**TRƯỜNG ĐẠI HỌC SƯ PHẠM KỸ THUẬT TP.HCM**

**KHOA CÔNG NGHỆ THÔNG TIN**

**BỘ MÔN CÔNG NGHỆ PHẦN MỀM**



**TIỂU LUẬN CHUYÊN NGÀNH**

**XÂY DỰNG ỨNG DỤNG   
NGHE NHẠC TRỰC TUYẾN**

**GIÁO VIÊN HƯỚNG DẪN**

**ThS. TRẦN CÔNG TÚ**

**SINH VIÊN THỰC HIỆN**

**NGUYỄN PHƯƠNG NAM : 19110402**

**KHÓA 2019 - 2023**

**LỜI NÓI ĐẦU**

Trong cuộc sống hiện đại ngày nay, việc phát triển các thiết bị công nghệ hiện đại ngày càng nhanh đang kéo theo sự thay đổi về cách học hỏi, tiếp thu kiến thức của con người. Giờ đây, mọi thông tin và kiến thức đang ngày càng được số hóa khiến cho việc học của con người giờ đây không còn thủ công như trước.

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 nhiều sự phát triển vượt bậc của ngành khoa học kỹ thuật. Phần mềm, ứng dụng cho điện thoại di động hiện nay rất đa dạng và phong phú trên các hệ điều hành di động. Hệ điều hành Android ra đời với sự kế thừa những ưu việt của các hệ điều hành ra đời trước và sự kết hợp của nhiều công nghệ tiên tiến nhất hiện nay. Android đã nhanh chóng là đối thủ cạnh tranh mạnh mẽ với các hệ điều hành khác và đang là hệ điều hành di động của tương lai và được nhiều người ưa chuộng nhất. Ngày nay, 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 đề tài “Xây dựng ứng dụng nghe nhạc trực tuyến” 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.

Do trong khuôn khổ thời gian ngắn, trình độ chuyên môn, kinh nghiệm và kiến thức của bản thân còn hạn chế, nên em rất mong được sự góp ý của thầy để đề tài nghiên cứu của em ngày càng hoàn thiện hơn và được ứng dụng trong thực tế.

Em xin chân thành cảm ơn!

**LỜI CẢM ƠN**

Để hoàn thành tốt tiểu luận này, em xin gửi lời cảm ơn chân thành đến thầy Trần Công Tú, người đã trực tiếp hỗ trợ em trong suốt quá trình làm đề tài. Thầy đã đưa ra những lời khuyên từ kinh nghiệm thực tiễn của mình để định hướng cho em đi đúng với yêu cầu của đề tài đã chọn, luôn giải đáp thắc mắc và đưa ra những góp ý, chỉnh sửa kịp thời giúp em khắc phục nhược điểm và hoàn thành tốt cũng như đúng thời hạn đã đề ra.

Em xin kính chúc thầy luôn dồi dào sức khỏe và thành công hơn nữa trong sự nghiệp trồng người. Một lần nữa em xin chân thành cảm ơn.

**DANH MỤC HÌNH ẢNH**

[**Hình 1. Sơ đồ chức năng** 8](#_Toc122286879)

[**Hình 2. Mô hình hoá cơ sở dữ liệu trực tuyến** 16](#_Toc122286880)

[**Hình 3. Mô hình hoá cơ sở dữ liệu ngoại tuyến** 17](#_Toc122286881)

[**Hình 4. Cơ sở dữ liệu trực tuyến** 20](#_Toc122286882)

[**Hình 5. Cơ sở dữ liệu ngoại tuyến** 20](#_Toc122286883)

[**Hình 6. Màn hình chủ** 21](#_Toc122286884)

[**Hình 7. Màn hình quản lý tài khoản (quyền quản trị)** 22](#_Toc122286885)

**MỤC LỤC**

[CHƯƠNG 1: TỔNG QUAN VỀ ĐỀ TÀI 1](#_Toc122286831)

[1. Lý do chọn đề tài 1](#_Toc122286832)

[2. Yêu cầu chức năng 1](#_Toc122286833)

[3. Yêu cầu phi chức năng 2](#_Toc122286834)

[CHƯƠNG 2: KHẢO SÁT HIỆN TRẠNG THỰC TẾ 3](#_Toc122286835)

[1. Spotify 3](#_Toc122286836)

[2. SoundCloud 3](#_Toc122286837)

[3. Zing Mp3 4](#_Toc122286838)

[4. NCT 4](#_Toc122286839)

[5. Kết luận 5](#_Toc122286840)

[6. Mục tiêu của ứng dụng 5](#_Toc122286841)

[CHƯƠNG 3: CƠ SỞ LÝ THUYẾT 6](#_Toc122286842)

[1. Mô hình phát triển ứng dụng Model – View – ViewModel 6](#_Toc122286843)

