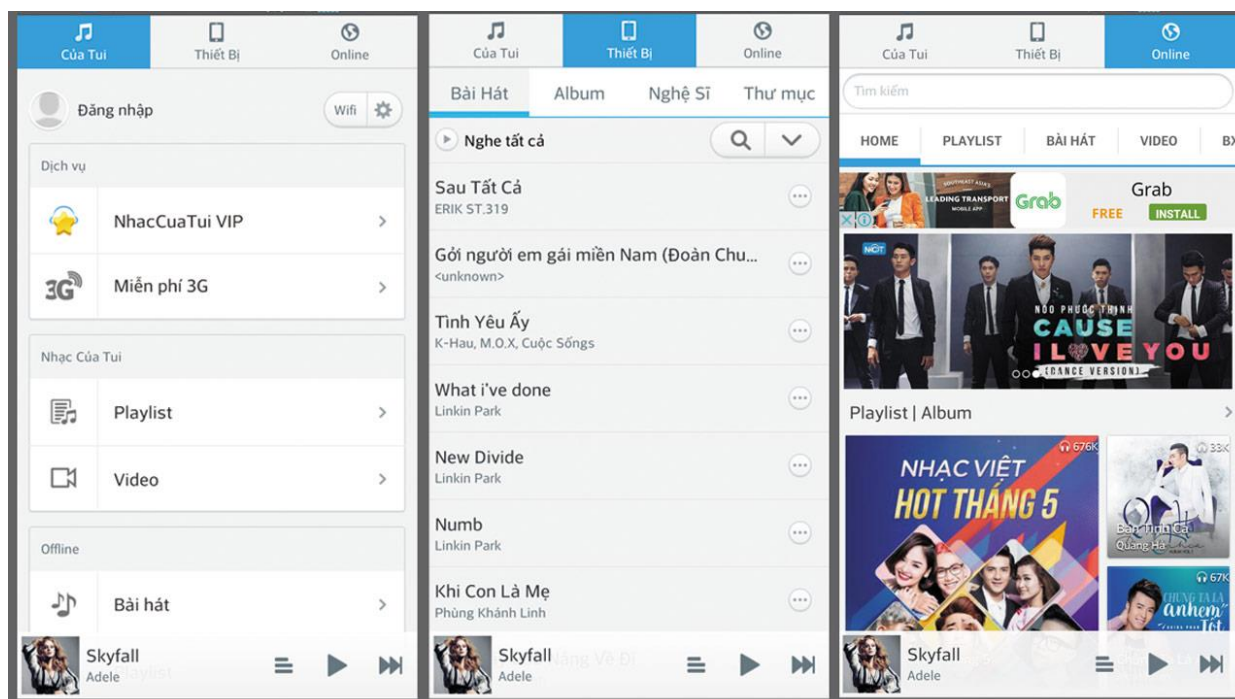
##### Khảo sát thực trạng

- Thực tế ứng dụng nghe nhạc,... đang rất phổ biến trên điện thoại di động với nhiều ứng dụng nổi tiếng như: App ZingMP3 , App NhạcCuaTui ,... những ứng dụng này hiện nay được rất nhiều người dùng sử dụng.





Hình 1.1. Giao diện ứng dụng “Zing Mp3”



Hình 1.2. Giao diện ứng dụng “Nhạc của tôi”

- Các ứng dụng này có nhiều tính năng hay như bạn có thể tạo một album các bài hát yêu thích riêng cho mình.
- Giao diện hiện đại, luôn được cập nhật để phù hợp với người dùng.

### 2.1.2 Các tính năng cơ bản của ứng dụng

Của người dùng	Của người quản trị trang web
+ Thêm thư viện riêng	+ Thêm, sửa, xóa thư viện riêng
+ Tìm kiếm bài hát	+ Thêm, sửa, xóa chủ đề
+ Nghe bài hát	+ Thêm, sửa, xóa bài hát
+ Yêu thích bài hát	+ Xem lượt yêu thích
+ Đăng nhập và chia sẻ bài hát lên Facebook.	

Hình 2.1. Các tính năng cơ bản của ứng dụng

## 2.2. PHÂN TÍCH HỆ THỐNG VỀ DỮ LIỆU

### 2.2.1 Biểu đồ use case, đặt tả use case:

#### - Biểu đồ use case (Use case diagram):

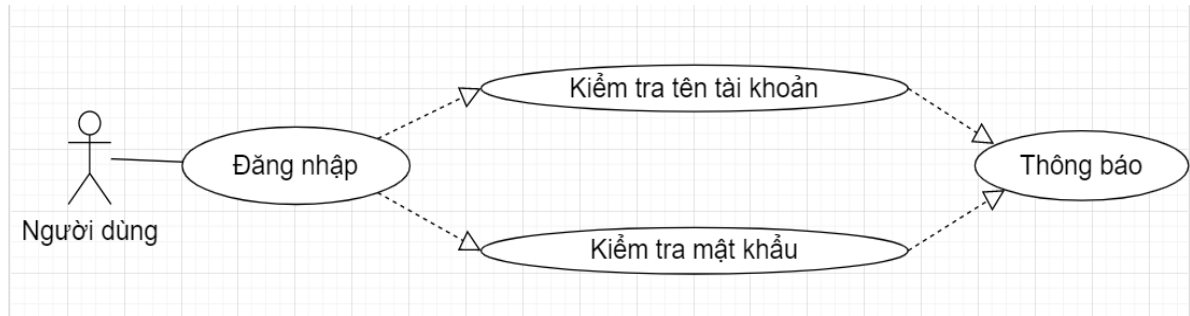
Một biểu đồ Use case chỉ ra một số lượng các tác nhân ngoại cảnh và mối liên kết của chúng đối với Use case mà hệ thống cung cấp. Một Use case là một lời miêu tả của một chức năng mà hệ thống cung cấp. Lời miêu tả Use case thường là một văn bản tài liệu, nhưng kèm theo đó cũng có thể là một biểu đồ hoạt động. Các Use case được miêu tả duy nhất theo hướng nhìn từ ngoài vào của các tác nhân (hành vi của hệ thống theo như sự mong đợi của người sử dụng), không miêu tả chức năng được cung cấp sẽ hoạt động nội bộ bên trong hệ thống ra sao. Các Use case định nghĩa các yêu cầu về mặt chức năng đối với hệ thống.



Hình 2.3. Biểu đồ Use case tổng quát

### 2.2.2 Biểu đồ Use Case

#### Use case đăng nhập quản trị:



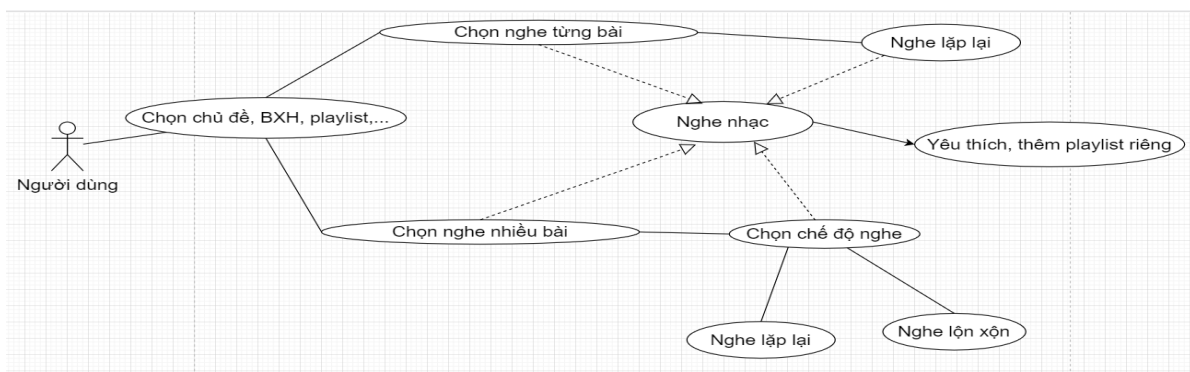
Hình 2.4. Use case đăng nhập quản trị:

- + Đối tượng sử dụng bao gồm: quản trị viên.
- + Use case này mô tả các bước đăng nhập của các người quản trị vào trang web quản trị ứng dụng.

Các bước thực hiện:

- + Hệ thống yêu cầu cung cấp thông tin đăng nhập gồm tên đăng nhập người dùng và mật khẩu.
- + Người dùng nhập xong thông tin đăng nhập và click nút đăng nhập.
- + Hệ thống check lại thông tin đăng nhập trên cơ sở dữ liệu và thông báo thành công/thất bại cho người dùng. Nếu đăng nhập thành công hệ thống dựa trên thông tin đăng nhập sẽ đồng thời phân quyền tùy theo loại người dùng (quản trị viên, người dùng).
- + Nếu đăng nhập thất bại, hệ thống sẽ hiện thông báo cho người dùng và yêu cầu đăng nhập lại.

#### Use case nghe nhạc:



Hình 2.5. Use case nghe nhạc

- + Đối tượng sử dụng: người dùng.
- + Use case này mô tả các bước nghe nhạc của người dùng trên trang ứng dụng.

Các bước thực hiện:

- + Đầu tiên, cần chọn các chủ đề, thể loại, album hay bài hát yêu thích. Khi đã chọn sản phẩm xong và nhấn nút nghe nhạc.
- + Ở màn hình nghe nhạc người dùng có thể tùy chỉnh các chế độ nghe nhạc.

#### 2.2.4.1 Đặc tả Use Case

*Bảng 2.1. Đặc tả Use Case Đăng ký tài khoản*

Use Case	Đăng kí
Tác nhân	Người dùng
Mục đích	Thiết lập tài khoản trên hệ thống cho người dùng
Điều kiện trước	Không có
Luồng sự kiện chính	<ol style="list-style-type: none"> <li>1. Người dùng chọn mục “Đăng ký”</li> <li>2. Hệ thống hiển thị form Đăng ký</li> <li>3. Người dùng nhập thông tin đăng ký, nhấn “Đăng ký”</li> <li>4. Hệ thống kiểm tra thông tin người quản lý, người dùng vừa mới đăng ký. Hệ thống sẽ hiển thị thông báo “Đăng ký thành công” và lưu nội dung vào CSDL.</li> </ol>
Luồng sự kiện phụ	Nếu các trường thông tin “Đăng ký” chưa được điền đầy đủ hoặc điền sai, hệ thống yêu cầu nhập lại. Màn hình thông báo lỗi và yêu cầu nhập lại.
Điều kiện sau	Đăng ký tài khoản thành công
Yêu cầu đặc biệt	Không có

*Bảng 2.2. Đặc tả Use Case Đăng nhập tài khoản*

Use Case	Đăng nhập
Tác nhân	Người quản trị, người dùng
Mục đích	Đăng nhập vào hệ thống để thực hiện các tác vụ
Điều kiện trước	Không có
Luồng sự kiện chính	<ol style="list-style-type: none"> <li>1. Người quản trị hay người dùng chọn mục “Đăng nhập”</li> <li>2. Hệ thống hiển thị form Đăng nhập</li> <li>3. Đăng nhập bằng tài khoản email và mật khẩu</li> <li>4. Actor điền thông tin đăng nhập rồi ấn nút “Đăng nhập”</li> <li>5. Hệ thống kiểm tra thông tin đăng nhập có tồn tại trong CSDL không. Nếu có hệ thống xác nhận đăng nhập thành công.</li> </ol>
Luồng sự kiện phụ	- Nếu Actor điền thiếu hoặc sai thông tin đăng nhập thì hệ thống gửi thông báo lỗi và yêu cầu nhập lại.
Điều kiện sau	Đăng nhập tài khoản thành công
Yêu cầu đặc biệt	Không có

