

**BỘ GIÁO DỤC VÀ ĐÀO TẠO**  
**TRƯỜNG ĐẠI HỌC GIAO THÔNG VẬN TẢI**  
**PHÂN HIỆU TẠI THÀNH PHỐ HỒ CHÍ MINH**  
**BỘ MÔN CÔNG NGHỆ THÔNG TIN**



**BÁO CÁO THỰC TẬP CHUYÊN MÔN**  
**ĐỀ TÀI: XÂY DỰNG ỨNG DỤNG NGHE NHẠC**  
**SỬ DỤNG ANDROID STUDIO**

Giảng viên hướng dẫn:	Ths. Trần Thị Dung Trần Quốc Khánh
Sinh viên thực hiện:	Phạm Hoàng Yến
Mã số sinh viên:	6151071031
Lớp:	Công nghệ thông tin
Khóa:	61

Thành phố Hồ Chí Minh, tháng 7 năm 2023

## MỤC LỤC

DANH MỤC HÌNH ẢNH.....	4
CHƯƠNG 1. TỔNG QUAN .....	6
1.1. Lý do chọn đề tài .....	6
1.2. Mục tiêu của đề tài.....	7
1.3. Lựa chọn công cụ cài đặt .....	7
CHƯƠNG 2. CƠ SỞ LÝ THUYẾT .....	8
2.1. Công nghệ sử dụng .....	8
2.1.1. Material UI .....	8
2.1.2. SDK.....	8
2.1.3. API .....	8
2.1.4. MySQL.....	9
2.2. Ngôn ngữ lập trình.....	10
2.2.1. HTML .....	10
2.2.2. PHP .....	10
2.2.3. Java.....	11
2.3. Công cụ Android Studio .....	14
2.3.1. Giới thiệu.....	14
2.3.2. Ưu điểm.....	14
2.3.3. Nhược điểm .....	15
2.4. Hình thức nghe nhạc online .....	15
2.4.1. Tính năng cho người dùng .....	15
2.4.2. Cách thức sử dụng app.....	15
CHƯƠNG 3. PHÂN TÍCH VÀ THIẾT KẾ HỆ THỐNG .....	16

3.1. Tác nhân và hồ sơ .....	16
3.1.1. Tác nhân .....	16
3.1.2. Hồ sơ dữ liệu .....	16
3.2. Các thực thể và thuộc tính.....	16
3.3. Xây dựng mô hình thực thể liên kết .....	19
3.4. Chuyển từ mô hình thực thể liên kết sang mô hình quan hệ .....	19
CHƯƠNG 4. GIAO DIỆN NGƯỜI DÙNG.....	20
4.1. Giao diện trang chủ.....	20
4.2. Giao diện xem thêm.....	23
4.3. Giao diện danh sách bài hát .....	27
4.4. Giao diện phát 1 bài nhạc.....	28
4.5. Giao diện phát tất cả .....	30
4.6. Giao diện tìm kiếm .....	32
4.7. Icon của ứng dụng .....	33
4.8. Giao diện khi nhấn thích một bài .....	34
CHƯƠNG 5. XÂY DỰNG ỨNG DỤNG.....	34
5.1. Xây dựng cơ sở dữ liệu.....	34
5.1.1. Cơ sở dữ liệu.....	34
5.1.2. Bảng album.....	35
5.1.3. Bảng bài hát.....	35
5.1.4. Bảng ca sĩ .....	35
5.1.5. Bảng chủ đề .....	35
5.1.6. Bảng playlist.....	36
5.1.7. Bảng thể loại.....	36
5.2. Xây dựng PHP service .....	36

5.3. Xây dựng trang người dùng .....	39
CHƯƠNG 6. KẾT LUẬN VÀ KIẾN NGHỊ.....	40
6.1. Kết quả đạt được.....	40
6.2. Nhược điểm.....	41
6.3. Hướng phát triển.....	41
TÀI LIỆU THAM KHẢO .....	42

## DANH MỤC HÌNH ẢNH

Hình 3.1. Mô hình thực thể liên kết .....	<b>Error! Bookmark not defined.</b>
Hình 0.1. Giao diện trang chủ 1 .....	<b>Error! Bookmark not defined.</b>
Hình 0.2. Giao diện trang chủ 2.....	20
Hình 0.3. Giao diện trang chủ 3 .....	21
Hình 0.4. Giao diện khi nhấn icon xem thêm trong playlist .....	22
Hình 0.5. Giao diện khi nhấn icon xem thêm trong thể loại .....	23
Hình 0.6. Giao diện khi nhấn icon xem thêm trong album .....	24
Hình 0.7. Giao diện khi nhấn icon xem thêm trong nghệ sĩ.....	25
Hình 0.8. Giao diện khi nhấn vào mục trong thể loại .....	26
Hình 0.9. Giao diện khi nhấn vào phát tất cả mà danh sách trống .....	27
Hình 0.10. Giao diện khi nhấn vào icon phát tất cả ca khúc 1 (Phần play nhạc).28	
Hình 0.11. Giao diện khi nhấn trộn bài .....	28
Hình 0.12. Giao diện khi nhấn lặp bài .....	29
Hình 0.13. Giao diện khi nhấn vào icon phát tất cả ca khúc 2 (Phần playlist) ....	30
Hình 0.14. Giao diện khi tìm kiếm.....	31
Hình 0.15. Icon ứng dụng .....	32
Hình 0.16. Giao diện khi nhấn thích một bài .....	33

Hình 5.1. Cơ sở dữ liệu .....	33
Hình 5.2. Bảng album.....	34
Hình 5.3. Bảng bài hát.....	34
Hình 5.4. Bảng ca sĩ .....	34
Hình 5.5. Bảng chủ đề .....	34
Hình 5.6. Bảng playlist.....	35
Hình 5.7. Bảng thể loại.....	35

## CHƯƠNG 1. TỔNG QUAN

### 1.1. Lý do chọn đề tài

Hiện nay, với sự phát triển mạnh mẽ của internet và những ứng dụng của nó trong đời sống. Máy tính, điện thoại thông minh không còn là một phương tiện lạ lẫm đối với mọi người mà nó đã dần trở thành một công cụ làm việc và giải trí thông dụng và hữu ích của chúng ta ở mọi nơi nhờ có kết nối internet. Do đó việc xây dựng các ứng dụng cho điện thoại di động đang là một ngành công nghiệp mới đầy tiềm năng và hứa hẹn sự phát triển vượt bậc.

Phần mềm, ứng dụng cho điện thoại di động hiện nay rất đa dạng và phong phú trên các hệ điều hành di động. Các hệ điều hành như Android, IOS, Hybrid,... đã rất phát triển trên thị trường truyền thông di động.

Trong nhiều năm trở lại đây, hệ điều hành android ra đời với sự kế thừa những ưu việt của các hệ điều hành trước và sự kết hợp của nhiều công nghệ tiên tiến nhất hiện nay. Android đã nhanh chóng trở thành đối thủ cạnh tranh mạnh mẽ với các hệ điều hành trước đó và đang là hệ điều hành được ưa chuộng nhất.

Với sự phát triển nhanh chóng của xã hội nhu cầu giải trí thông qua điện thoại di động ngày càng phổ biến, vì vậy em đã chọn xây dựng đề tài **“Xây dựng ứng dụng nghe nhạc online trên hệ điều hành Android”** với mục đích nghiên cứu, tìm hiểu về ứng dụng nghe nhạc trên android để có thể đáp ứng nhu cầu giải trí đó, giúp mọi người có thể thư giãn thông qua ứng dụng.

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.

Trong khuôn khổ đề tài của mình, em chỉ dừng lại việc các chức năng cơ bản của trình nghe nhạc và phần tìm kiếm. Mặc dù em đã cố gắng rất nhiều để hoàn thành công việc do thời gian có giới hạn, kiến thức có hạn nên còn nhiều thiếu sót cần được bổ sung. Em mong nhận được ý kiến đóng góp của thầy cô.