[2. Cơ sở dữ liệu Room và Firestore 6](#_Toc122286844)

[3. Nền tảng xây dựng chức năng và giao diện ứng dụng 7](#_Toc122286845)

[CHƯƠNG 4: PHÂN TÍCH VÀ THIẾT KẾ HỆ THỐNG 8](#_Toc122286846)

[1. Sơ đồ chức năng 8](#_Toc122286847)

[2. Đặc tả chức năng 9](#_Toc122286848)

[2.1. Đăng ký 9](#_Toc122286849)

[2.2. Đăng nhập 10](#_Toc122286850)

[2.3. Đăng xuất 10](#_Toc122286851)

[2.4. Quên mật khẩu 11](#_Toc122286852)

[2.5. Quản lý thông tin cá nhân 11](#_Toc122286853)

[2.6. Tìm kiếm 12](#_Toc122286854)

[2.7. Phát bài hát 13](#_Toc122286855)

[2.8. Tải bài hát 13](#_Toc122286856)

[2.9. Quản lý bình luận 14](#_Toc122286857)

[2.10. Quản lý danh sách phát (cá nhân) 14](#_Toc122286858)

[2.11. Quản lý các thực thể của hệ thống 15](#_Toc122286859)

[3. Thiết kế cơ sở dữ liệu 16](#_Toc122286860)

[3.1. Trực tuyến 16](#_Toc122286861)

[3.2. Ngoại tuyến 17](#_Toc122286862)

[4. Thiết kế giao diện 17](#_Toc122286863)

[CHƯƠNG 5: XÂY DỰNG ỨNG DỤNG 18](#_Toc122286864)

[1. Tổ chức ứng dụng 18](#_Toc122286865)

[2. Xây dựng chức năng 18](#_Toc122286866)

[3. Cài đặt cơ sở dữ liệu 20](#_Toc122286867)

[3.1. Trực tuyến 20](#_Toc122286868)

[3.2. Ngoại tuyến 20](#_Toc122286869)

[4. Thiết kế giao diện 21](#_Toc122286870)

[CHƯƠNG 6: TỔNG KẾT 23](#_Toc122286871)

[1. Kết quả đạt được 23](#_Toc122286872)

[2. Ưu điểm 23](#_Toc122286873)

[3. Nhược điểm 23](#_Toc122286874)

[4. Khó khăn 23](#_Toc122286875)

[5. Bài học kinh nghiệm 23](#_Toc122286876)

[6. Hướng phát triển 24](#_Toc122286877)

[TÀI LIỆU THAM KHẢO 25](#_Toc122286878)

# TỔNG QUAN VỀ ĐỀ TÀI

Công nghệ phát triển ngày càng nhanh, những sản phẩm, phần mềm lập trình yêu cầu sự linh hoạt, hiệu quả, có thể cải tiến và phát triển dễ dàng. Vì vậy em quyết định lựa chọn lập trình ứng dụng theo hướng đối tượng sử dụng ngôn ngữ Kotlin là một trong những hướng phát triển phổ biến hiện nay. Phương pháp này giải quyết được những điểm yếu khi thiết kế theo phương pháp lập trình thủ tục thuần túy:

* Mã chương trình rõ ràng, dễ đọc, dễ hiểu và cô đọng.
* Chương trình được tổ chức thành những lớp phối hợp với nhau một cách linh hoạt, thống nhất.
* Mỗi lớp gồm có nhiều phương thức thực hiện các công việc khác nhau trong chương trình.
* Chương trình có tính linh hoạt cao.
* Có khả năng tái sử dụng tài nguyên, và mở rộng phát triển.

## Lý do chọn đề tài

Hiện nay, việc xây dựng các ứng dụng cho điện thoại di động đang là một ngành công nghiệp mới đầy tiềm năng và hứa hẹn nhiều sự phát triển vượt bậc của ngành khoa học kỹ thuật. Ứng dụng cho điện thoại di động hiện nay rất đa dạng và phong phú. 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 đề tài “Xây dựng ứng dụng nghe nhạc trực tuyến”.

## Yêu cầu chức năng

Các chức năng cần thiết cho ứng dụng:

* Đăng nhập, đăng ký, đăng xuất, ghi nhớ tài khoản đăng nhập, quên mật khẩu, đổi mật khẩu.
* Quản lý danh sách phát.
* Quản lý nghệ sĩ.
* Quản lý thể loại.
* Quản lý album.
* Quản lý quốc gia.
* Quản lý bài hát.
* Quản lý bình luận.
* Quản lý thông tin cá nhân của người dùng.
* Quản lý tài khoản.
* Tìm kiếm bài hát, tìm kiếm danh sách phát, tìm kiếm nghệ sĩ, tìm kiếm thể loại, tìm kiếm album, tìm kiếm quốc gia.
* Phát bài hát, tải bài hát.
* Phát hiện, phát các bài hát có trên thiết bị, thêm bài hát vào danh sách phát và quản lý danh sách phát (ngoại tuyến).