*Bảng 2.3. Đặc tả Use Case Tìm kiếm bài hát*

Use Case	Tìm kiếm bài hát
Tác nhân	Người quản trị, người dùng
Mục đích	Tìm kiếm bài hát
Điều kiện trước	Không có
Luồng sự kiện chính	<ol style="list-style-type: none"> <li>1. Hiện thị giao diện người dùng</li> <li>2. Actor nhập vào thanh tìm kiếm để tìm bài hát hoặc tên ca sĩ</li> <li>3. Hệ thống tìm kiếm trong CSDL và hiển thị bài hát đã tìm kiếm hoặc liên quan</li> </ol>

	4. Nhấn vào bài hát để xem chi tiết
Luồng sự kiện phụ	- Nếu thông tin tìm kiếm không tồn tại trong CSDL thì hệ thống gửi thông báo “Không tìm thấy bài hát yêu cầu”
Yêu cầu đặc biệt	Không có

*Bảng 2.4. Đặc tả Use Case nghe nhạc*

Use Case	Nghe nhạc
Tác nhân	Người quản trị, người dùng
Mục đích	Nghe bài hát muốn nghe
Điều kiện trước	Đã chọn được bài hát muốn nghe
Luồng sự kiện chính	<ol style="list-style-type: none"> <li>1. Người dung chọn bài hát muốn nghe</li> <li>2. Hệ thống hiển thị các chức năng: <ul style="list-style-type: none"> <li>- Nghe bài hát</li> <li>- Thêm vào danh sách yêu thích hoặc Album</li> </ul> </li> <li>3. Actor chọn nghe bài hát</li> <li>4. Hệ thống hiển thị form chế độ nghe gồm: <ul style="list-style-type: none"> <li>- Nghe một lần</li> <li>- Nghe lặp lại</li> <li>- Nghe lộn xộn</li> </ul> </li> <li>5. Actor chọn chế độ nghe</li> <li>6. Hệ thống tìm kiếm trong CSDL và phát bài nhạc theo yêu cầu mà Actor đã chọn</li> </ol>
Luồng sự kiện phụ	- Nếu bài nhạc đã bị xóa khỏi CSDL thì hệ thống sẽ gửi thông báo “Không thể phát bài hát này”
Điều kiện sau	Nếu không có lỗi xảy ra thì người dung sẽ dc nghe bài hát đã chỉ định theo chế độ nghe đã chọn
Yêu cầu đặc biệt	Không có

*Bảng 2.5. Đặc tả Use Case chọn Album và bài hát yêu thích*

Use Case	Chọn Album và bài hát yêu thích
Tác nhân	Người quản trị, người dùng
Mục đích	Chọn nhưng bài hát mà mình muốn nghe
Điều kiện trước	Không có
Luồng sự kiện chính	<ol style="list-style-type: none"> <li>1. Actor chọn chức năng tạo Album.</li> <li>2. Hệ thống sẽ hiển thị một form gồm: <ul style="list-style-type: none"> <li>- Thêm bài hát vào Album đã tạo</li> <li>- Thêm Album mới</li> <li>- Trở về</li> </ul> </li> <li>3. Actor chọn chức năng thêm bài hát vào Album đã tạo</li> <li>4. Hệ thống hiển thị danh sách bài hát đã có trong album, hiển thị chức năng tìm kiếm</li> <li>5. Actor tìm kiếm bài hát hoặc ca sĩ</li> <li>6. Hệ thống hiển thị danh sách bài hát có liên quan</li> <li>7. Người dung chọn bài hát để thêm vào album</li> </ol>
Luồng sự kiện phụ	- Nếu người dung chọn thêm Album mới, hệ thống sẽ hiển thị form đặt tên cho album. Người dung đặt tên xong thì quay lại bước 2
Điều kiện sau	Đăng nhập tài khoản thành công
Yêu cầu đặc biệt	Không có

*Bảng 2.6. Đặc tả Use Case xem các quảng cáo bài hát chủ đề*

<b>Use Case</b>	<b>xem các quảng cáo bài hát chủ đề</b>
Tác nhân	Người quản trị, người dùng
Mục đích	Tìm hiểu các bài hát chủ đề
Điều kiện trước	Không có
Luồng sự kiện chính	<ol style="list-style-type: none"> <li>1. Hệ thống hiển thị các chủ đề</li> <li>2. Actor chọn chủ đề</li> <li>3. Hệ thống kiểm tra CSDL và đưa ra danh sách các bài hát liên quan</li> </ol>
Luồng sự kiện phụ	- Nếu chủ đề bị xóa khỏi csdl thì sẽ thông báo lỗi
Điều kiện sau	Đăng nhập tài khoản thành công
Yêu cầu đặc biệt	Không có



*Bảng 2.7. Đặc tả Use Case Quên mật khẩu*

<b>Ca sử dụng</b>	<b>Quên mật khẩu</b>
Tác nhân	Người quản trị, người dùng
Mục đích	Lấy lại mật khẩu khi quên
Điều kiện trước	Actor đã đăng ký tài khoản
Luồng sự kiện chính	<ol style="list-style-type: none"> <li>1. Chọn chức năng “Quên mật khẩu”</li> <li>2. Hệ thống hiển thị form nhập email</li> <li>3. Actor nhập email</li> <li>4. Hệ thống kiểm tra và gửi mã OTP đến email nếu email tồn tại đồng thời gửi form nhập mã OTP</li> <li>5. Actor nhập mã OTP</li> <li>6. Hệ thống kiểm tra CSDL và trả về form thay đổi mật khẩu nếu mã OTP trùng khớp</li> <li>7. Actor nhập thông tin thay đổi</li> <li>8. Hệ thống kiểm tra giá trị nhập vào, mã hoá và lưu vào CSDL đồng thời hiển thị thông báo “Thay đổi mật khẩu thành công”</li> </ol>
Luồng sự kiện phụ	<ul style="list-style-type: none"> <li>- Nhập sai email thì thông báo không tồn tại email</li> <li>- Nhập sai mã OTP thông báo mã OTP không hợp lệ</li> <li>- Không nhập thông tin thông báo yêu cầu nhập thông tin</li> <li>- Không nhập mật khẩu mới thì sẽ mật khẩu sẽ không được cập nhập vào cơ sở dữ liệu.</li> </ul>
Điều kiện sau	Nếu không có lỗi xảy ra thì mật khẩu mới được lưu vào CSDL, người dùng có thể đăng nhập với mật khẩu mới
Yêu cầu đặc biệt	Không có

*Bảng 2.8. Đặc tả Use Case quản lý CSDL*

Use Case	Quản lý csdl
Tác nhân	Người quản trị
Mục đích	Quản lý csdl
Điều kiện trước	Đăng nhập thành công vào hệ thống
Luồng sự kiện chính	<ol style="list-style-type: none"> <li>1. Chọn CSDL cần quản lý</li> <li>2. Hệ thống hiển thị form quản lý để người quản trị thao tác: <ul style="list-style-type: none"> <li>- Thêm CSDL</li> <li>- Sửa CSDL</li> <li>- Xóa CSDL</li> </ul> </li> <li>3. Người quản trị chọn chức năng để quản lý</li> <li>4. Hệ thống kiểm tra thông tin mới cập nhật và lưu vào hệ thống</li> </ol>
Luồng sự kiện phụ	- Nếu cập nhật thiếu hoặc sai thông tin, hệ thống sẽ báo lỗi và yêu cầu cập nhật lại
Điều kiện sau	Không có lỗi xảy ra thì người quản trị đã thay đổi CSDL thành công
Yêu cầu đặc biệt	Không có

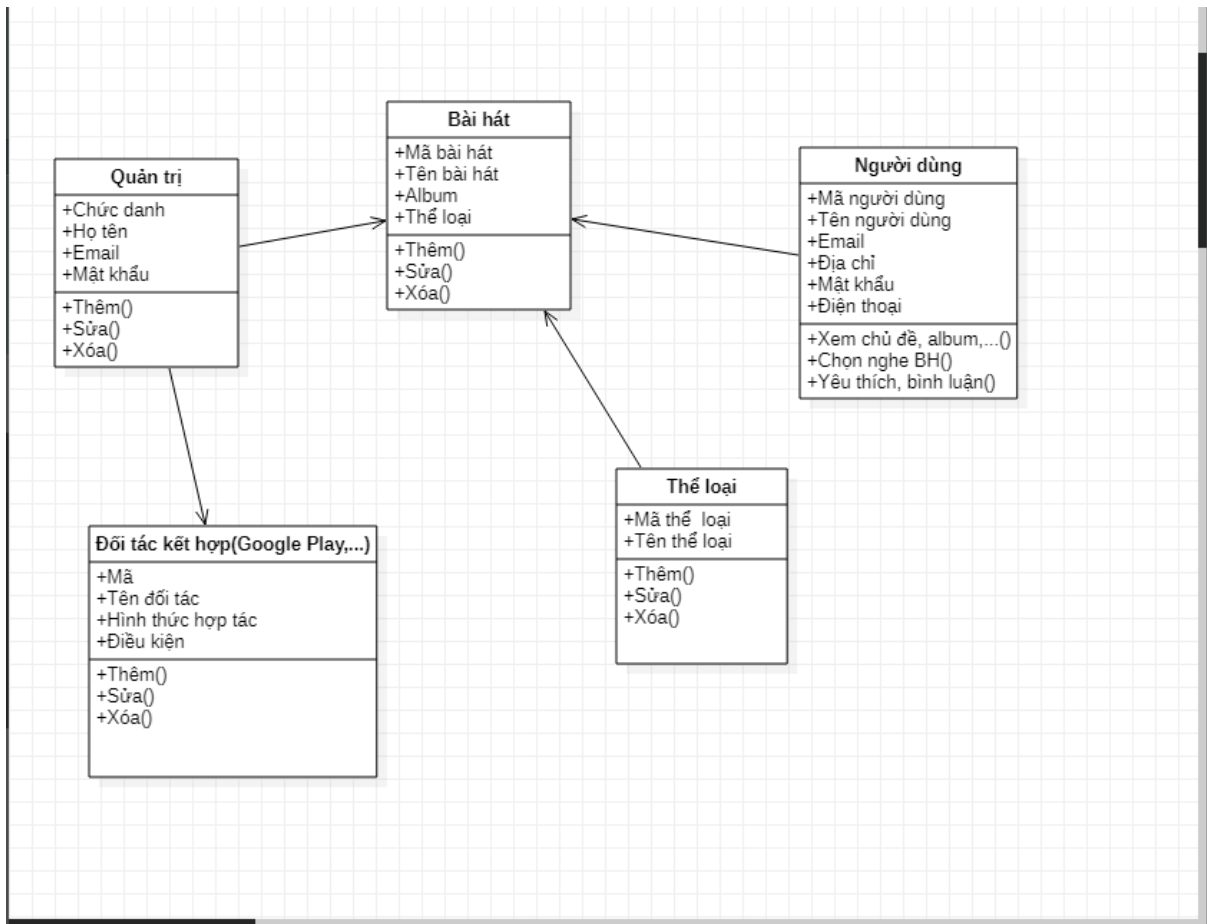
#### **2.2.4 Xây dựng biểu đồ lớp (Class diagram)**

##### **- Xác định các lớp từ use case:**

Xét từ bài toán quản lý bán hàng và use case chi tiết đã xây dựng ở trên chúng ta có thể liệt kê các lớp cơ bản như bài hát, quản trị, người dùng, thể loại, đối tác kết hợp.