## 1.2. Mục tiêu của đề tài

Với mục đích nghiên cứu, tìm hiểu về ứng dụng nghe nhạc trên android để có thể đáp ứng nhu cầu giải trí đó, giúp cho mọi người có thể thư giãn thông qua ứng dụng một cách dễ dàng

## 1.3. Lựa chọn công cụ cài đặt

Front-end:

- Phát triển ứng dụng trên nền tảng Android
- Công cụ phát triển Android Studio
- Ngôn ngữ: Java

Back-end:

- Cơ sở dữ liệu: phpMyAdmin của webhost
- Ngôn ngữ: PHP, JavaScript, HTML-CSS

Thư viện

- UI (material, design, circle image view,...)
- Retrofit2
- Picasso
- RecyclerView
- Cardview

## CHƯƠNG 2. CƠ SỞ LÝ THUYẾT

### 2.1. Công nghệ sử dụng

#### 2.1.1. Material UI

- Material UI là một thư viện các React Component đã được tích hợp thêm cả Google's Material Design và được xây dựng giữa React và Google's Material Design.
- Material UI đem đến cho bạn và trang web của bạn một giao diện hoàn toàn mới, với những button, textfield, toggle... được design theo một phong cách mới lạ, thay vì việc nhà nhà người người dùng Bootstrap như hiện nay.

#### 2.1.2. SDK

- Bộ công cụ SDK Android: sử dụng ngôn ngữ Java, được dùng để lập trình các ứng dụng trên nền tảng Android.
- SDK viết tắt của Software Development Kit là các công cụ và phần mềm dùng để phát triển ứng dụng thông qua một nền tảng nhất định
- Theo đó, SDK cung cấp các thư viện, tài liệu, mẫu template, sample code, tiện ích gỡ rối (debugging), các ghi chú hỗ trợ (documentation) hoặc các tài liệu bổ sung,... để nhà phát triển có thể tích hợp thêm vào phần mềm/ứng dụng của mình. Đa phần chúng thường là chức năng hiển thị quảng cáo, push notification,...
- SDK cũng có thể chứa các API dưới dạng thư viện hoặc một hệ thống phân cứng phức tạp nào đó.

#### 2.1.3. API

- API là cơ chế cho phép 2 thành phần phần mềm giao tiếp với nhau bằng một tập hợp các định nghĩa và giao thức
- Kiến trúc API thường được giải thích dưới dạng máy chủ và máy khách. Ứng dụng gửi yêu cầu được gọi là máy khách, còn ứng dụng gửi phản hồi được gọi là máy chủ. Như vậy, trong ví dụ về thời tiết, cơ sở dữ liệu của cơ quan thời tiết là máy chủ còn ứng dụng di động là máy khách.
- API hoạt động theo 4 cách khác nhau, tùy vào thời điểm và lý do chúng được tạo ra.



## **API SOAP**

- Các API này sử dụng Giao thức truy cập đối tượng đơn giản. Máy chủ và máy khách trao đổi thông điệp bằng XML. Đây là loại API kém linh hoạt được dùng phổ biến trước đây.

## **API RPC**

- Những API này được gọi là Lệnh gọi thủ tục từ xa. Máy khách hoàn thành một hàm (hoặc thủ tục) trên máy chủ còn máy chủ gửi kết quả về cho máy khách.

## **API Websocket**

- API Websocket là một bản phát triển API web hiện đại khác sử dụng các đối tượng JSON để chuyển dữ liệu. API WebSocket hỗ trợ hoạt động giao tiếp hai chiều giữa ứng dụng máy khách và máy chủ. Máy chủ có thể gửi thông điệp gọi lại cho các máy khách được kết nối, điều này khiến loại API này hiệu quả hơn API REST.

## **API REST**

- Đây là loại API phổ biến và linh hoạt nhất trên web hiện nay. Máy khách gửi yêu cầu đến máy chủ dưới dạng dữ liệu. Máy chủ dùng dữ liệu đầu vào từ máy khách này để bắt đầu các hàm nội bộ và trả lại dữ liệu đầu ra cho máy khách. Hãy cùng xem xét API REST chi tiết hơn ở bên dưới.

### **2.1.4. MySQL**

MySQL là một hệ quản trị cơ sở dữ liệu, một hệ quản trị cơ sở dữ liệu quan hệ, được phát triển, phân phối và hỗ trợ bởi MySQL AB.

Để làm việc với MySQL cần đăng ký kết nối, tạo cơ sở dữ liệu, quản lý người dùng, phân quyền sử dụng, thiết kế đối tượng Table của cơ sở dữ liệu và xử lý dữ liệu. Để quản lý và thao tác trên cơ sở dữ liệu ta có thể sử dụng giao diện đồ họa hoặc dạng Command line

MySQL cũng giống như các hệ quản trị CSDL: Access, SQL Server, PostgreSQL, Oracle, phần mềm mã nguồn mở do đó có thể tải miễn phí từ trang

chủ. Nó có nhiều phiên bản cho các hệ điều hành khác nhau: phiên bản Win32 cho các hệ điều hành dòng Windows, Linux, Mac OS X, Unix, Solaris...

MySQL là cơ sở dữ liệu tốc độ cao, ổn định và dễ sử dụng, có tính khả chuyển, hoạt động trên nhiều hệ điều hành cung cấp một hệ thống lớn các hàm tiện ích rất mạnh. Với tốc độ và tính bảo mật cao, MySQL rất thích hợp cho các ứng dụng có truy cập CSDL trên internet. MySQL server hoạt động trong các hệ thống nhúng hoặc client/server.

## **2.2. Ngôn ngữ lập trình**

### **2.2.1. HTML**

HTML là viết tắt của Hypertext Markup Language (ngôn ngữ đánh dấu siêu văn bản) là một ngôn ngữ đánh dấu chuẩn cho việc tạo ra các trang web. Thành phần chính của HTML là các thẻ tag, nhằm khai báo các phần tử và vị trí sắp xếp, xây dựng cấu trúc của các phần tử trong website.

### **2.2.2. PHP**

**PHP**(Hypertext Preprocessor) là ngôn ngữ script trên server được thiết kế để dễ dàng xây dựng các trang Web động. Mã PHP có thể thực thi trên Webserver để tạo ra mã HTML và xuất ra trình duyệt web theo yêu cầu của người sử dụng. Ngôn ngữ PHP ra đời năm 1994 Rasmus Lerdorf sau đó được phát triển bởi nhiều người trải qua nhiều phiên bản. Phiên bản hiện tại là PHP 5.0 đã được công bố 07/2004.

Có nhiều lý do khiến cho việc sử dụng ngôn ngữ này chiếm ưu thế, sau đây là một số lý do cơ bản:

- Miễn phí, download dễ dàng từ Internet.
- Ngôn ngữ rất dễ học, dễ viết
- Mã nguồn mở(open source code)
- Mã nguồn không phải sửa lại nhiều khi viết chạy cho các hệ điều hành từ Windows, Linux, Unix.
- Rất đơn giản trong việc kết nối với nhiều nguồn DBMS ví dụ như : MySQL, Microsoft SQL Server 2000, Oracle, PostgreSQL, Adabas, dBase, Empress, FilePro, Informix, InterBase, mSQL, Solid, Sybase, Velocis và nhiều hệ thống

CSDL thuộc Hệ Điều Hành Unix (Unix dbm) cùng bất cứ DBMS nào có sự hỗ trợ cơ chế ODBC (Open Database Connectivity) ví dụ như DB2 của IBM.

### **Cú pháp PHP:**

Đoạn mã php luôn được bắt đầu và kết thúc bởi cặp thẻ theo cú pháp:

```
<?php
    các lệnh của php;
?>
```

Đoạn mã php có thể đặt bất kỳ đâu trong tài liệu. Thông thường một trang php bao gồm các thẻ HTML như một trang HTML nhưng có thêm các đoạn mã php.

### **Các kiểu dữ liệu PHP**

Php hỗ trợ 5 kiểu dữ liệu như sau: Integer: sử dụng cho giá trị có kiểu dữ liệu là số nguyên. Double: sử dụng cho giá trị có kiểu dữ liệu là số thực. String: sử dụng cho các giá trị có kiểu dữ liệu là chuỗi và ký tự. Array: sử dụng cho các giá trị có kiểu dữ liệu là mảng. Object: sử dụng cho các giá trị có kiểu dữ liệu là đối tượng của lớp.