## Yêu cầu phi chức năng

Các yêu cầu phi chức năng của ứng dụng:

* Giao diện người dùng thân thiện, đẹp và dễ thao tác.
* Bảo mật thông tin của người dùng.
* Hiệu suất nhanh (các tác vụ phản hồi dưới 0,5s).
* Dễ dàng bảo trì và nâng cấp.

# KHẢO SÁT HIỆN TRẠNG THỰC TẾ

Hiện nay trên thị trường có khá nhiều ứng dụng nghe nhạc trực tuyến đã và đang phát triển rất thành công. Thông qua quá trình trải nghiệm các ứng dụng như Spotify, SoundCloud, Zing Mp3 và NCT, em đã đúc kết và đưa ra những nhận xét chủ quan dưới đây.

## Spotify

Spotify là dịch vụ cung cấp âm nhạc kỹ thuật số từ các hãng thu âm như Sony, EMI, Warner Music Group và Universal. Trụ sở pháp lý của hãng được đặt tại Luxembourg và trụ sở hoạt động tại Stockholm, Thụy Điển. Spotify được ra mắt ngày 7 tháng 10 năm 2008.

Ưu điểm:

* Giao diện đẹp, đơn giản và dễ dàng thao tác.
* Bộ sưu tập nhạc khổng lồ.
* Tiếp cận các bản phát hành rất nhanh.
* Danh sách nghe nhạc được điều chỉnh liên tục theo sở thích của người dùng.
* Có phiên bản miễn phí.
* Thường xuyên sản xuất các nội dung khác liên quan đến nghệ sĩ.

Nhược điểm:

* Phiên bản miễn phí có quá nhiều quảng cáo và giới hạn.
* Không cho phép bỏ qua bài hát thường xuyên và chất lượng âm thanh giới hạn ở 128 kpbs ở phiên bản miễn phí.

## SoundCloud

SoundCloud là trang mạng chia sẻ nhạc trực tuyến cho phép người dùng tải lên, quảng bá, chia sẻ âm thanh cũng như cho phép người nghe truyền phát âm thanh. Công ty có trụ sở tại Berlin, Đức do hai nhà sáng lập người Thụy Điển là Alexander Ljung và Eric Wahlforss khởi tạo vào năm 2007, chính thức hoạt động tháng 10 năm 2008.

Ưu điểm:

* Giao diện đơn giản, dễ dàng thao tác.
* Bộ sưu tập nhạc đa dạng.
* Đăng ký nghệ sĩ yêu thích.
* Danh sách nghe nhạc được điều chỉnh liên tục theo sở thích của người dùng.
* Có thể tham khảo danh sách phát của người khác.
* Luôn được cập nhật nhạc mới bởi cộng đồng sử dụng.
* Hoàn toàn miễn phí.

Nhược điểm:

* Chất lượng âm thanh không được đảm bảo.
* Không kiểm soát được bài hát do cộng đồng đóng góp.
* Không thể tìm kiếm nâng cao.
* Có bản trả phí nhưng không tương xứng với chất lượng nó mang lại.

## Zing Mp3

Zing MP3 là một dịch vụ âm nhạc kỹ thuật số thuộc hệ thống Zing được quản lý và vận hành bởi VNG. Ra mắt lần đầu từ năm 2007, đến nay Zing MP3 được coi là một trong những trang web âm nhạc trực tuyến lớn nhất Việt Nam.

Ưu điểm:

* Giao diện nhiều tùy chọn, quyến rũ với tông màu tím và trắng.
* Chất lượng âm thanh có thể lên đến 320 kpbs và tải nhạc miễn phí ở chất lượng 128 kpbs.
* Danh sách nghe nhạc được điều chỉnh liên tục theo sở thích của người dùng.
* Đường truyền riêng khi sử dụng phiên bản trả phí.
* Tạo, quản lý và phát nhạc, danh sách phát ngoại tuyến.

Nhược điểm:

* Thao tác khá rườm rà.
* Cung cấp quá nhiều thông tin không cần thiết.
* Phiên bản miễn phí khá nhiều quảng cáo.

## NCT

NCT (NhacCuaTui) là một dịch vụ âm nhạc kỹ thuật số thành lập từ năm 2007, được coi là một trong những trang web âm nhạc trực tuyến lớn nhất Việt Nam.

Ưu điểm:

* Giao diện dễ quan sát, đơn giản và mới lạ.
* Danh sách nghe nhạc được điều chỉnh liên tục theo sở thích của người dùng.
* Tạo, quản lý và phát nhạc, danh sách phát ngoại tuyến.
* Đường truyền riêng khi sử dụng phiên bản trả phí.