##### **- Xác định các thuộc tính và một số phương thức cơ bản:**

Từ các lớp xác định ở bước trên ta có thể thấy một số thuộc tính và phương thức cơ bản như trong Quản trị thì có thuộc tính chức danh, họ tên, email, mật khẩu hoặc Lớp bài hát có thuộc tính: mã bài hát, tên bài hát, album, thể loại. Cụ thể như hình sau:



Hình 2.6. Biểu đồ lớp

### 2.2.5 Biểu đồ tuần tự

Mục đích: biểu diễn tương tác giữa những người dùng và những đối tượng bên trong hệ thống. Biểu đồ này cho biết các thông điệp được truyền tuần tự như thế nào theo thời gian. Thứ tự các sự kiện trong biểu đồ tuần tự hoàn toàn tương tự như trong scenario mô tả use case tương ứng.

Biểu diễn: Biểu đồ tuần tự được biểu diễn bởi các đối tượng và message truyền đi giữa các đối tượng đó.

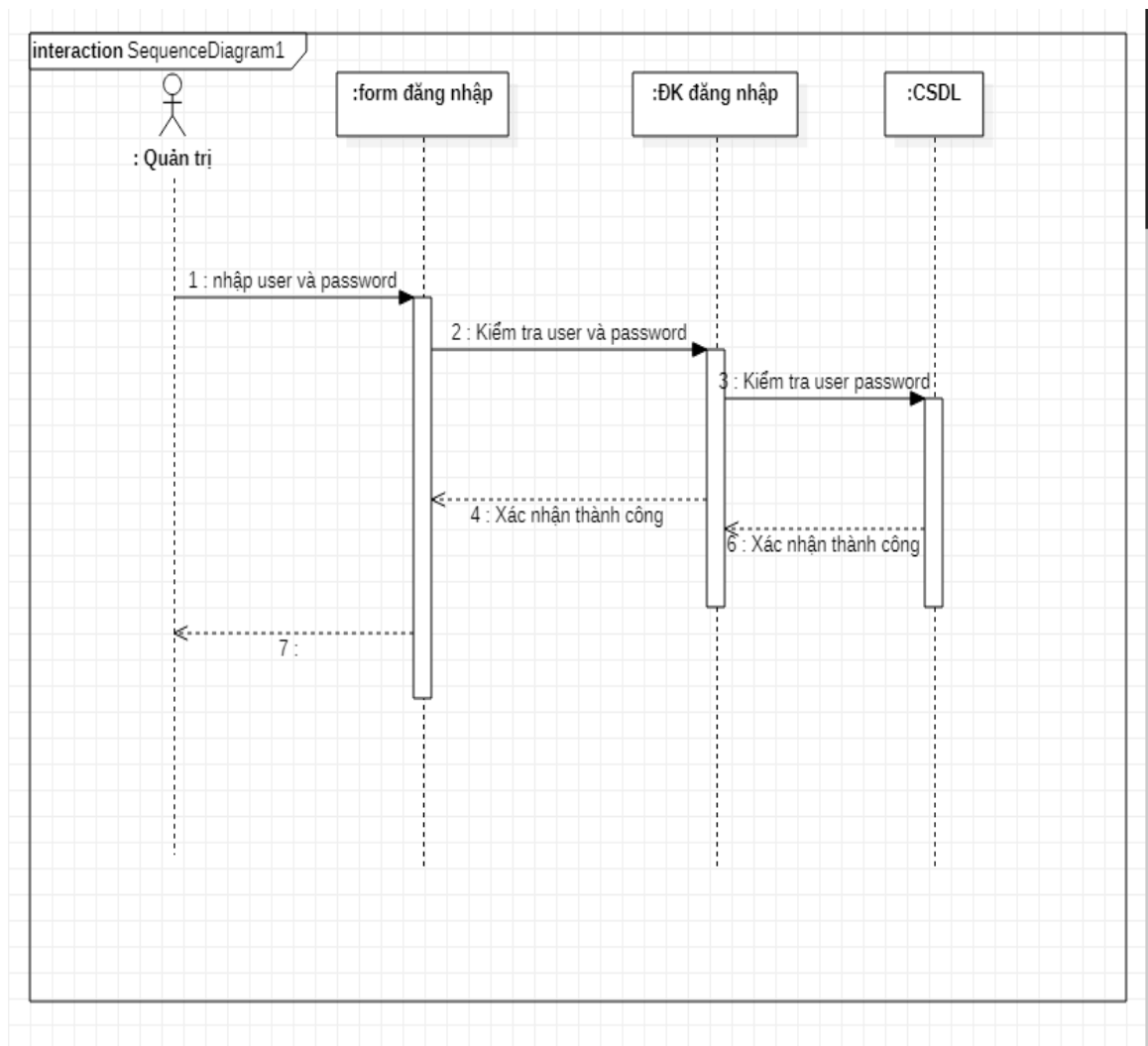
- Trong hệ thống quản lý bán hàng, chúng ta lựa chọn biểu đồ tương tác dạng tuần tự để biểu diễn các tương tác giữa các đối tượng. Để xác định rõ các thành phần cần bổ sung trong biểu đồ lớp, trong mỗi biểu đồ tuần tự của hệ thống quản lý bài hát sẽ thực hiện:

+ Xác định rõ kiểu của đối tượng tham gia trong tương tác (ví dụ giao diện, điều khiển hay thực thể).

- + Mỗi biểu đồ tuần tự có thể có ít nhất một lớp giao diện (Form) tương ứng với chức năng (use case) mà biểu đồ đó mô tả
- + Mỗi biểu đồ tuần tự có thể liên quan đến một hoặc nhiều đối tượng thực thể.

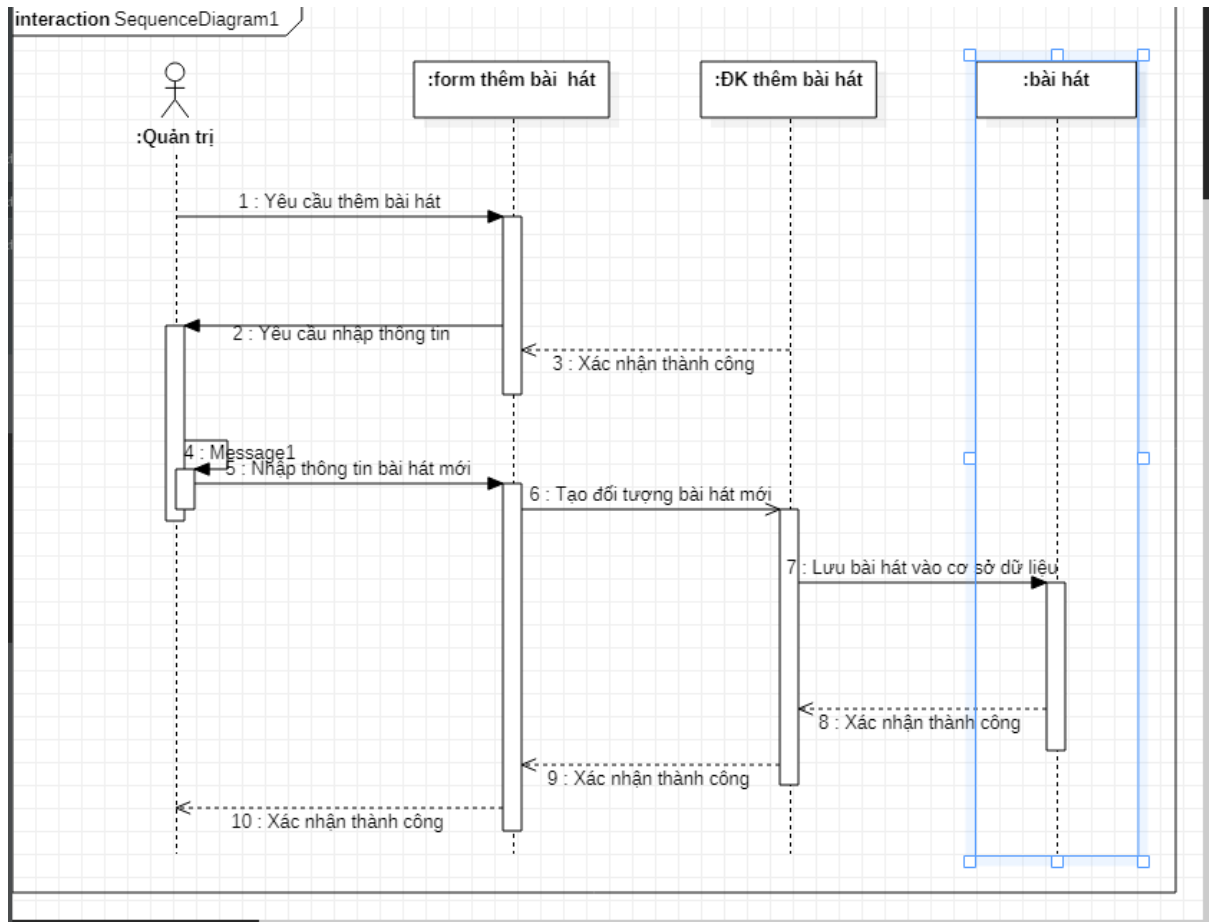
Các đối tượng thực thể chính là các đối tượng của các lớp đã được xây dựng trong biểu đồ thiết kế chi tiết. Dưới đây là một số biểu đồ tuần tự cho các chức năng của ứng dụng nghe nhạc:

. **Đăng nhập:** Các đối tượng tham gia gồm User, Form Đăng nhập, đối tượng điều khiển UserController, CSDL