#### **2.2.3. Java**

Java là một trong những ngôn ngữ lập trình hướng đối tượng. Nó được sử dụng trong phát triển phần mềm, trang web, game hay ứng dụng trên các thiết bị di động.

- **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 MicroSystem 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. Cụ thể như sau

Khi viết mã, hệ thống tạo ra một tệp .java. Khi biên dịch mã nguồn của chương trình sẽ được biên dịch ra mã byte code. Máy ảo Java (Java Virtual Machine) sẽ thông dịch mã byte code này thành machine code (hay native code) khi nhận được yêu cầu chạy chương trình.

- **Tổng quan về ngôn ngữ lập trình java**

**Ưu điểm:** Phương pháp này giúp các đoạn mã viết bằng Java có thể chạy được trên nhiều nền tảng khác nhau. Với điều kiện là JVM có hỗ trợ chạy trên nền tảng này.

**Nhược điểm:** Cũng như các ngôn ngữ thông dịch khác, quá trình chạy các đoạn mã Java là chậm hơn các ngôn ngữ biên dịch khác (tuy nhiên vẫn ở trong một mức chấp nhận được).

- **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

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.

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.

## **2.3. Công cụ Android Studio**

### **2.3.1. Giới thiệu**

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 cho trường hợp người dùng không nhớ chính xác cú pháp giúp bạn tiết kiệm thời gian hiệu quả hơn.

### **2.3.2. Ưu điểm**

- Được phát triển bởi Google, cũng là chủ sở hữu của hệ điều hành Android
- Các công cụ hỗ trợ và được cập nhật mới nhất và đầy đủ
- Tính năng dễ làm quen và giao diện thân thiện, nó là điểm cộng lớn.
- Có tài liệu tham khảo và hướng dẫn đầy đủ cùng các diễn đàn dành cho lập trình viên Android.

### 2.3.3. Nhược điểm

- Là công cụ hỗ trợ tích hợp tất cả nên dữ liệu phải phát triển tối ưu nhất. Lượng lớn dữ liệu chiếm nhiều không gian bộ nhớ máy tính của bạn.
- Có thể kiểm tra được cash hoạt động của app thông qua giả lập của Android studio. Điều này làm đơ máy, lag, nóng máy tính và gây tiêu tốn nhiều pin.
- Android là một công cụ lập trình hỗ trợ mạnh mẽ với các hỗ trợ và cập nhật các tính năng mới nhất từ google. Google đã khắc phục vấn đề tối ưu tài nguyên máy tính giúp giảm bớt Android Studio trên máy tính cũ.

## 2.4. Hình thức nghe nhạc online

### 2.4.1. Tính năng cho người dùng

- Người dùng có thể xem ngẫu nhiên các chủ đề, nghệ sĩ, album được làm mới mỗi ngày và bài hát được nghe nhiều dựa vào độ yêu thích của từng bài.
- Tìm kiếm bài hát theo ý thích

### 2.4.2. Cách thức sử dụng app

Người dùng có thể truy cập vào ứng dụng vào màn hình chính và chọn các mục như chủ đề, nghệ sĩ, album, bài hát được yêu thích nhất để nghe.

Người dùng sử dụng cái button được cài đặt để chuyển bài, lặp bài hay phát ngẫu nhiên các ca khúc trong play list

Tìm kiếm ca khúc muốn nghe bằng cách nhấn vào mục tìm kiếm

Nhấn yêu thích bài hát bằng cách nhấn vào hình trái tim hoặc bỏ yêu thích bằng cách nhấn một lần nữa

## CHƯƠNG 3. PHÂN TÍCH VÀ THIẾT KẾ HỆ THỐNG

### 3.1. Tác nhân và hồ sơ

#### 3.1.1. Tác nhân

- Người dùng

#### 3.1.2. Hồ sơ dữ liệu

- Thông tin bài hát
- Thông tin nghệ sĩ
- Thông tin playlist
- Thông tin độ yêu thích
- Thông tin trình phát nhạc
- Thông tin tìm kiếm

### 3.2. Các thực thể và thuộc tính

#### ➤ Tập thực thể bài hát

**Mô tả:** Bài hát bao gồm các thuộc tính mã bài hát, tên bài hát, hình bài hát, ca sĩ bài hát, link bài hát, mã album, mã thể loại, mã playlist, mã ca sĩ, lượt thích bài hát. Một bài hát có thể có nhiều lượt thích hoặc không có lượt thích nào

STT	Thuộc tính	Mô tả
1	<b><u>idBaiHat</u></b>	Mã bài hát
2	tenBaiHat	Tên bài hát
3	hinhBaiHat	Hình bài hát
4	casiBaiHat	Tên ca sĩ bài hát
5	linkBaiHat	Link bài hát
6	<b>idAlbum</b>	Mã album
7	<b>idPlaylist</b>	Mã playlist
8	<b>idTheLoai</b>	Mã thể loại



9	<b>idCaSi</b>	Mã ca sĩ
10	LuotThich	Lượt thích

➤ **Tập thực thể album**

**Mô tả:** Album bao gồm các thuộc tính mã album, tên album, ca sĩ album, hình album, mã ca sĩ album

STT	Thuộc tính	Mô tả
1	<b><u>idAlbum</u></b>	Mã album
2	tenAlbum	Tên album
3	casiAlbum	Ca sĩ album
4	hinhAlbum	Hình album
5	<b>idCaSi</b>	Mã ca sĩ

➤ **Tập thực thể ca sĩ**

**Mô tả:** Ca sĩ bao gồm mã ca sĩ, tên ca sĩ và hình ca sĩ

STT	Thuộc tính	Mô tả
1	<b><u>idCaSi</u></b>	Mã ca sĩ
2	TenKD	Tên ca sĩ
3	hinhCaSi	Hình ảnh ca sĩ

➤ **Tập thực thể chủ đề**

**Mô tả:** Màu sắc bao gồm mã chủ đề, tên chủ đề, hình chủ đề

STT	Thuộc tính	Mô tả
1	<b><u>idChuDe</u></b>	Mã chủ đề
2	tenChuDe	Tên chủ đề

3	hinhChuDe	Hình chủ đề
---	-----------	-------------

➤ **Tập thực thể playlist**

**Mô tả:** Playlist bao gồm mã playlist, tên playlist, hình playlist, icon playlist.

Một playlist có nhiều bài hát, ca sĩ

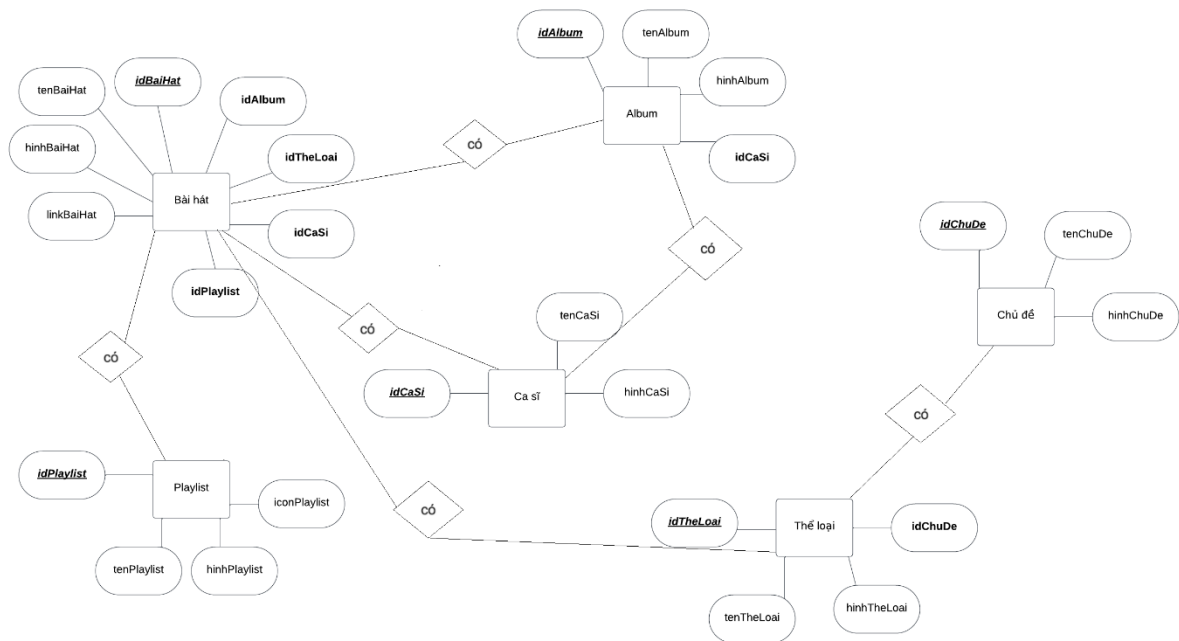
STT	Thuộc tính	Mô tả
1	<b><u>idPlaylist</u></b>	Mã playlist
2	tenPlaylist	Tên playlist
3	hinhPlaylist	Hình playlist
4	iconPlaylist	Icon playlist