Nhược điểm:

* Tốc độ phản hồi hơi trễ.
* Phiên bản miễn phí khá nhiều quảng cáo.

## Kết luận

Từ những nhận xét về các ứng dụng nghe nhạc trực truyến phổ biến hiện nay như trên, em quyết định sẽ xây dựng một ứng dụng nghe nhạc trực tuyến trên hệ điều hành Android giúp người yêu nhạc thoả mãn nhu cầu thưởng thức của mình một cách dễ dàng và trọn vẹn nhất, đồng thời có đủ các chức năng cần thiết của một ứng dụng nghe nhạc trực tuyến thông dụng.

## Mục tiêu của ứng dụng

Dựa trên chức năng của các ứng dụng được khảo sát, các yêu cầu đối với ứng dụng:

* Hỗ trợ phát nhạc trực tuyến và ngoại tuyến với các bài hát có sẵn trên máy.
* Tìm kiếm và phân loại bài hát theo tên, theo album, theo nghệ sĩ, theo thể loại, theo quốc gia, theo danh sách phát (tuỳ chỉnh hoặc có sẵn).
* Quản lý bài hát, danh sách phát, nghệ sĩ, album, thể loại, quốc gia, thông tin tài khoản, người dùng.
* Đăng nhập, đăng ký, đăng xuất, khôi phục mật khẩu.
* Để lại bình luận cho từng bài hát và tải bài hát về thiết bị.
* Tạo giao diện trực quan và dễ dàng thao tác.
* Bảo vệ thông tin khách hàng, hiệu suất nhanh và dễ dàng thao tác, bảo trì.

# CƠ SỞ LÝ THUYẾT

## Mô hình phát triển ứng dụng Model – View – ViewModel

MVVM là viết tắt của Model - View - ViewModel, đây là mô hình hỗ trợ kết nối dữ liệu hai chiều giữa View và View Model. Mô hình MVVM được trình bày như sau:

* Model là các đối tượng giúp truy xuất và thao tác trên dữ liệu thực sự.
* View là phần giao diện của ứng dụng để hiển thị dữ liệu và nhận tương tác của người dùng.
* ViewModel sẽ đảm nhận công việc đồng bộ dữ liệu từ Model lên View. Mối quan hệ giữa View và ViewModel là View sẽ được ánh xạ tới ViewModel nhưng ViewModel lại không biết thông tin gì về View vì nó được ẩn giấu qua cách sử dụng Data-binding và cơ chế của mô hình quan sát (Observer). Một ViewModel có thể được ánh xạ từ nhiều View.

Trong mô hình MVVM, các tầng bên dưới sẽ không biết được các thông tin gì về tầng bên trên nó (theo thứ tự View, ViewModel, Model). [1]

Trong quá trình phát triển ứng dụng Android bằng ngôn ngữ Kotlin, MVVM thường kết hợp với Coroutines và Dependency Injection, cụ thể:

* Coroutines về cơ bản có thể hiểu nó như một Thread hạng nhẹ, nhưng nó không phải là 1 Thread, chúng chỉ hoạt động tương tự 1 Thread). Coroutine có thể chạy song song, chờ lẫn nhau hoặc giao tiếp với nhau. Điểm khác biệt lớn nhất là Coroutine có chi phí rất thấp so với Thread, chúng ta có thể tạo ra hàng ngàn Coroutine một cách dễ dàng. [2]
* Dependency Injection là một dạng mẫu thiết kế ứng dụng với mục đích ngăn chặn sự phụ thuộc giữa các lớp, để khiến cho mã nguồn dễ hiểu hơn, trực quan hơn, nhằm phục vụ cho mục đích bảo trì và nâng cấp. [3]

## Cơ sở dữ liệu Room và Firestore

Room Database (Room Persistence Library) là một phần trong Android Architecture Components, nó giúp cho việc thao tác với đối tượng SQLiteDatabase trở lên dễ dàng hơn, giảm thiểu số lượng câu truy vấn bằng những chú thích sẵn có và xác minh truy vấn tại thời điểm biên dịch. [4]

Firebase Database là một cơ sở dữ liệu NoSQL được lưu trữ đám mây cho phép lưu trữ và đồng bộ dữ liệu. Dữ liệu được lưu trữ dưới dạng JSON và được đồng bộ hóa theo thời gian thực cho mọi máy kết nối. [5]