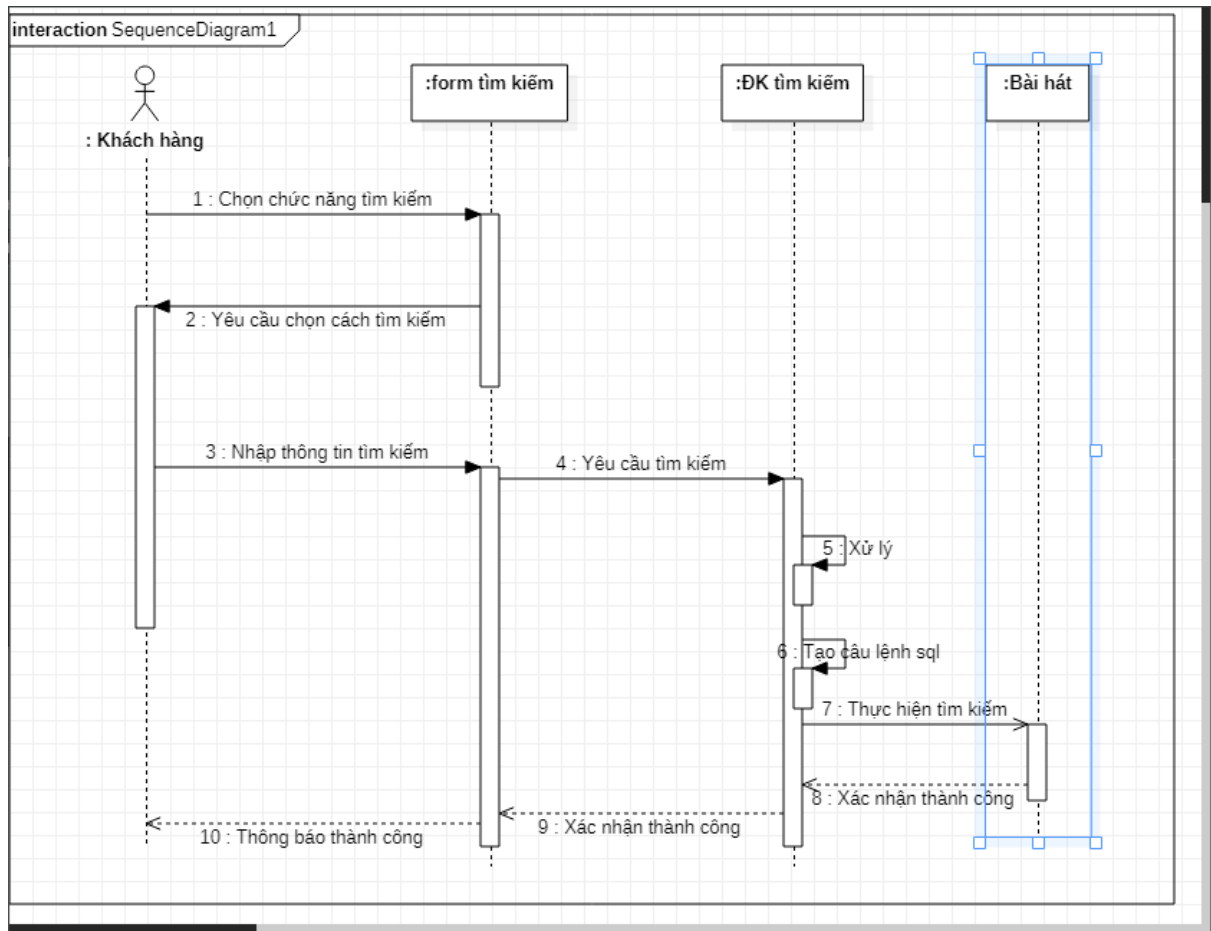
Hình 2.7. Biểu đồ tuần tự của chức năng đăng nhập

. **Chức năng thêm bài hát:** các đối tượng tham gia gồm Quản trị viên, Form thêm bài hát, đối tượng điều khiển User Controller thêm bài hát, CSDL bài hát



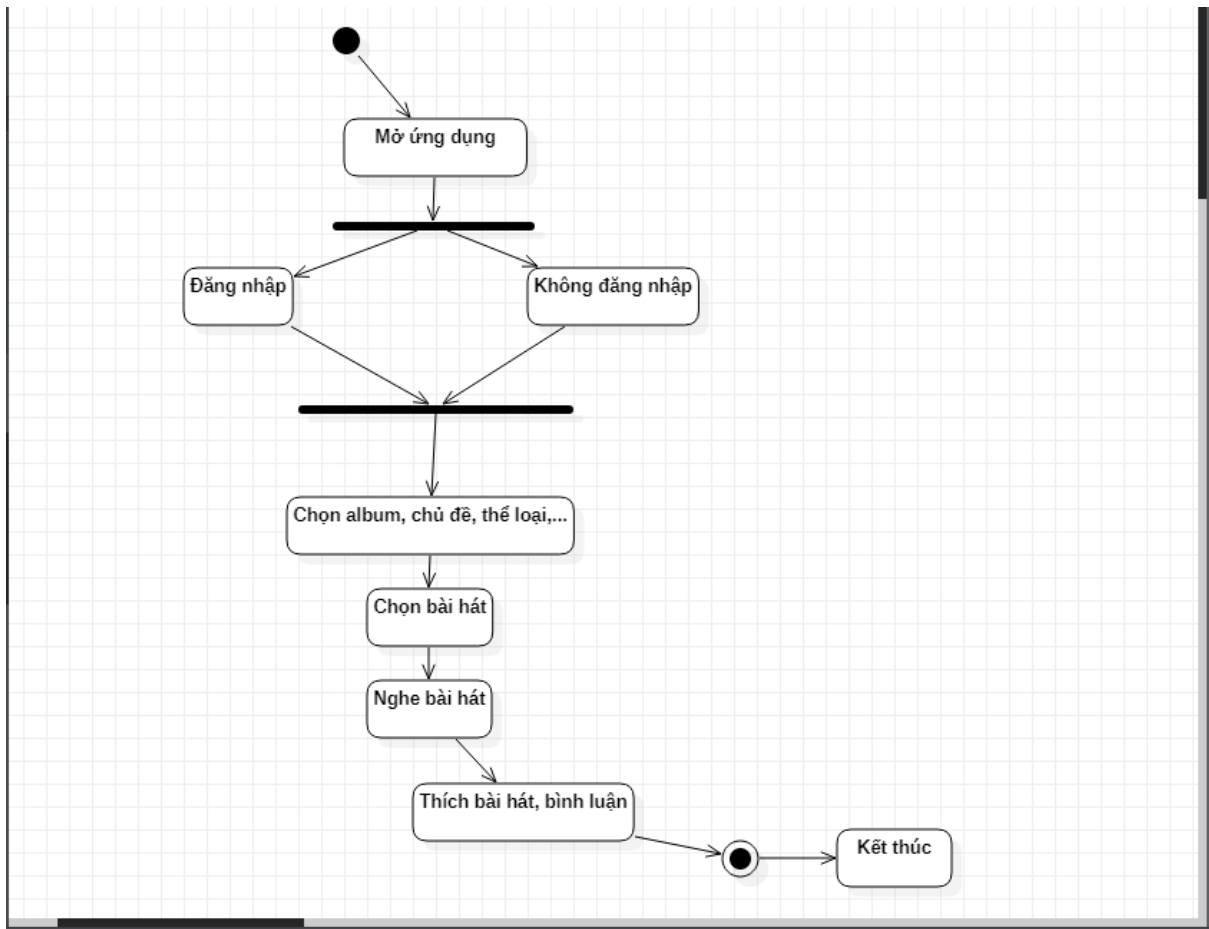
Hình 2.8. Biểu đồ tuần tự của chức năng thêm bài hát

. **chức năng tìm kiếm bài hát:** đối tượng tham gia gồm : User, Form tìm kiếm, đối tượng điều khiển user controller tìm kiếm, csdl bài hát



Hình 2.9. Biểu đồ tuần tự của chức năng tìm kiếm bài hát

### 2.2.5 Biểu đồ hoạt động của người dùng:



Hình 2.10. Biểu đồ hoạt động của người dùng

Giải thuật của hành động của người dùng:

- Bước 1: Người dùng mở ứng dụng
- Bước 2: Người dùng có thể chọn chức năng đăng nhập/hoặc không
- Bước 3: Người dùng chọn Album, thể loại, chủ đề âm nhạc
- Bước 4: Người dùng chọn bài hát muốn nghe
- Bước 5: Người dùng nghe bài hát
- Bước 6: Người dùng like, đưa phản hồi về bài hát

## 2.3. THIẾT KẾ VỀ CƠ SỞ DỮ LIỆU

### 2.3.1 Các bảng của cơ sở dữ liệu:

**Bảng người dùng**

#	Tên	Kiểu	Bảng mã đối chiếu	Thuộc tính	Null	Mặc định	Ghi chú	Thêm	Hành động
1	<b>UserName</b>	varchar(255)	utf8_unicode_ci		Không	Không			Thay đổi  Xóa  Thêm
2	<b>Password</b>	varchar(255)	utf8_unicode_ci		Không	Không			Thay đổi  Xóa  Thêm
3	<b>Name</b>	varchar(255)	utf8_unicode_ci		Không	Không			Thay đổi  Xóa  Thêm
4	<b>Email</b>	varchar(255)	utf8_unicode_ci		Không	Không			Thay đổi  Xóa  Thêm
5	<b>Image</b>	varchar(255)	utf8_unicode_ci		Không	Không			Thay đổi  Xóa  Thêm

*Hình 2.11. Bảng người dùng*

Bảng này lưu trữ các thông tin đăng nhập của người dùng. Bảng Người dùng được xây dựng với các trường:

- + **UserName**: trường UserName này của bảng Người dùng được dùng để lưu trữ tên đăng nhập và làm khóa chính cho bảng.
- + **Password**: trường Password lưu trữ mật khẩu đăng nhập.
- + **Name**: trường Name lưu trữ tên của người dùng .
- + **Email**: lưu trữ email của người dùng.
- + **Image**: lưu trữ ảnh đại diện của người dùng.

**Bảng Bài hát**

#	Tên	Kiểu	Bảng mã đối chiếu	Thuộc tính	Null	Mặc định	Ghi chú	Thêm	Hành động
1	<b>IdBaiHat</b>	int(11)			Không	Không		AUTO_INCREMENT	Thay đổi  Xóa  Thêm
2	<b>TenBaiHat</b>	varchar(255)	utf8_unicode_ci		Không	Không			Thay đổi  Xóa  Thêm
3	<b>TenCaSi</b>	varchar(255)	utf8_unicode_ci		Không	Không			Thay đổi  Xóa  Thêm
4	<b>HinhBaiHat</b>	varchar(255)	utf8_unicode_ci		Không	Không			Thay đổi  Xóa  Thêm
5	<b>LinkBaiHat</b>	varchar(255)	utf8_unicode_ci		Không	Không			Thay đổi  Xóa  Thêm
6	<b>IdBangXepHang</b>	varchar(255)	utf8_unicode_ci		Không	Không			Thay đổi  Xóa  Thêm
7	<b>IdChuDe</b>	varchar(255)	utf8_unicode_ci		Không	Không			Thay đổi  Xóa  Thêm
8	<b>IdNgheSi</b>	varchar(255)	utf8_unicode_ci		Không	Không			Thay đổi  Xóa  Thêm
9	<b>IdPhoBien</b>	varchar(255)	utf8_unicode_ci		Không	Không			Thay đổi  Xóa  Thêm
10	<b>IdPlaylist</b>	varchar(255)	utf8_unicode_ci		Không	Không			Thay đổi  Xóa  Thêm
11	<b>LuotThich</b>	int(11)			Không	Không			Thay đổi  Xóa  Thêm
12	<b>IdThinhHanh</b>	varchar(255)	utf8_unicode_ci		Không	Không			Thay đổi  Xóa  Thêm

*Hình 2.12. Bảng Bài hát*



Trong cơ sở dữ liệu của một ứng dụng nghe nhạc, đây là một trong những bảng quan trọng nhất. Bảng này chứa dữ liệu các bài hát của ứng dụng, từ đó các chủ đề, bảng xếp hạng hay playlist có thể liên kết với bảng baihat này để đưa ra được các bài hát thuộc thể bảng xếp hạng nào, chủ đề nào, hay playlist nào. Trong bảng này sẽ có đầy đủ các thông tin của bài hát:

- + IdBaiHat: được sử dụng làm khoá chính.
- + TenBaiHat: lưu trữ tên bài hát
- + TenCaSi: lưu trữ tên ca sĩ ứng với bài hát
- + HinhBaiHat: chứa đường link hình ảnh đúng với tên bài hát.
- + LinkBaiHat: chứa đường link của bài hát.
- + IdBangXepHang: là khoá ngoại liên kết 2 bảng bangxephang và baihat.
- + IdChuDe: là khoá ngoại liên kết 2 bảng chude và baihat.
- + IdNgheSi: là khoá ngoại liên kết 2 bảng nghesi và baihat.
- + IdPhoBien: là khoá ngoại liên kết 2 bảng phobien và baihat.
- + IdPlayList: là khoá ngoại liên kết 2 bảng playlist và baihat.
- + LuotThich: là trường lưu số lượt thích của mỗi bài hát.
- + IdThinhHanh: là khoá ngoại liên kết 2 bảng tinhhanh và baihat.