➤ **Tập thực thể thể loại**

**Mô tả:** Thể loại bao gồm mã thể loại, mã chủ đề, tên thể loại, hình thể loại

STT	Thuộc tính	Mô tả
1	<b><u>idTheLoai</u></b>	Mã thể loại
2	<b><u>idChuDe</u></b>	Mã chủ đề
3	tenTheLoai	Tên thể loại
4	hinhTheLoai	Hình thể loại

### 3.3. Xây dựng mô hình thực thể liên kết



Hình 3.1. Mô hình thực thể liên kết

### 3.4. Chuyển từ mô hình thực thể liên kết sang mô hình quan hệ

BaiHat(idBaiHat, idAlbum, idTheLoai, idCaSi, idPlaylist, linkBaiHat, tenBaiHat, hinhBaiHat)

Album(idAlbum, idCaSi, tenAlbum, hinhAlbum)

Playlist(idPlaylist, tenPlaylist, hinhPlaylist, iconPlaylist)

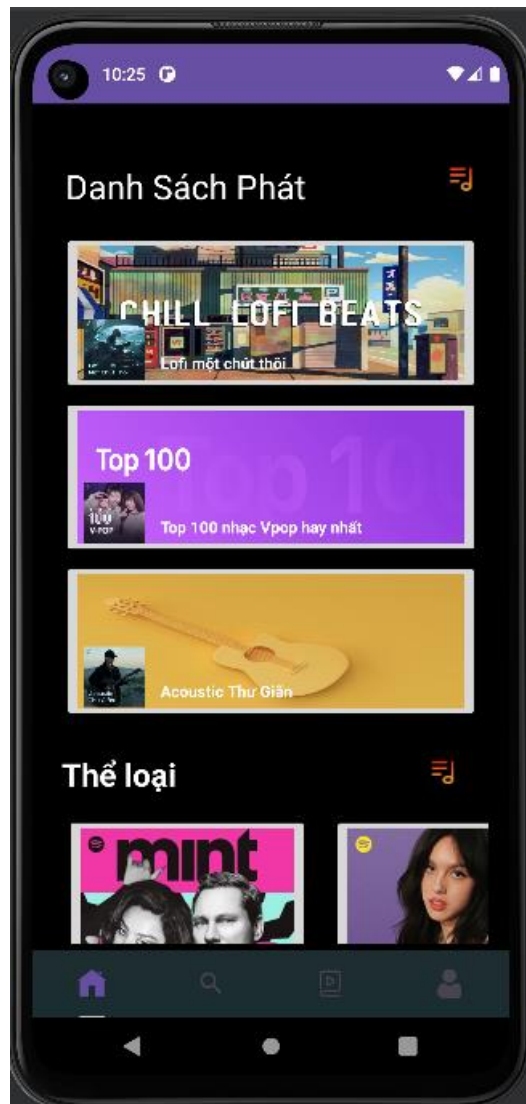
CaSi(idCaSi, tenCaSi, hinhCaSi)

TheLoai(idTheLoai, idChuDe, tenTheLoai, hinhTheLoai)

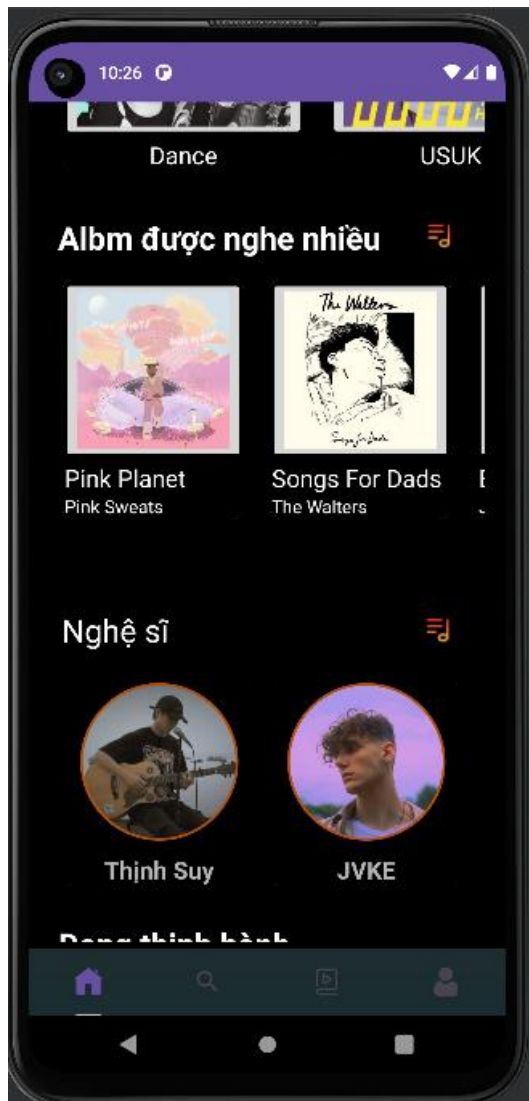
ChuDe(idChuDe, tenChuDe, hinhChuDe)