## Nền tảng xây dựng chức năng và giao diện ứng dụng

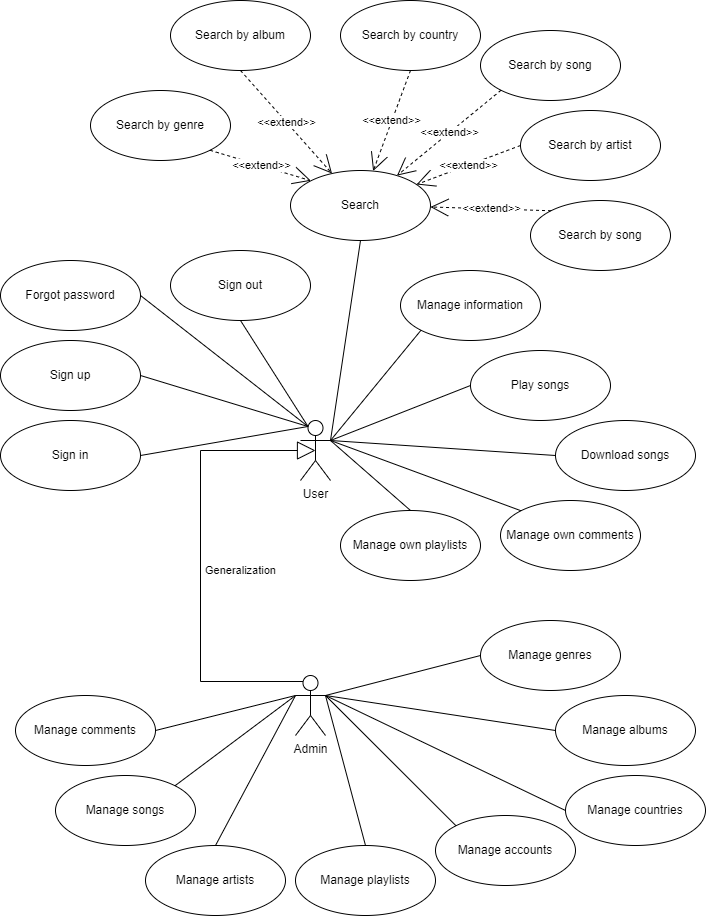
Sử dụng các kĩ thuật lập trình cơ bản với ngôn ngữ Kotlin.

Sử dụng phương pháp lập trình hướng đối tượng (Object Oriented Programing - OOP) trong ngôn ngữ lập trình Kotlin để thiết kế các lớp và các đối tượng.

Sử dụng XML (eXtensible Markup Language) để xây dựng giao diện ứng dụng.

# PHÂN TÍCH VÀ THIẾT KẾ HỆ THỐNG

## Sơ đồ chức năng



**Hình 1. Sơ đồ chức năng**

Với khách (User):

* Đăng nhập, đăng ký, đăng xuất, ghi nhớ tài khoản đăng nhập, quên mật khẩu, đổi mật khẩu.
* Quản lý danh sách phát của chính họ.
* Thêm bài hát vào danh sách phát, thêm danh sách phát của hệ thống thành của họ.
* Quản lý thông tin cá nhân.
* Quản lý bình luận.
* Tìm kiếm bài hát, tìm kiếm danh sách phát, tìm kiếm nghệ sĩ, tìm kiếm thể loại, tìm kiếm album, tìm kiếm quốc gia.
* Phát bài hát trực tuyến, tải bài hát trực tuyến.
* Phát các bài hát có trên thiết bị, thêm bài hát vào danh sách phát và quản lý danh sách phát (ngoại tuyến).

Với người quản lí (Admin):

* Có toàn quyền của khách.
* Quản lý danh sách phát hệ thống.
* Quản lý nghệ sĩ.
* Quản lý thể loại.
* Quản lý album.
* Quản lý quốc gia.
* Quản lý bài hát.
* Quản lý tài khoản.

## Đặc tả chức năng

### Đăng ký

|  |  |
| --- | --- |
| Tên | Đăng ký |
| Mô tả | Đăng ký để tạo tài khoản dùng cho việc đăng nhập vào ứng dụng |
| Tác nhân | Khách hàng, người quản trị |
| Độ ưu tiên | Phải có |
| Điều kiện kích hoạt | Người dùng muốn đăng ký để tạo tài khoản |
| Tiền điều kiện | Thiết bị đã được kết nối internet khi thực hiện |
| Kết quả mong đợi | Đăng ký tạo tài khoản thành công |
| Luồng thực hiện | 1. Người dùng mở ứng dụng 2. Người dùng nhập thông tin tài khoản và nhấn đăng ký 3. Hệ thống xác nhận thông tin đăng ký thành công, tạo tài khoản và cho phép người dùng truy cập vào ứng dụng |