### Bảng chủ đề

#	Tên	Kiểu	Bảng mã đối chiếu	Thuộc tính	Null	Mặc định	Ghi chú	Thêm	Hành động
<input type="checkbox"/> 1	IdChuDe	int(11)			Không	Không		AUTO_INCREMENT	Thay đổi  Xóa  Thêm
<input type="checkbox"/> 2	TenChuDe	varchar(255)	utf8_unicode_ci		Không	Không			Thay đổi  Xóa  Thêm
<input type="checkbox"/> 3	HinhChuDe	varchar(255)	utf8_unicode_ci		Không	Không			Thay đổi  Xóa  Thêm

☐ Theo dõi bảng Lưu mục đã chọn Duyệt Thay đổi Xóa Chính Duy nhất Chỉ mục Spatial 1

Hình 2.13. Bảng chủ đề

Bảng chude lưu trữ thông tin của các chủ đề khác nhau. Trong bảng gồm các trường:

- + IdChuDe: được sử dụng làm khóa chính cho bảng chude và khóa ngoại trong bảng baihat.

- + TenChude: lưu trữ tên một chủ đề trong âm nhạc, ví dụ như: chủ đề nhạc Vpop, Kpop,...
- + HinhChude: chứa đường link hình ảnh đúng với tên chủ đề.

### Bảng xếp hạng

#	Tên	Kiểu	Bảng mã đối chiếu	Thuộc tính	Null	Mặc định	Ghi chú	Thêm	Hành động
1	IdBangXepHang	int(11)			Không	Không		AUTO_INCREMENT	Thay đổi  Xóa  Thêm
2	TenBangXepHang	varchar(255)	utf8_unicode_ci		Không	Không			Thay đổi  Xóa  Thêm
3	HinhBangXepHang	varchar(255)	utf8_unicode_ci		Không	Không			Thay đổi  Xóa  Thêm

Hình 2.14. Bảng xếp hạng

Bảng bangxephang lưu trữ thông tin của bảng xếp hạng. Trong bảng gồm các trường:

- + IdBangXepHang: trường này là khóa chính của bảng bangxephang
- + TenBangXepHang: lưu trữ tên của bảng xếp hạng.
- + HinhBangXepHang: chứa đường link hình ảnh đúng với tên bảng xếp hạng.

### Bảng Bài hát thư viện Playlist

#	Tên	Kiểu	Bảng mã đối chiếu	Thuộc tính	Null	Mặc định	Ghi chú	Thêm	Hành động
1	IdBaiHatThuVienPlayList	int(11)			Không	Không		AUTO_INCREMENT	Thay đổi  Xóa  Thêm
2	IdThuVienPlayList	int(11)			Không	Không			Thay đổi  Xóa  Thêm
3	IdBaiHat	int(11)			Không	Không			Thay đổi  Xóa  Thêm
4	TenBaiHat	varchar(255)	utf8_unicode_ci		Không	Không			Thay đổi  Xóa  Thêm
5	TenCaSi	varchar(255)	utf8_unicode_ci		Không	Không			Thay đổi  Xóa  Thêm
6	HinhBaiHat	varchar(255)	utf8_unicode_ci		Không	Không			Thay đổi  Xóa  Thêm
7	LinkBaiHat	varchar(255)	utf8_unicode_ci		Không	Không			Thay đổi  Xóa  Thêm

Hình 2.15. Bảng Bài hát thư viện Playlist

Bảng baihatthuvienplaylist lưu trữ các bài hát của mỗi người dùng khi tạo playlist riêng. Bảng baihatthuvienplaylist được xây dựng bởi các trường:

- + IdBaiHatThuVienPlayList: được sử dụng làm khóa chính cho bảng baihatthuvienplaylist.
- + IdThuVienPlayList: là khoá ngoại liên kết 2 bảng thuvienplaylist và baihatthuvienplaylist.

- + IdBaiHat: là khoá ngoại liên kết 2 bảng baihat và baihatthuvienplaylist.
- + TenBaiHat: lưu trữ tên bài hát.
- + TenCaSi: lưu trữ tên ca sĩ.
- + HinhBaiHat: chứa đường link hình ảnh đúng với tên bài hát.
- + LinkBaiHat: chứa đường link của bài hát.

### Bảng Nghệ sĩ

#	Tên	Kiểu	Bảng mã đối chiếu	Thuộc tính	Null	Mặc định	Ghi chú	Thêm	Hành động
1	IdNgheSi	int(11)			Không	Không		AUTO_INCREMENT	Thay đổi  Xóa  Thêm
2	TenNgheSi	varchar(255)	utf8_unicode_ci		Không	Không			Thay đổi  Xóa  Thêm
3	HinhNgheSi	varchar(255)	utf8_unicode_ci		Không	Không			Thay đổi  Xóa  Thêm

Hình 2.16. Bảng Nghệ sĩ

Bảng nghesi này lưu thông tin các nghệ sĩ. Trong bảng nghesi gồm các trường:

- + IdNgheSi: trường IdNgheSi là khóa chính của bảng nghesi và là khoá ngoại của bảng baihat.
- + TenNgheSi: lưu trữ tên của nghệ sĩ.
- + HinhNgheSi: chứa đường link hình ảnh đúng với tên nghệ sĩ.

### Bảng phổ biến

#	Tên	Kiểu	Bảng mã đối chiếu	Thuộc tính	Null	Mặc định	Ghi chú	Thêm	Hành động
1	IdPhoBien	int(11)			Không	Không		AUTO_INCREMENT	Thay đổi  Xóa  Thêm
2	TenPhoBien	varchar(255)	utf8_unicode_ci		Không	Không			Thay đổi  Xóa  Thêm
3	HinhPhoBien	varchar(255)	utf8_unicode_ci		Không	Không			Thay đổi  Xóa  Thêm

Hình 2.17. Bảng phổ biến

Bảng phobien lưu thông tin các loại phổ biến. Trong bảng gồm các trường:

- + IdPhoBien: trường IdPhoBien là khóa chính của bảng phobien và là khoá ngoại của bảng baihat.
- + TenPhoBien: lưu tên loại phổ biến.
- + HinhPhoBien: chứa đường link hình ảnh đúng với tên của loại phổ biến.

## Bảng Người dùng yêu thích

#	Tên	Kiểu	Bảng mã đối chiếu	Thuộc tính	Null	Mặc định	Ghi chú	Thêm	Hành động
1	<b>IdYeuThich</b>	int(11)			Không	Không		AUTO_INCREMENT	Thay đổi  Xóa  Thêm
2	<b>UserName</b>	varchar(255)	utf8_unicode_ci		Không	Không			Thay đổi  Xóa  Thêm
3	<b>IdBaiHat</b>	int(11)			Không	Không			Thay đổi  Xóa  Thêm
4	<b>TenBaiHat</b>	varchar(255)	utf8_unicode_ci		Không	Không			Thay đổi  Xóa  Thêm
5	<b>TenCaSi</b>	varchar(255)	utf8_unicode_ci		Không	Không			Thay đổi  Xóa  Thêm
6	<b>HinhBaiHat</b>	varchar(255)	utf8_unicode_ci		Không	Không			Thay đổi  Xóa  Thêm
7	<b>LinkBaiHat</b>	varchar(255)	utf8_unicode_ci		Không	Không			Thay đổi  Xóa  Thêm

Hình 2.18. Bảng Người dùng yêu thích

Bảng nguoidungyeuthich chứa thông tin bài hát mà người dùng đã yêu thích. Trong bảng gồm các trường:

- + **IdYeuThich**: trường này là khoá chính của bảng nguoidungyeuthich.
- + **UserName**: là khoá ngoại liên kết 2 bảng nguoidung và nguoidungyeuthich.
- + **IdBaiHat**: là khoá ngoại liên kết 2 bảng baihay và nguoidungyeuthich.
- + **TenBaiHat**: lưu trữ tên bài hát.
- + **TenCaSi**: lưu trữ tên ca sĩ ứng với mỗi bài hát.
- + **HinhBaiHat**: chứa đường link hình ảnh đúng với tên của bài hát.
- + **LinkBaiHat**: chứa đường link của bài hát.

## Bảng Playlist

#	Tên	Kiểu	Bảng mã đối chiếu	Thuộc tính	Null	Mặc định	Ghi chú	Thêm	Hành động
1	<b>IdPlaylist</b>	int(11)			Không	Không		AUTO_INCREMENT	Thay đổi  Xóa  Thêm
2	<b>TenPlaylist</b>	varchar(255)	utf8_unicode_ci		Không	Không			Thay đổi  Xóa  Thêm
3	<b>HinhNen</b>	varchar(255)	utf8_unicode_ci		Không	Không			Thay đổi  Xóa  Thêm

Hình 2.19. Bảng Playlist

Bảng playlist chứa thông tin các playlist. Trong bảng gồm các trường:

- + **IdPlaylist**: trường này là khoá chính của bảng playlist và là khoá ngoại bảng baihat.
- + **TenPlaylist**: lưu trữ tên của mục playlist.
- + **HinhNen**: chứa đường link hình ảnh đúng với tên của mỗi playlist.

## Bảng Thỉnh hành

#	Tên	Kiểu	Bảng mã đối chiếu	Thuộc tính	Null	Mặc định	Ghi chú	Thêm	Hành động
1	IdThinhHanh	int(11)			Không	Không		AUTO_INCREMENT	Thay đổi  Xóa  Thêm
2	TenThinhHanh	varchar(255)	utf8_unicode_ci		Không	Không			Thay đổi  Xóa  Thêm
3	HinhThinhHanh	varchar(255)	utf8_unicode_ci		Không	Không			Thay đổi  Xóa  Thêm

Hình 2.20. Bảng Thỉnh hành

Bảng thínhhanh chứa thông tin các mục thỉnh hành. Trong bảng gồm các trường:

- + IdThinhHanh: trường này là khoá chính của bảng thínhhanh và là khoá ngoại bảng baihat.
- + TenThinhHanh: lưu trữ tên của mục thỉnh hành.
- + HinhNen: chứa đường link hình ảnh đúng với tên của mỗi mục thỉnh hành.