## CHƯƠNG 4. GIAO DIỆN NGƯỜI DÙNG

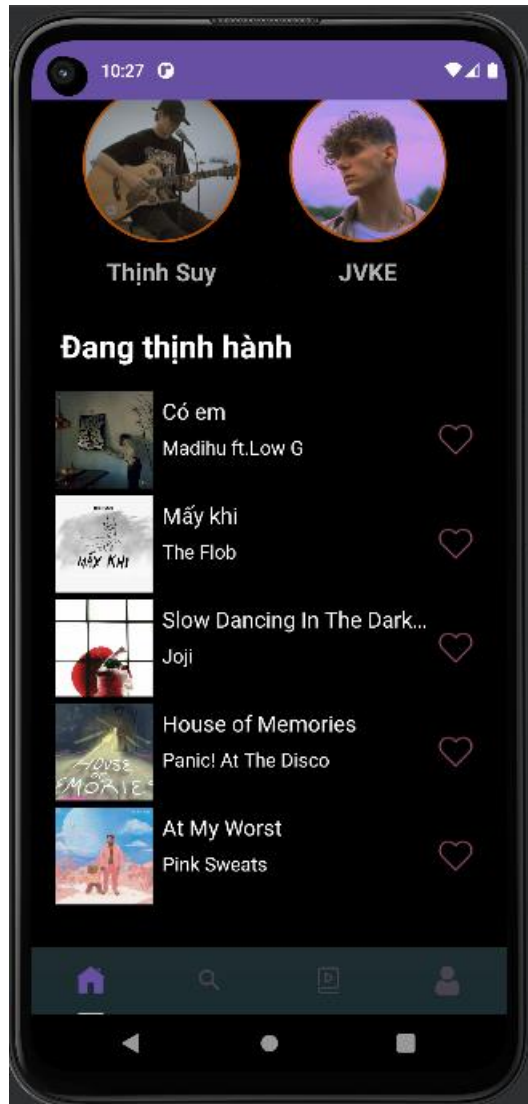
### 4.1. Giao diện trang chủ



Hình 4.1. Giao diện trang chủ 1

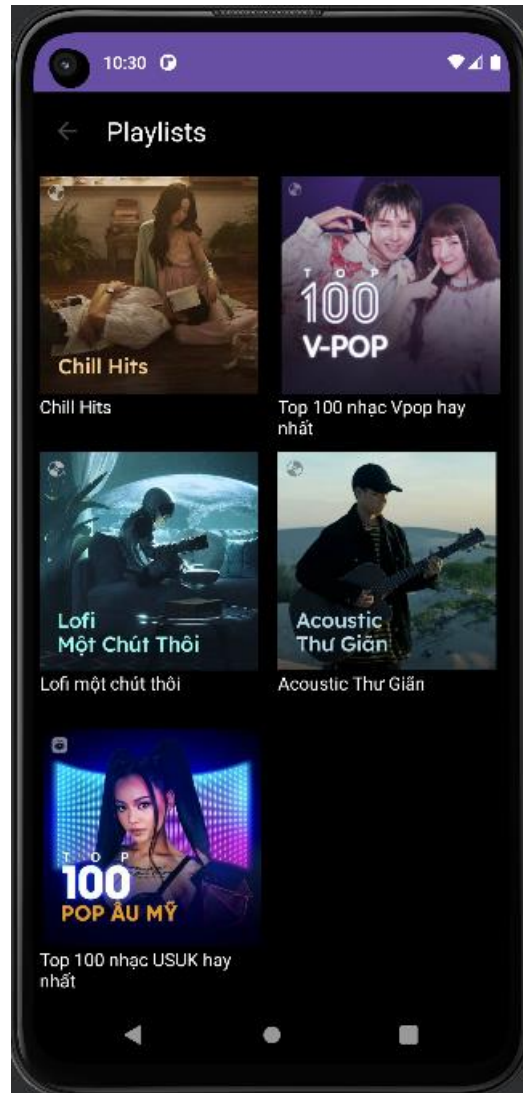


Hình 4.2. Giao diện trang chủ 2

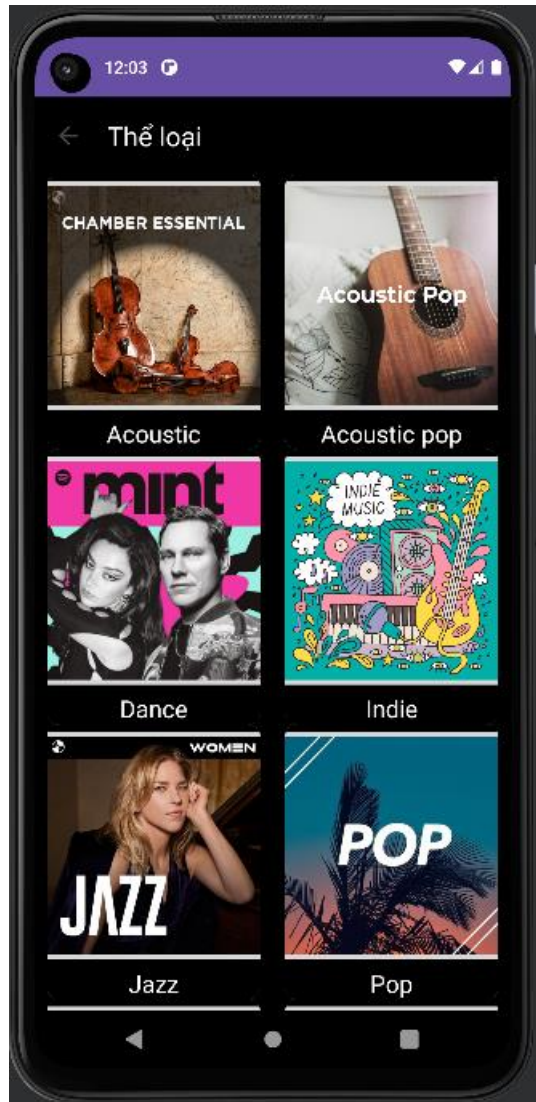


Hình 4.3. Giao diện trang chủ 3

## 4.2. Giao diện xem thêm

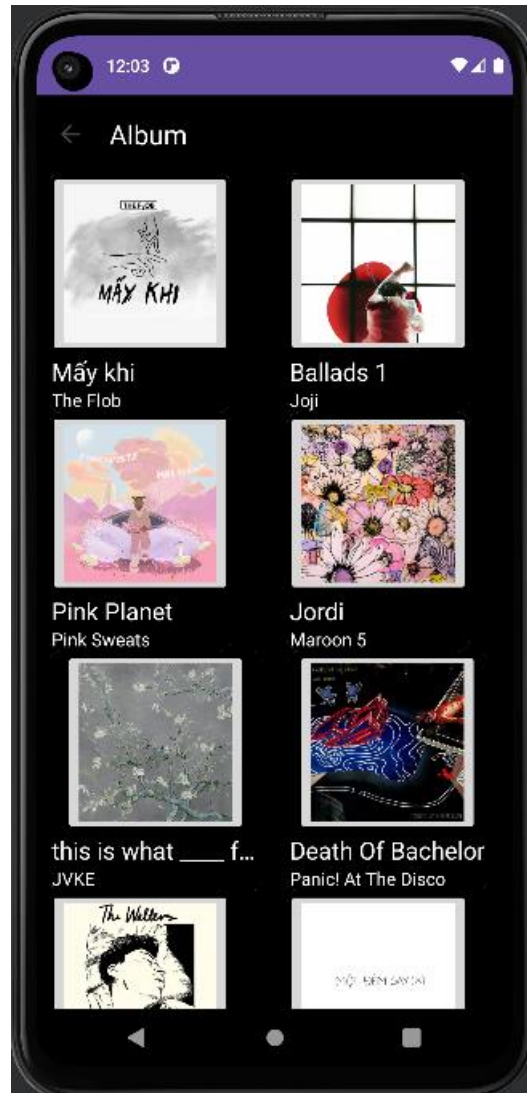


Hình 4.4. Giao diện khi nhấn icon xem thêm trong playlist

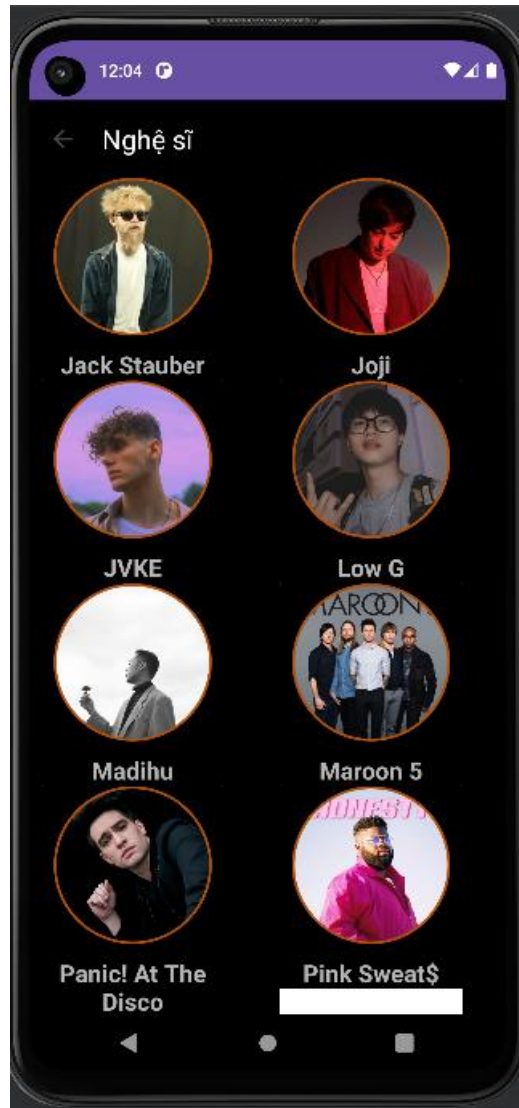


Hình 4.5. Giao diện khi nhấn icon xem thêm trong thể loại



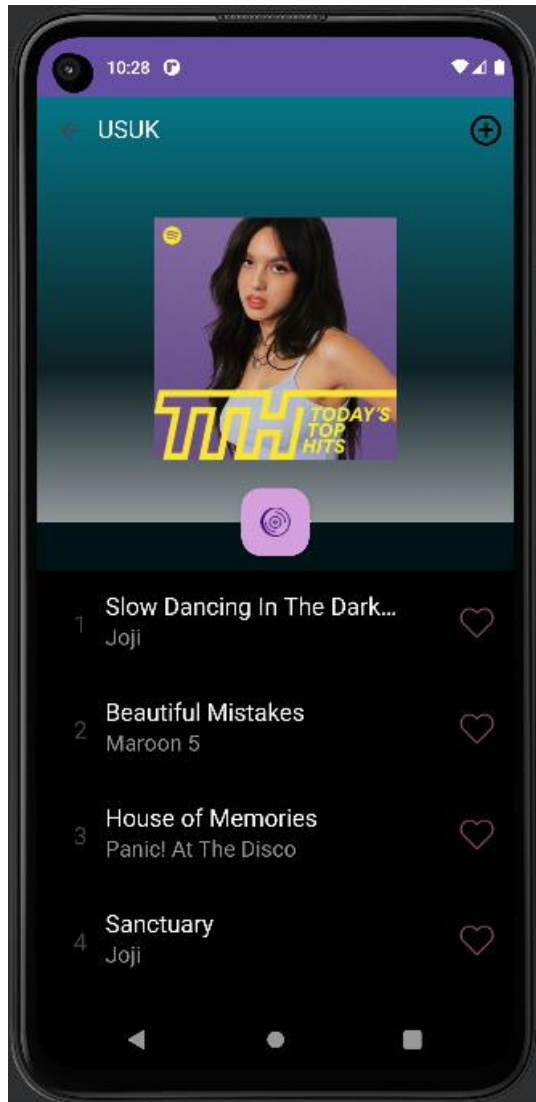


Hình 4.6. Giao diện khi nhấn icon xem thêm trong album



Hình 4.7. Giao diện khi nhấn icon xem thêm trong nghệ sĩ

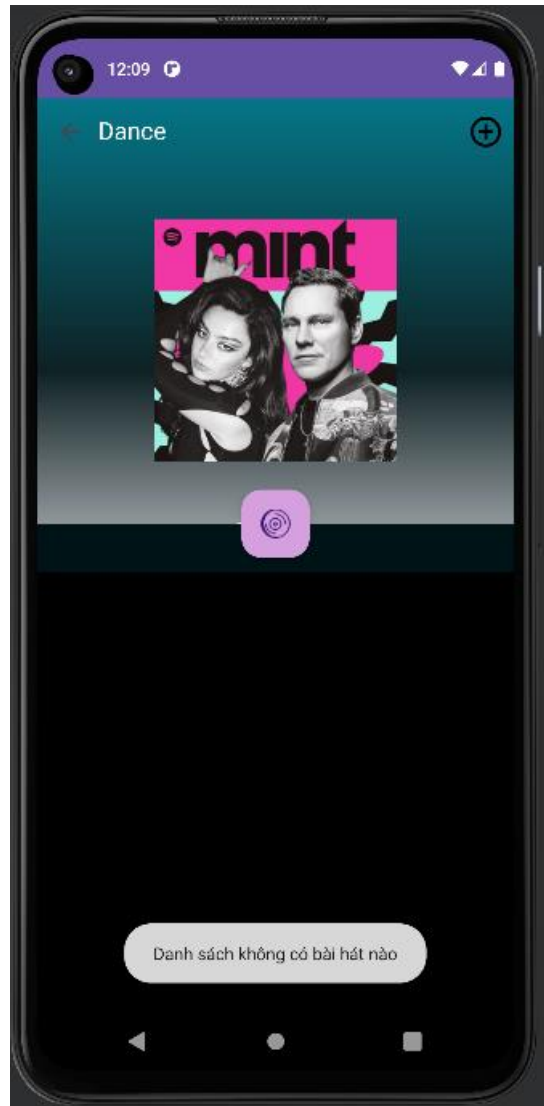
### 4.3. Giao diện danh sách bài hát



Hình 4.8. Giao diện khi nhấn vào mục trong thể loại

Khi nhấn vào icon cd ở dưới hình bài hát sẽ phát tất cả bài nhạc đang có trong danh sách.

Trường hợp danh sách không có bài nào thì sẽ phát một thông báo danh sách không có bài hát nào



*Hình 4.9. Giao diện khi nhấn vào phát tất cả mà danh sách trống*

#### **4.4. Giao diện phát 1 bài nhạc**

Người dùng có thể nhấn dừng hoặc tiếp tục phát ca khúc

Nhấn mũi tên để phát ca khúc tiếp theo hoặc phát ca khúc trước đó

Nhấn icon trộn bài để phát ngẫu nhiên các ca khúc trong danh sách sẵn có

Nhấn icon lặp lại bài hát để lặp lại bài đang bật

Không thể vừa bật trộn bài vừa bật lặp lại bài cùng lúc

Đĩa nhạc được cài animation để nó xoay tròn 360 độ

Trình phát được cài đặt thời gian nghỉ giữa các button cách nhau 5 giây để tránh việc crash app



Hình 4.10. Giao diện khi nhấn vào icon phát tất cả ca khúc 1 (Phần play nhạc)