### Đăng nhập

|  |  |
| --- | --- |
| Tên | Đăng nhập |
| Mô tả | Đăng nhập vào ứng dụng để sử dụng dịch vụ của ứng dụng |
| Tác nhân | Khách hàng, người quản trị |
| Độ ưu tiên | Phải có |
| Điều kiện kích hoạt | Người dùng muốn đăng nhập vào ứng dụng |
| Tiền điều kiện | Tài khoản đã được tạo sẵn  Tài khoản đã được phân quyền  Thiết bị đã được kết nối internet khi thực hiện |
| Kết quả mong đợi | Đăng nhập ứng dụng thành công |
| Luồng thực hiện | 1. Người dùng mở ứng dụng 2. Người dùng nhập thông tin tài khoản và nhấn đăng nhập 3. Hệ thống xác nhận thông tin đăng nhập thành công và cho phép người dùng truy cập vào ứng dụng |

### Đăng xuất

|  |  |
| --- | --- |
| Tên | Đăng xuất |
| Mô tả | Đăng xuất khỏi ứng dụng |
| Tác nhân | Khách hàng, người quản trị |
| Độ ưu tiên | Phải có |
| Điều kiện kích hoạt | Người dùng muốn đăng xuất |
| Tiền điều kiện | Tài khoản đã đăng nhập vào ứng dụng  Thiết bị đã được kết nối internet khi thực hiện |
| Kết quả mong đợi | Đăng xuất thành công |
| Luồng thực hiện | 1. Người dùng mở màn hình thông tin cá nhân 2. Người dùng nhấn đăng xuất 3. Hệ thống xác nhận đăng xuất và đưa người dùng về màn hình đăng nhập |

### Quên mật khẩu

|  |  |
| --- | --- |
| Tên | Quên mật khẩu |
| Mô tả | Người dùng quên mật khẩu và không thể đăng nhập vào ứng dụng |
| Tác nhân | Khách hàng, người quản trị |
| Độ ưu tiên | Phải có |
| Điều kiện kích hoạt | Người dùng muốn lấy lại mật khẩu |
| Tiền điều kiện | Tài khoản đã được tạo sẵn  Thiết bị đã được kết nối internet khi thực hiện |
| Kết quả mong đợi | Người dùng lấy lại được mật khẩu |
| Luồng thực hiện | 1. Người dùng mở màn hình đăng nhập và nhấn quên mật khẩu 2. Người dùng điền thông tin cần thiết và nhấn đặt lại mật khẩu 3. Hệ thống xác nhận thông tin tài khoản và gửi lại mật khẩu mới cho người dùng |

### Quản lý thông tin cá nhân

|  |  |
| --- | --- |
| Tên | Quản lý thông tin cá nhân |
| Mô tả | Người dùng quản lý thông tin cá nhân của họ (tên, ảnh đại diện, email, mật khẩu) |
| Tác nhân | Khách hàng, người quản trị |
| Độ ưu tiên | Phải có |
| Điều kiện kích hoạt | Người dùng muốn quản lý thông tin cá nhân |
| Tiền điều kiện | Tài khoản đã đăng nhập vào ứng dụng  Tài khoản đã được phân quyền  Thiết bị đã được kết nối internet khi thực hiện |
| Kết quả mong đợi | Người dùng hoàn tất việc quản lý thông tin cá nhân thành công |
| Luồng thực hiện | 1. Người dùng mở màn hình thông tin cá nhân và nhấn chỉnh sửa 2. Người dùng cập nhật thông tin cần thiết và nhấn lưu 3. Hệ thống lưu lại thông tin và gửi thông báo cho người dùng |

### Tìm kiếm

|  |  |
| --- | --- |
| Tên | Tìm kiếm |
| Mô tả | Tìm kiếm trong ứng dụng (bài hát, thể loại, album, quốc gia, nghệ sĩ, danh sách phát) |
| Tác nhân | Khách hàng, người quản trị |
| Độ ưu tiên | Bình thường |
| Điều kiện kích hoạt | Người dùng muốn tìm kiếm thông tin mình cần trong ứng dụng |
| Tiền điều kiện | Tài khoản đã đăng nhập vào ứng dụng  Tài khoản đã được phân quyền  Thiết bị đã được kết nối internet khi thực hiện |
| Kết quả mong đợi | Người dùng tìm kiếm được thông tin mình cần |
| Luồng thực hiện | 1. Người dùng mở màn hình tìm kiếm 2. Người dùng nhập thông tin cần tìm vào ô tìm kiếm 3. Hệ thống tìm kiếm thông tin và hiển thị lên màn hình cho người dùng |