## Bảng Thư viện Playlist

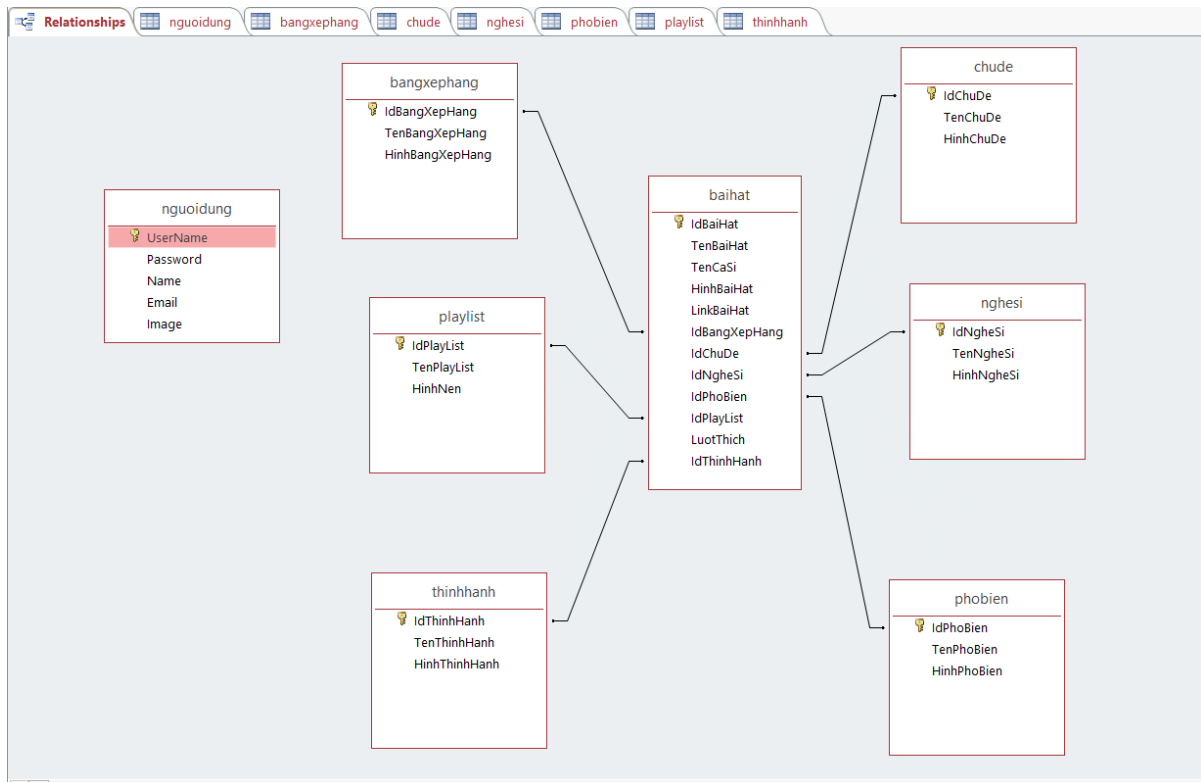
#	Tên	Kiểu	Bảng mã đối chiếu	Thuộc tính	Null	Mặc định	Ghi chú	Thêm	Hành động
1	IDThuVienPlayList	int(11)			Không	Không		AUTO_INCREMENT	Thay đổi  Xóa  Thêm
2	TenThuVienPlayList	varchar(255)	utf8_unicode_ci		Không	Không			Thay đổi  Xóa  Thêm
3	HinhThuVienPlayList	varchar(255)	utf8_unicode_ci		Không	Không			Thay đổi  Xóa  Thêm
4	UserName	varchar(255)	utf8_unicode_ci		Không	Không			Thay đổi  Xóa  Thêm

Hình 2.21. Bảng Thư viện Playlist

Bảng thuvienplaylist chứa thông tin playlist riêng của mỗi người dùng. Trong bảng này có các trường:

- + IDThuVienPlayList: trường này là khoá chính của bảng thuvienplaylist
- + TenThuVienPlayList: lưu trữ tên thư viện playlist của người dùng
- + HinhThuVienPlayList: chứa đường link hình ảnh ứng với mỗi thư viện playlist của người dùng.
- + UserName: là khoá ngoại liên kết 2 bảng nguoidung và thuvienplaylist.

### 2.3.2 Môi quan hệ giữa các bảng trong cơ sở dữ liệu:

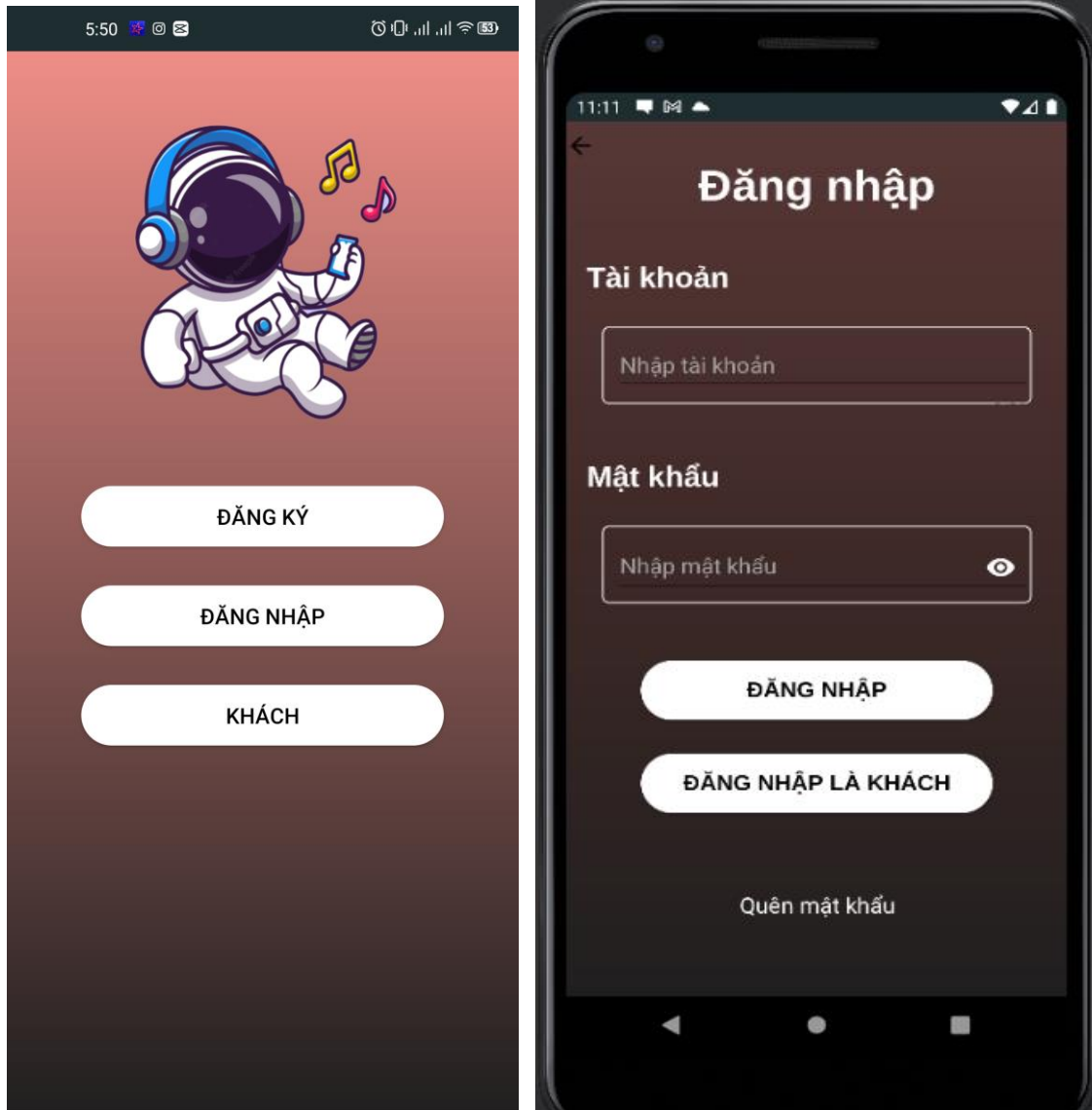


Hình 2.22. Môi quan hệ giữa các bảng trong cơ sở dữ liệu

## CHƯƠNG III. KẾT QUẢ THỰC NGHIỆM

### 3.1. GIAO DIỆN NGƯỜI DÙNG

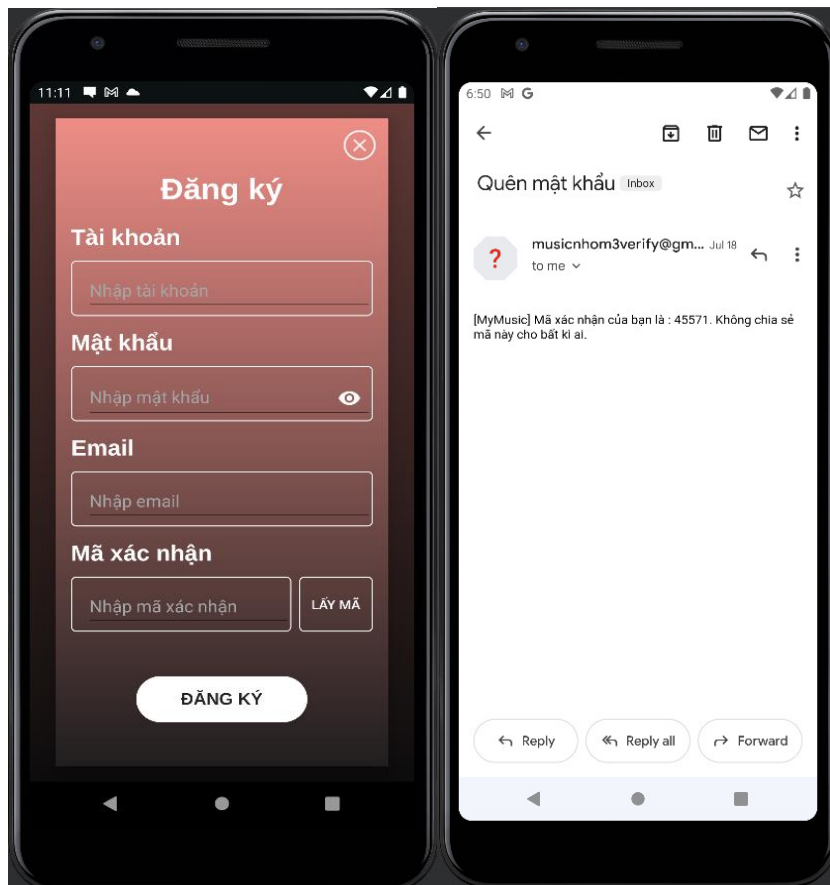
#### 3.1.1 Giao diện màn hình đăng nhập



Hình 3.1. Giao diện màn hình đăng nhập

Khi mở ứng dụng lần đầu tiên sẽ xuất hiện màn hình đăng nhập. Cho phép người dùng đăng nhập với người dùng đã có tài khoản. Người dùng đăng nhập thông tin tài khoản và mật khẩu. Hệ thống kiểm tra và cấp quyền theo từng loại tài khoản.