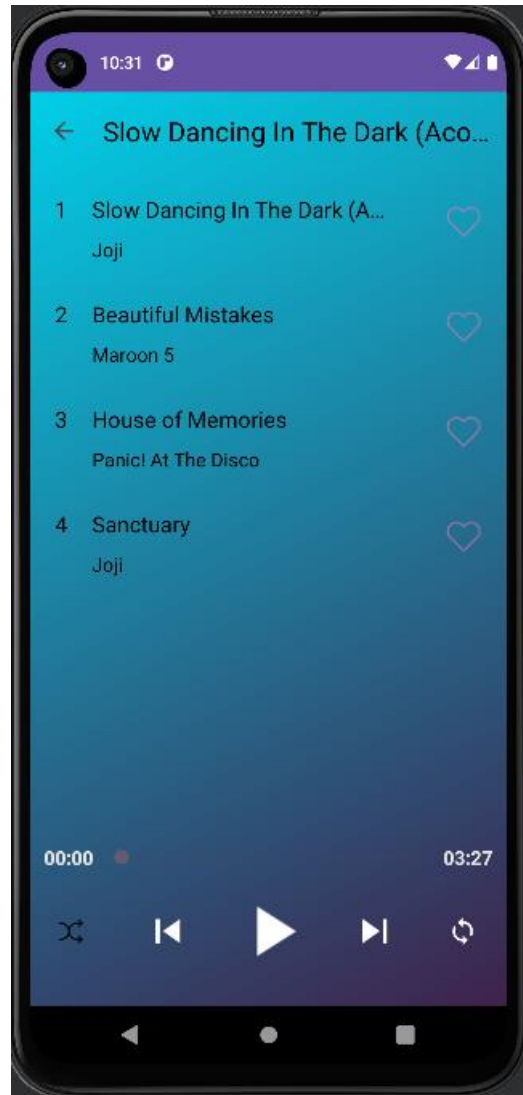
Hình 4.11. Giao diện khi nhấn trộn bài



Hình 4.12. Giao diện khi nhấn lặp bài

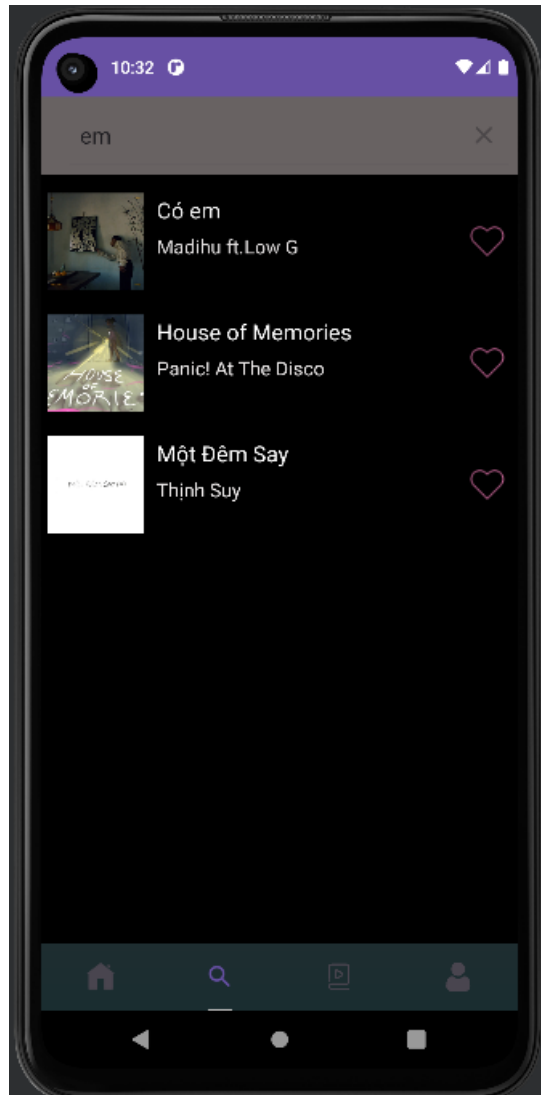
#### 4.5. Giao diện phát tất cả





Hình 4.13. Giao diện khi nhấn vào icon phát tất cả ca khúc 2 (Phần playlist)

#### 4.6. Giao diện tìm kiếm



Hình 4.14. Giao diện khi tìm kiếm

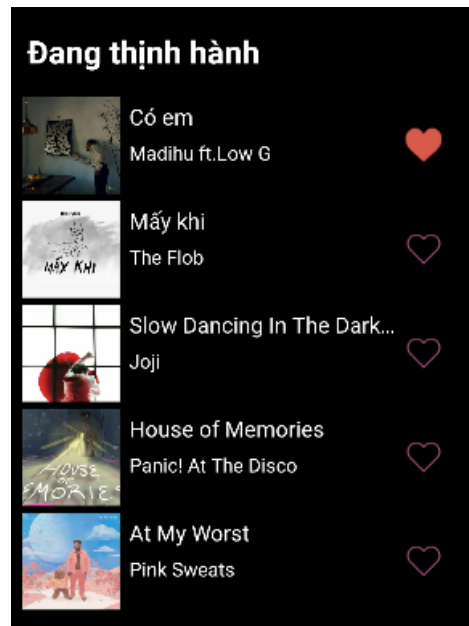


#### 4.7. Icon của ứng dụng



*Hình 4.15. Icon ứng dụng*

#### 4.8. Giao diện khi nhấn thích một bài



Hình 4.16. Nhấn thích 1 bài hát

## CHƯƠNG 5. XÂY DỰNG ỨNG DỤNG

### 5.1. Xây dựng cơ sở dữ liệu

#### 5.1.1. Cơ sở dữ liệu

Table	Action	Rows	Type	Collation	Size	Overhead
<input type="checkbox"/> album	★ Browse Structure Search Insert Empty Drop	9	InnoDB	utf8_unicode_ci	16.0 KiB	-
<input type="checkbox"/> baihat	★ Browse Structure Search Insert Empty Drop	11	InnoDB	utf8_unicode_ci	16.0 KiB	-
<input type="checkbox"/> casi	★ Browse Structure Search Insert Empty Drop	11	InnoDB	utf8_unicode_ci	32.0 KiB	-
<input type="checkbox"/> chude	★ Browse Structure Search Insert Empty Drop	6	InnoDB	utf8_unicode_ci	16.0 KiB	-
<input type="checkbox"/> playlist	★ Browse Structure Search Insert Empty Drop	5	InnoDB	utf8_unicode_ci	16.0 KiB	-
<input type="checkbox"/> theloai	★ Browse Structure Search Insert Empty Drop	8	InnoDB	utf8_unicode_ci	16.0 KiB	-
6 tables	Sum	50	InnoDB	utf8_unicode_ci	112.0 KiB	0 B

Hình 5.1. Cơ sở dữ liệu

### 5.1.2. Bảng album

		Table structure		Relation view						
#	Name	Type	Collation	Attributes	Null	Default	Comments	Extra	Action	
<input type="checkbox"/>	1 idAlbum	int(11)			No	None		AUTO_INCREMENT	Change	Drop  More
<input type="checkbox"/>	2 tenAlbum	varchar(255)	utf8_unicode_ci		No	None			Change	Drop  More
<input type="checkbox"/>	3 casAlbum	varchar(255)	utf8_unicode_ci		No	None			Change	Drop  More
<input type="checkbox"/>	4 hinhAlbum	varchar(255)	utf8_unicode_ci		No	None			Change	Drop  More
<input type="checkbox"/>	5 idCaSi	int(11)			No	None			Change	Drop  More

Hình 5.2. Bảng album

### 5.1.3. Bảng bài hát

#	Name	Type	Collation	Attributes	Null	Default	Comments	Extra	Action	
<input type="checkbox"/>	1 idBaiHat	int(11)			No	None		AUTO_INCREMENT	Change	Drop  More
<input type="checkbox"/>	2 tenBaiHat	varchar(255)	utf8_unicode_ci		No	None			Change	Drop  More
<input type="checkbox"/>	3 hinhBaiHat	varchar(255)	utf8_unicode_ci		No	None			Change	Drop  More
<input type="checkbox"/>	4 casBaiHat	varchar(255)	utf8_unicode_ci		No	None			Change	Drop  More
<input type="checkbox"/>	5 linkBaiHat	varchar(255)	utf8_unicode_ci		No	None			Change	Drop  More
<input type="checkbox"/>	6 idAlbum	int(11)			No	None			Change	Drop  More
<input type="checkbox"/>	7 idTheLoai	int(11)			No	None			Change	Drop  More
<input type="checkbox"/>	8 idPlaylist	int(11)			No	None			Change	Drop  More
<input type="checkbox"/>	9 idCaSi	int(11)			No	None			Change	Drop  More
<input type="checkbox"/>	10 LuotThich	int(11)			No	None			Change	Drop  More

Hình 5.3. Bảng bài hát

### 5.1.4. Bảng ca sĩ

#	Name	Type	Collation	Attributes	Null	Default	Comments	Extra	Action	
<input type="checkbox"/>	1 idCaSi	int(11)			No	None		AUTO_INCREMENT	Change	Drop  More
<input type="checkbox"/>	2 tenCaSi	varchar(250)	utf8_unicode_ci		No	None			Change	Drop  More
<input type="checkbox"/>	3 hinhCaSi	varchar(250)	utf8_unicode_ci		No	None			Change	Drop  More

Hình 5.4. Bảng ca sĩ

### 5.1.5. Bảng chủ đề

#	Name	Type	Collation	Attributes	Null	Default	Comments	Extra	Action	
<input type="checkbox"/>	1 idChuDe	int(11)			No	None		AUTO_INCREMENT	Change	Drop  More
<input type="checkbox"/>	2 tenChuDe	varchar(255)	utf8_unicode_ci		No	None			Change	Drop  More
<input type="checkbox"/>	3 hinhChuDe	varchar(255)	utf8_unicode_ci		No	None			Change	Drop  More

Hình 5.5. Bảng chủ đề

### 5.1.6. Bảng playlist

#	Name	Type	Collation	Attributes	Null	Default	Comments	Extra	Action
<input type="checkbox"/>	1 idPlaylist	int(11)			No	None		AUTO_INCREMENT	Change  Drop  More
<input type="checkbox"/>	2 tenPlaylist	varchar(255)	utf8_unicode_ci		No	None			Change  Drop  More
<input type="checkbox"/>	3 hinhPlaylist	varchar(255)	utf8_unicode_ci		No	None			Change  Drop  More
<input type="checkbox"/>	4 iconPlaylist	varchar(255)	utf8_unicode_ci		No	None			Change  Drop  More