### Phát bài hát

|  |  |
| --- | --- |
| Tên | Phát bài hát |
| Mô tả | Người dùng phát bài hát từ màn hình bài hát hoặc từ các danh sách phát trong các màn hình khác |
| Tác nhân | Khách hàng, người quản trị |
| Độ ưu tiên | Phải có |
| Điều kiện kích hoạt | Người dùng muốn nghe bài hát |
| Tiền điều kiện | Tài khoản đã đăng nhập vào ứng dụng  Tài khoản đã được phân quyền  Thiết bị đã được kết nối internet khi thực hiện |
| Kết quả mong đợi | Người dùng nghe bài hát được chọn thành công |
| Luồng thực hiện | 1. Người dùng mở màn hình có chứa bài hát cần phát và nhấn vào bài hát 2. Hệ thống tiếp nhận và thực hiện phát bài hát |

### Tải bài hát

|  |  |
| --- | --- |
| Tên | Tải bài hát |
| Mô tả | Người dùng tải bài hát từ màn hình bài hát hoặc từ các danh sách phát trong các màn hình khác |
| Tác nhân | Khách hàng, người quản trị |
| Độ ưu tiên | Bình thường |
| Điều kiện kích hoạt | Người dùng muốn tải bài hát |
| Tiền điều kiện | Tài khoản đã đăng nhập vào ứng dụng  Tài khoản đã được phân quyền  Thiết bị đã được kết nối internet khi thực hiện |
| Kết quả mong đợi | Người dùng tải bài hát được chọn thành công |
| Luồng thực hiện | 1. Người dùng mở màn hình có chứa bài hát cần tải và nhấn tải bài hát 2. Hệ thống tiếp nhận và thực hiện tải bài hát về thiết bị của người dùng |

### Quản lý bình luận

|  |  |
| --- | --- |
| Tên | Quản lý bình luận |
| Mô tả | Người dùng quản lý bình luận của họ (thêm bình luận mới, xóa bình luận, cập nhật bình luận) |
| Tác nhân | Khách hàng, người quản trị |
| Độ ưu tiên | Bình thường |
| Điều kiện kích hoạt | Người dùng muốn quản lý bình luận của họ |
| Tiền điều kiện | Tài khoản đã đăng nhập vào ứng dụng  Tài khoản đã được phân quyền  Thiết bị đã được kết nối internet khi thực hiện |
| Kết quả mong đợi | Người dùng quản lý bình luận của họ thành công |
| Luồng thực hiện | 1. Người dùng mở màn hình phát nhạc và nhấn vào bình luận 2. Người dùng thực hiện cập nhật thông tin bình luận và nhấn lưu 3. Hệ thống lưu lại thông tin và thông báo cho người dùng |

### Quản lý danh sách phát (cá nhân)

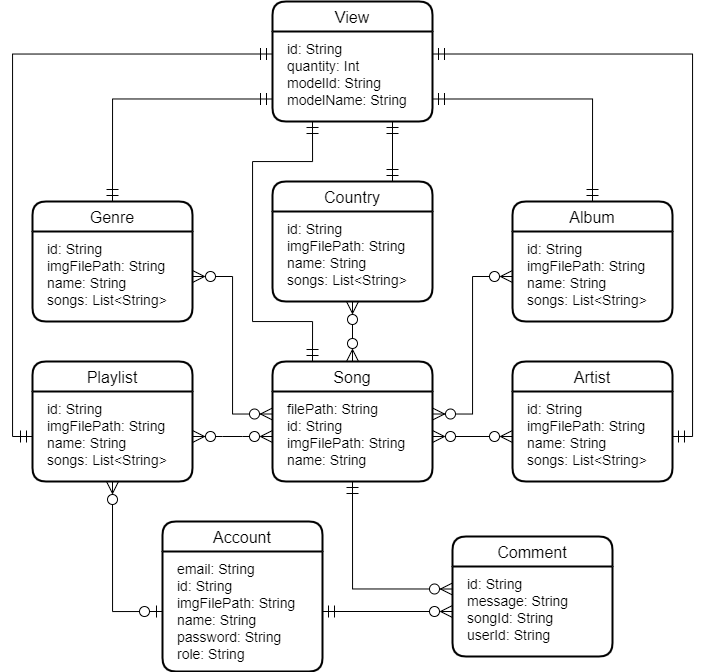
|  |  |
| --- | --- |
| Tên | Quản lý danh sách phát cá nhân |
| Mô tả | Người dùng quản lý danh sách phát của họ (thêm danh sách phát mới, xóa danh sách phát, cập nhật danh sách phát, thêm/xóa bài hát khỏi danh sách phát) |
| Tác nhân | Khách hàng, người quản trị |
| Độ ưu tiên | Phải có |
| Điều kiện kích hoạt | Người dùng muốn quản lý danh sách phát cá nhân |
| Tiền điều kiện | Tài khoản đã đăng nhập vào ứng dụng  Tài khoản đã được phân quyền  Thiết bị đã được kết nối internet khi thực hiện |
| Kết quả mong đợi | Người dùng quản lý danh sách phát của họ thành công |
| Luồng thực hiện | 1. Người dùng mở màn hình danh sách phát 2. Người dùng thực hiện cập nhật thông tin danh sách phát 3. Hệ thống lưu lại thông tin và thông báo cho người dùng |