### 3.1.2 Giao diện màn hình đăng ký



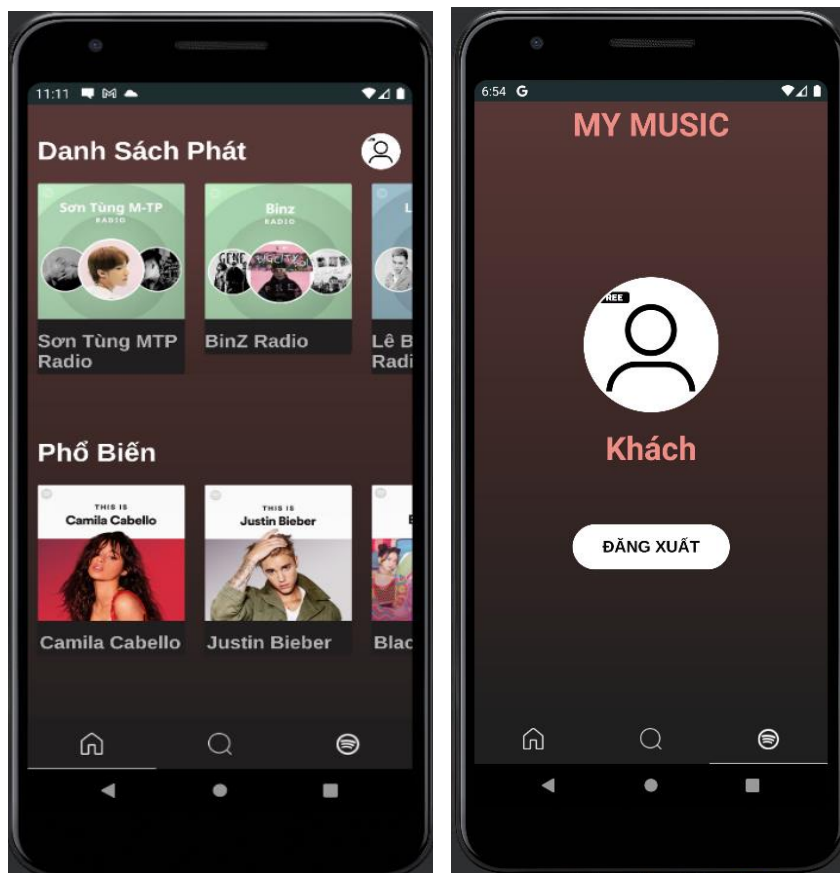
Hình 3.2. Giao diện màn hình đăng ký

Người dùng tạo tài khoản và xác minh tài khoản bằng email cá nhân. Sau khi nhập đủ thông tin hợp lệ hệ thống gửi mã xác nhận qua email. Người dùng nhập lại mã vào ô tương ứng và tạo thành công tài khoản



### 3.1.3 Giao diện màn hình chính ứng dụng

- a. Khi đăng nhập với tư cách “Khách” (không có trang thư viện của tôi)



Hình 3.3. Khi đăng nhập với tư cách “Khách”

Hồ sơ người dùng là khách.

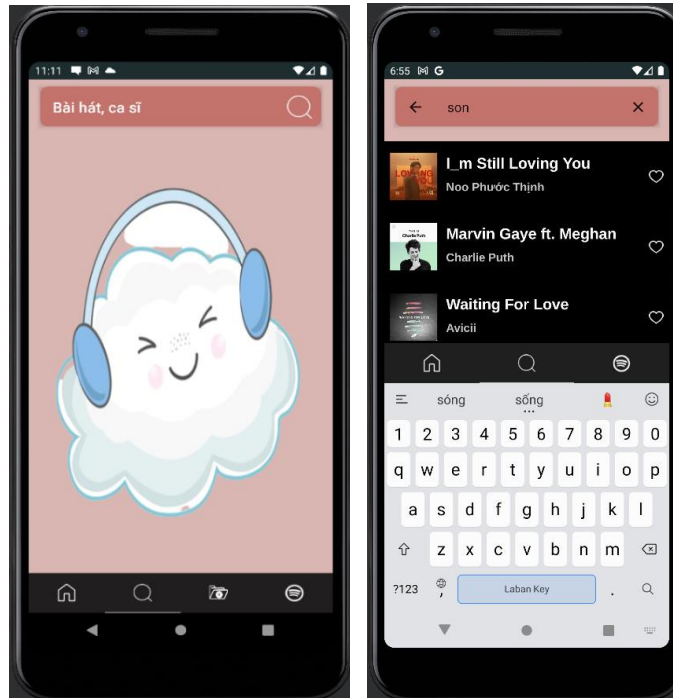
b. Khi đăng nhập với tư cách “Người dùng” (có trang thư viện của tôi)



Hình 3.4. Khi đăng nhập với tư cách “Người dùng”

Khi đăng nhập với người dùng đã có tài khoản, người dùng sẽ được sử dụng chức năng tạo Album riêng của mình. Được cập nhật ảnh đại diện, sửa đổi thông tin người dùng

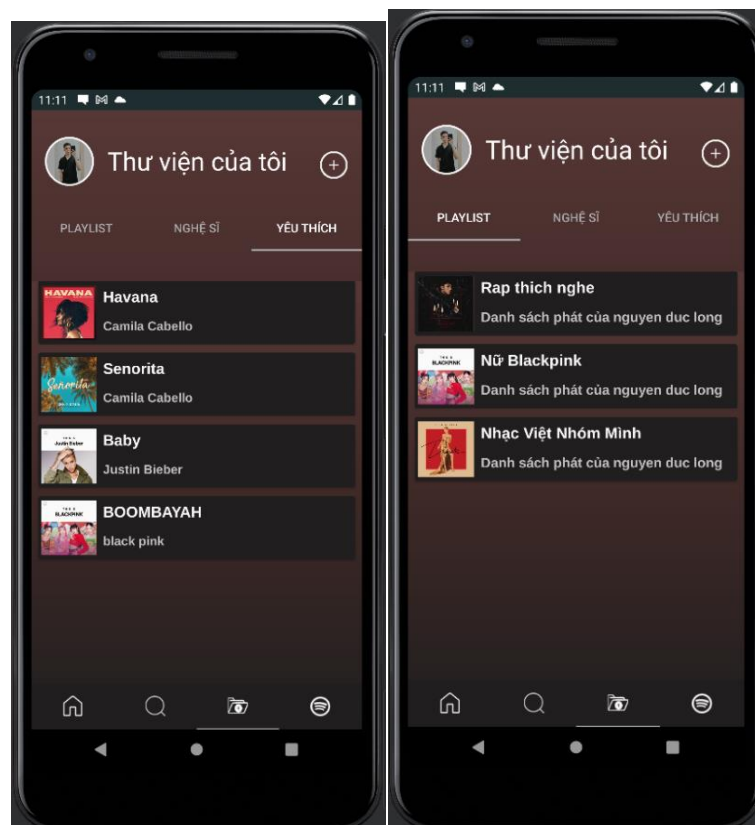
### 3.1.4 Giao diện màn hình tìm kiếm



Hình 3.5. Giao diện màn hình tìm kiếm

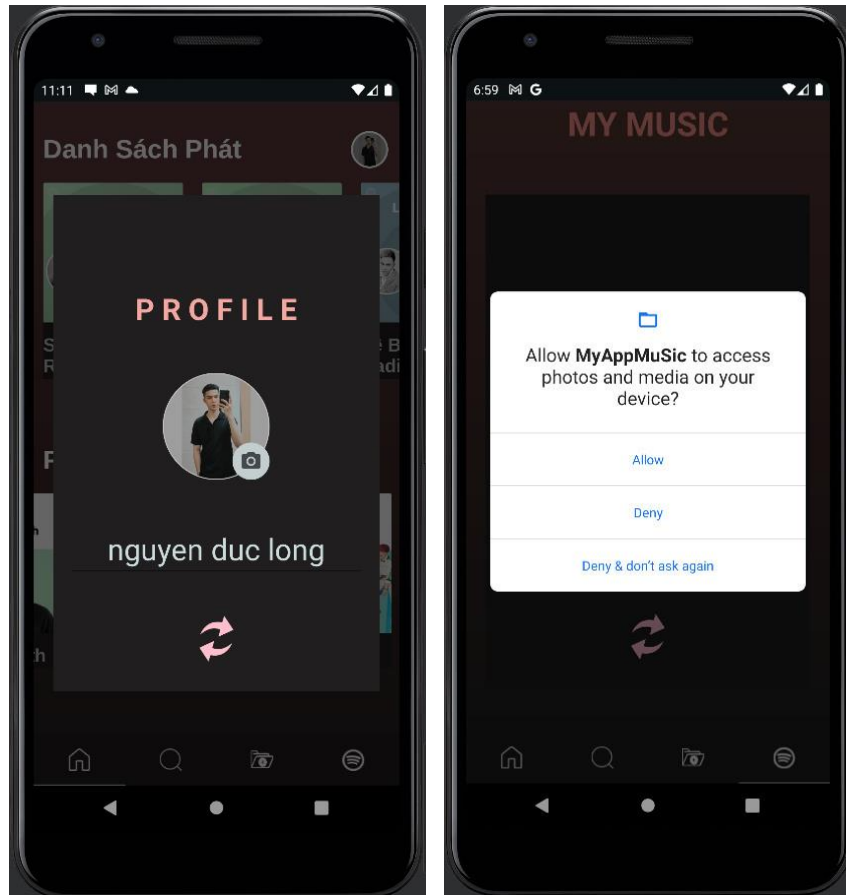
Giao diện màn hình tìm kiếm bài hát, hệ thống sẽ đề xuất bài hát có từ khóa mà người dùng nhập vào.

### 3.1.5 Giao diện màn hình thư viện của tôi



Hình 3.6. Giao diện màn hình thư viện của tôi

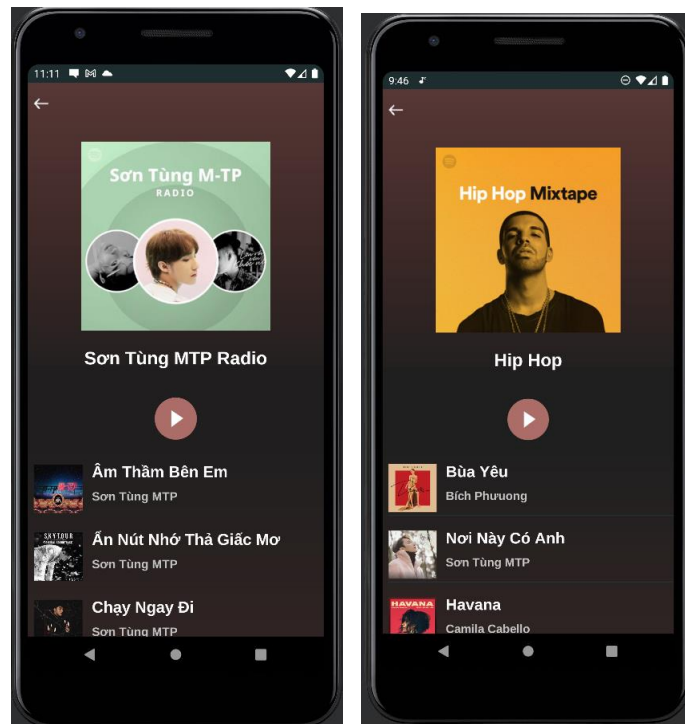
### 3.1.5 Giao diện màn hình thông tin người dùng



Hình 3.7. Giao diện màn hình thông tin người dùng

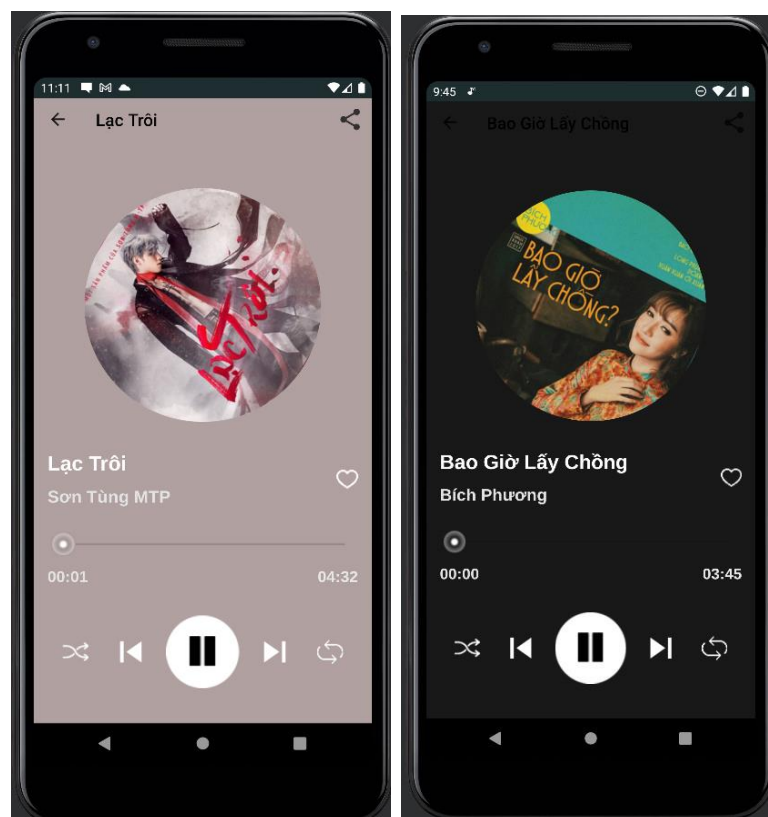
Người dùng có thể thay đổi ảnh trên profile của mình. Khi click vào icon hình ảnh thì hệ thống yêu cầu quyền truy cập bộ nhớ điện thoại. Người dùng chọn ảnh để thay đổi profile của mình.