Hình 5.6. Bảng playlist

### 5.1.7. Bảng thể loại

#	Name	Type	Collation	Attributes	Null	Default	Comments	Extra	Action
<input type="checkbox"/>	1 idTheLoai	int(11)			No	None		AUTO_INCREMENT	Change  Drop  More
<input type="checkbox"/>	2 idChuDe	int(11)			No	None			Change  Drop  More
<input type="checkbox"/>	3 tenTheLoai	varchar(255)	utf8_unicode_ci		No	None			Change  Drop  More
<input type="checkbox"/>	4 hinhTheLoai	varchar(255)	utf8_unicode_ci		No	None			Change  Drop  More

Hình 5.7. Bảng thể loại

## 5.2. Xây dựng PHP service

Kết nối cơ sở dữ liệu với app: connect.php

```
<?php

$hostname = "localhost";

$username = "id20832040_phyen02";

$password = "01032002aK@";

$dbname = "id20832040_musicapp";

$con = mysqli_connect($hostname, $username, $password,
$dbname);

mysqli_query($con, "SET NAMES 'utf8'");

?>
```

Lấy ra ngẫu nhiên 5 bài trong bảng album thay đổi theo ngày: albumhot.php

```
<?php
```

```

require "connect.php";

$query = "SELECT DISTINCT * FROM album ORDER BY rand(" .
date("Ymd"). ") LIMIT 5";

$data = mysqli_query($con, $query);

$arrayalbumhot = array();

while ($row = mysqli_fetch_assoc($data)) {

    $arrayalbumhot[] = $row;

}

print(json_encode($arrayalbumhot));

?>

```

Cập nhật lượt thích cho bài hát

```

<?php

require "connect.php";

$LuotThich = "1";

$idBaiHat = $_POST['idBaiHat'];

$query = "SELECT LuotThich FROM baihat WHERE idBaiHat =
'$idBaiHat'";

$dataLuotThich = mysqli_query($con,$query);

$row = mysqli_fetch_assoc($dataLuotThich);

$tongLuotThich = $row['LuotThich'];

if(isset($LuotThich)){

    $tongLuotThich += $LuotThich;

    $querySum = "UPDATE baihat SET LuotThich =
'$tongLuotThich' WHERE idBaiHat = '$idBaiHat'";

    $dataUpdate = mysqli_query($con, $querySum);

    if ($dataUpdate) {

```

```

        echo "Success";

    } else {

        echo "Fail";

    }

}

?>

```

Xoá lượt thích

```

<?php

    require "connect.php";

    $LuotThich = "1";

    $idBaiHat = $_POST['idBaiHat'];

    $query = "SELECT LuotThich FROM baihat WHERE idBaiHat =
'idBaiHat'";

    $dataLuotThich = mysqli_query($con,$query);

    $row = mysqli_fetch_assoc($dataLuotThich);

    $tongLuotThich = $row['LuotThich'];

    if(isset($LuotThich)){

        $tongLuotThich -= $LuotThich;

        $querySum = "UPDATE baihat SET LuotThich =
'$tongLuotThich' WHERE idBaiHat = '$idBaiHat'";

        $dataUpdate = mysqli_query($con, $querySum);

        if ($dataUpdate) {

            echo "Success";

        } else {

            echo "Fail";

        }

    }

}

```

?>

### 5.3. Xây dựng trang người dùng

Gửi phương thức và nhận dữ liệu từ phía Server

DataService.java

```
public interface DataService {
    @GET("playlistforcurrentday.php")
    Call<List<Playlist>> GetPlaylistCurrentDay();

    @GET("genreforcurrentday.php")
    Call<List<Genre>> GetGenreCurrentDay();

    @GET("albumhot.php")
    Call<List<Album>> GetAlbumHot();

    @GET("commonsong.php")
    Call<List<Song>> GetCommonSong();

    @GET("artistforcurrentday.php")
    Call<List<Artist>> GetArtist();

    @FormUrlEncoded
    @POST("song.php")
    Call<List<Song>> GetSongPlaylistList(@Field("idPlaylist")
String idPlaylist);

    @FormUrlEncoded
    @POST("song.php")
    Call<List<Song>> GetSongGenreList(@Field("idTheLoai")
String idPlaylist);

    @FormUrlEncoded
    @POST("song.php")
    Call<List<Song>> GetSongAlbumList(@Field("idAlbum") String
idAlbum);

    @FormUrlEncoded
    @POST("song.php")
    Call<List<Song>> GetSongArtistList(@Field("idCaSi") String
idCaSi);

    @GET("playlist_List.php")
    Call<List<Playlist>> GetPlaylist_List();

    @GET("genre_List.php")
    Call<List<Genre>> GetGenre_List();

    @GET("album_List.php")
```

```

Call<List<Album>> GetAlbum_List();

@GET("artist_List.php")
Call<List<Artist>> GetArtist_List();

@FormUrlEncoded
@POST("updateloved.php")
Call<String> UpdateLoved(@Field("LuotThich") String
LuotThich, @Field("idBaihat") String idBaiHat);

@FormUrlEncoded
@POST("deletelove.php")
Call<String> DeleteLove(@Field("LuotThich") String
LuotThich, @Field("idBaihat") String idBaiHat);

@FormUrlEncoded
@POST("search.php")
Call<List<Song>> GetSearchSong(@Field("key") String key);
}

```

## CHƯƠNG 6. KẾT LUẬN VÀ KIẾN NGHỊ

### 6.1. Kết quả đạt được

Trong suốt khoảng thời gian nghiên cứu và thực hiện đề tài, em đã cố gắng làm hết khả năng của mình và vì thời gian làm đồ án có hạn nên kết quả đạt được vẫn còn nhiều hạn chế nhưng em cũng học hỏi được khá nhiều kiến thức mới mẻ cũng như chuyên sâu về thuật toán, lập trình... và những việc em đã đạt được như sau:

- Xây dựng được ứng dụng nghe nhạc với các chức năng:
  - Giao diện gồm 4 mục được random từ cơ sở dữ liệu 5 mục mỗi ngày: playlist, nghệ sĩ, album, thể loại
  - Giao diện gồm bài hát được yêu thích dựa trên lượt thích của từng bài
  - Xem thêm tất cả trong 4 mục random
  - Danh sách các bài hát
  - Phát tất cả danh sách bài hát
  - Phát một bài hát
  - Tìm kiếm bài hát theo tên bài hát và nghệ sĩ
  - Các chức năng của button trong trình phát nhạc: phát kế tiếp, phát bài trước đó, ngẫu nhiên bài hát, lặp lại bài hát. Trình phát tự tắt khi thoát ra ngoài. Không thể vừa ngẫu nhiên bài hát vừa lặp lại bài được



## **6.2. Nhược điểm**

Trong quá trình làm đồ án, hiển nhiên sẽ có những lỗi chưa khắc phục hoàn toàn được cùng với thời gian có hạn nên sẽ có những chức năng chưa hoàn thiện:

- Vẫn còn thiếu nhiều chức năng như phân quyền, đăng nhập, đăng ký, thêm bài hát vào playlist.
- Thiếu chức năng thêm dữ liệu vào cơ sở dữ liệu

## **6.3. Hướng phát triển**

Từ những hạn chế trên, trong thời gian tới, em sẽ bổ sung thêm dữ liệu, hoàn thiện các chức năng để có thể sử dụng vào thực tế.

## TÀI LIỆU THAM KHẢO

- [1]      HowKTeam.com
- [2]      Youtube: Khoa Nguyễn
- [3]      dataotester.com
- [4]      php.net
- [5]      w3school.com
- [6]      Huỳnh Văn Giáp, giáo trình PHP và cơ sở dữ liệu MySQL, nhà xuất bản  
            Phương Nam
- [7]      Dương Thị Thu Hiền, slide môn PHP, trường CD CNTT Hữu nghị Việt  
            Hàn
- [8] Một số tài liệu tổng hợp PHP và Mysql của trường Học viện Bưu chính viễn  
            thông

Link Github: <https://github.com/redduck-02/MusicApp.git>