### 3.1.6 Giao diện màn hình danh sách bài hát



Hình 3.8. Giao diện màn hình danh sách bài hát

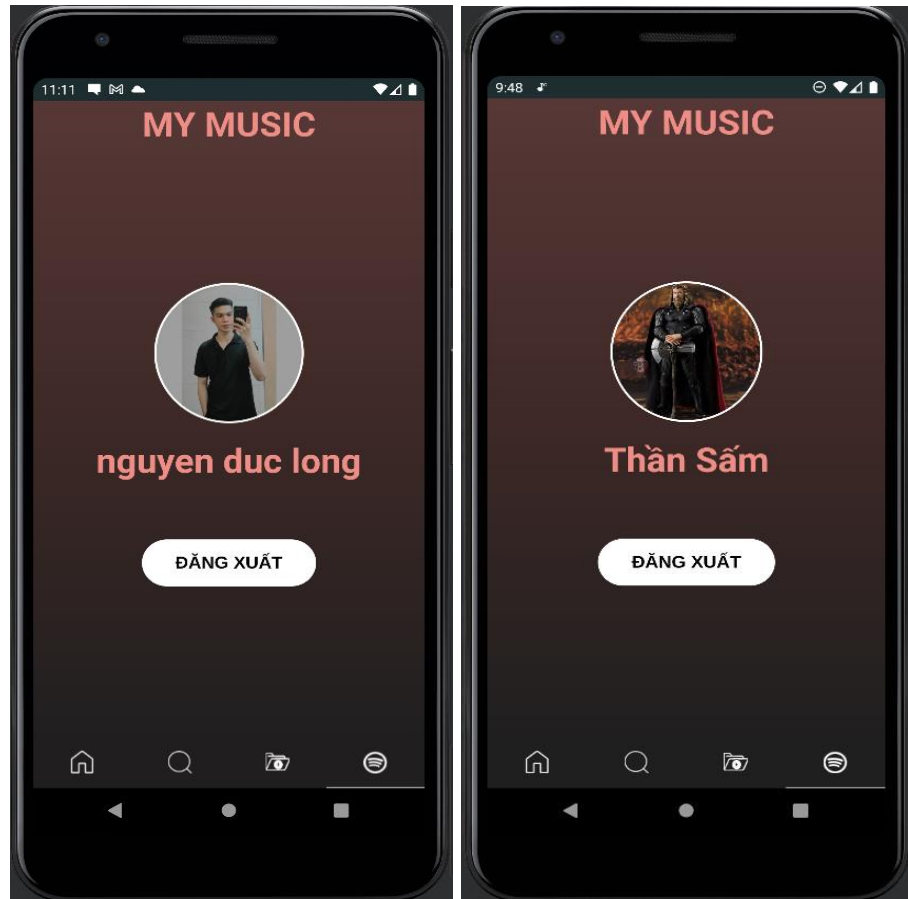
### 3.1.7 Giao diện màn hình phát nhạc



Hình 3.9. Giao diện màn hình phát nhạc

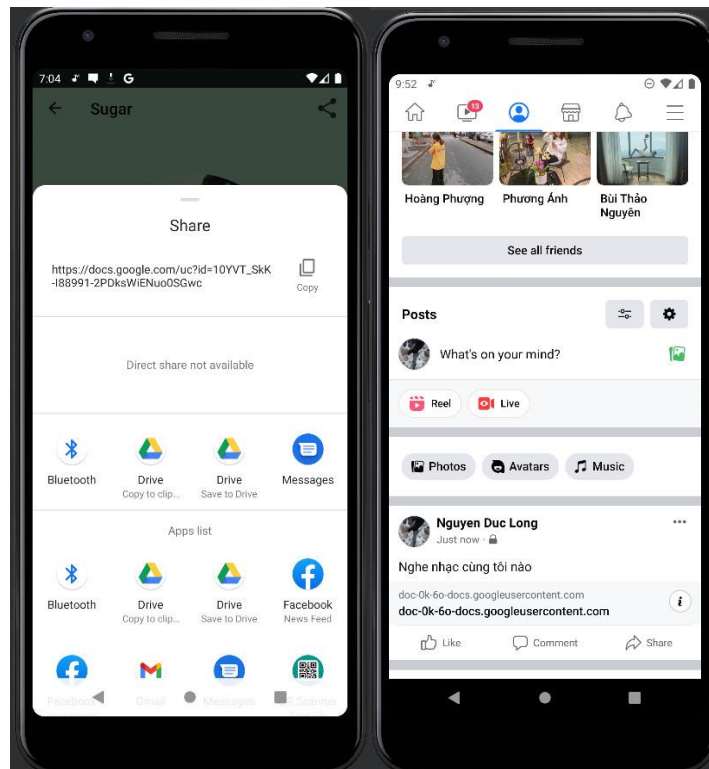
Người dùng có thể dừng, phát, chuyển bài, lặp bài, lặp xáo trộn trong album tại giao diện này.

### 3.1.8 Giao diện màn hình đăng xuất



Hình 3.10. Giao diện màn hình đăng xuất

### 3.1.9 Giao diện màn hình chia sẻ bài hát



Hình 3.11 Giao diện chia sẻ bài hát

## 3.2. GIAO DIỆN CHO ADMIN

### 3.2.1 Giao diện màn hình quản lý bài hát

Cột	Kiểu	Hàm	Null	Giá trị
IdBaiHat	int(11)			153
TenBaiHat	varchar(255)			Body Electric
TenCaSi	varchar(255)			Lana Del Rey
HinhBaiHat	varchar(255)			https://apptld.000webhostapp.com/HinhAnh/lanadelRey/Body%20Electric.jpg
LinkBaiHat	varchar(255)			https://docs.google.com/uc?id=1HI3wNX1myczErQ2r21wV5Bmg-oquFBnk

Hình 3.12. Giao diện màn hình quản lý bài hát

### 3.2.2 Giao diện màn hình quản trị chủ đề

Cột	Kiểu	Hàm	Null	Giá trị
IdChuDe	int(11)			1
TenChuDe	varchar(255)			Acoustic
HinhChuDe	varchar(255)			https://music4b.000webhostapp.com/HinhAnh/ChuDe/Acoustic.jpg

Thực hiện

Hình 3.13. Giao diện màn hình quản trị chủ đề

### 3.2.3 Giao diện màn hình trang quản trị tài khoản người dùng

Cột	Kiểu	Hàm	Null	Giá trị
UserName	varchar(255)			khách
Password	varchar(255)			khách
Name	varchar(255)			khách
Email	varchar(255)			khách@gmail.com
Image	varchar(255)			https://apptid.000webhostapp.com/HinhAnh/person-icon.jpg

Hình 3.14. Giao diện màn hình trang quản trị tài khoản người dùng

### 3.2.4 Giao diện màn hình trang quản trị nghệ sĩ

Cột	Kiểu	Hàm	Null	Giá trị
IdNghesi	int(11)			1
TenNghesi	varchar(255)			Bích Phương
HinhNghesi	varchar(255)			https://music4b.000webhostapp.com/HinhAnh/Nghesi/B%C3%A0ch%20Ph%C6%B0%C6%A1ng.jpg

Hình 3.15. Giao diện màn hình trang quản trị nghệ sĩ



### **3.3 KẾT LUẬN CHƯƠNG III**

#### **1. Những điểm ứng dụng làm được:**

##### **+ Đối với người dùng:**

- Cung cấp đến người dùng những album, bài hát được yêu thích nhất hiện nay.
- Cho phép người dùng đăng nhập vào ứng dụng bằng tài khoản Facebook của mình.
- Sau khi đăng nhập, người dùng có thể chia sẻ những bài hát mình yêu thích với mọi người.

##### **+ Đối với người quản trị:**

- Người quản trị có thể sửa đổi, cập nhật dữ liệu của ứng dụng thông qua trang web quản trị được xây dựng bằng framework Laravel.

#### **2. Nhược điểm của ứng dụng:**

- Nội dung của ứng dụng còn thiếu sót.
- Các chức năng chưa tối ưu.
- Bố cục, màu sắc của ứng dụng thiếu thu hút.

#### **3. Hướng phát triển của ứng dụng:**

- Ứng dụng hướng tới mục tiêu nâng cao trải nghiệm của người dùng, từ trải nghiệm về nghe nhạc, xem MV ca nhạc và phát triển chức năng tải về điện thoại những bài hát yêu thích, từ đó người dùng có thể tạo cho mình một playlist, một album nhạc của riêng mình.
- Dùng nhiều kỹ thuật cao hơn trong việc phát triển và quản lý ứng dụng giúp ứng dụng có giao diện hiện đại, chức năng đầy đủ, tính bảo mật cao và hơn hết là thuận lợi trong việc phát triển ứng dụng về sau.

#### **4. Những kiến thức kỹ năng đạt được sau khi thực hiện đồ án:**

- Sau khi thực hiện đồ án này, em thêm được một công cụ để tạo ứng dụng trên hệ điều hành Android, cải thiện được hơn trong việc code Java, PHP, CSS, MYSQL... áp dụng được framework Laravel vào việc xây dựng trang web quản lý ứng dụng để quản lý ứng dụng Android của mình(backend).
- Sau mỗi lần thực hiện đồ án em lại tích lũy được nhiều kinh nghiệm hơn để hoàn thành các đồ án sau tốt hơn.

## TÀI LIỆU THAM KHẢO

1. <https://developer.android.com/studio/run?hl=vi>
2. <https://viblo.asia/p/lien-ket-android-voi-serverlocalhost-n6BkGyAxG5aV>
3. <https://ntcde.com/android/tao-web-service-bang-php-va-mysql-cho-ung-dung-di-dong-part-1.html>
4. <https://giasutinhoc.vn/huong-dan-thuc-hanh/huong-dan-android/ket-noi-android-voi-mysql-phan-1-2/>
5. <https://tuandc.com/lap-trinh/huong-dan-cach-tao-form-gui-email-bang-php-su-dung-gmail-smtp.html>
6. <https://netcorecloud.com/tutorials/send-an-email-via-gmail-smtp-server-using-php/>
7. <https://www.tutorialspoint.com/laravel/index.htm>