### Quản lý các thực thể của hệ thống

|  |  |
| --- | --- |
| Tên | Quản lý các thực thể của hệ thống (thể loại, album, bài hát, nghệ sĩ, danh sách phát, quốc gia, bình luận, tài khoản) |
| Mô tả | Với mỗi thực thể của hệ thống đều có thêm mới, xóa, cập nhật thông tin |
| Tác nhân | Người quản trị |
| Độ ưu tiên | Phải có |
| Điều kiện kích hoạt | Người dùng muốn quản lý các thực thể của hệ thống |
| Tiền điều kiện | Tài khoản đã đăng nhập vào ứng dụng  Tài khoản đã được phân quyền  Thiết bị đã được kết nối internet khi thực hiện |
| Kết quả mong đợi | Người dùng quản lý các thực thể của hệ thống thành công |
| Luồng thực hiện | 1. Người dùng mở màn hình quản lý thực thể hệ thống 2. Người dùng thực hiện cập nhật thông tin 3. Hệ thống lưu lại thông tin và thông báo cho người dùng |

## Thiết kế cơ sở dữ liệu

### Trực tuyến



**Hình 2. Mô hình hoá cơ sở dữ liệu trực tuyến**

* View: Chứa thông tin về số lượt xem của người dùng đối với thể loại, quốc gia, album, nghệ sĩ, danh sách phát. Bao gồm id, số lượng, id thực thể và tên thực thể.
* Genre, Country, Album, Playlist, Artist: Chứa thông tin về thực thể tương ứng. Bao gồm id, url file ảnh, tên và danh sách mã các bài hát.
* Song: Chứa thông tin của bài hát. Bao gồm id, tên, url file nhạc, url file ảnh.
* Comment: Chứa thông tin bình luận của bài hát. Bao gồm id, nội dung, id người dùng, id bài hát.
* Account: Chứa thông tin tài khoản có trên hệ thống. Bao gồm id, email, mật khẩu, tên, url file ảnh, vai trò.
* Sử dụng Firebase Storage để lưu file ảnh và file nhạc.

### Ngoại tuyến



**Hình 3. Mô hình hoá cơ sở dữ liệu ngoại tuyến**

* Playlist: Chứa thông tin danh sách phát. Bao gồm id và tên.
* Song: Chứa thông tin bài hát. Bao gồm id, uri file nhạc, tên, nghệ sĩ thực hiện, độ dài, kích thước, đường dẫn file nhạc.
* Playlist\_Song: Chứa thông tin về mối liên kết n-n giữa Playlist và Song. Bao gồm id, id nhạc và id danh sách phát.

## Thiết kế giao diện

Sử dụng XML kết hợp các thành phần giao diện mặc định của Android Studio và tham khảo các mẫu giao diện của Zing Mp3, SoundCloud và từ đó đúc kết để tạo thành giao diện cho ứng dụng.

# XÂY DỰNG ỨNG DỤNG

## Tổ chức ứng dụng

Ứng dụng được xây dựng theo mô hình kiến trúc MVVM và được tổ chức với cây thư mục như sau (cả phần trực tuyến và ngoại tuyến):

* Data: Chứa các lớp thực thể và các lớp thao tác với dữ liệu.
* Di: Chứa các lớp cung cấp các phụ thuộc của đối tượng khác.
* Repositories: Chứa các lớp cung cấp các hàm tương tác với cơ sở dữ liệu.
* Services: Chứa các lớp cung cấp dịch vụ.
* UI: Chứa các lớp thể hiện giao diện.
* ViewModels: Chứa các lớp cung cấp khả năng kết nối giữa Repositories và UI.
* Res: Chứa các file đa phương tiện.

## Xây dựng chức năng

Một số lớp, hàm điều khiển dữ liệu:

* Lớp bài hát

data class OnlineSong(  
 var id: String? = "",  
 var name: String? = "",  
 var imgFilePath: String? = "",  
 var filePath: String? = ""  
) : Serializable

* Hàm lấy dữ liệu tất cả bài hát

override fun getAllSongs(result: (UiState<List<OnlineSong>>) -> Unit) {  
 database  
 .collection(FireStoreCollection.SONG).orderBy("name")  
 .addSnapshotListener **{** value, \_ **->** val songs: ArrayList<OnlineSong> = ArrayList()  
 if (value != null) {  
 for (document in value){  
 val song = document

.toObject(OnlineSong::class.*java*)  
 songs.add(song)  
 }  
 }  
 result.invoke(UiState.Success(songs))  
 **}**}

* Hàm phát nhạc

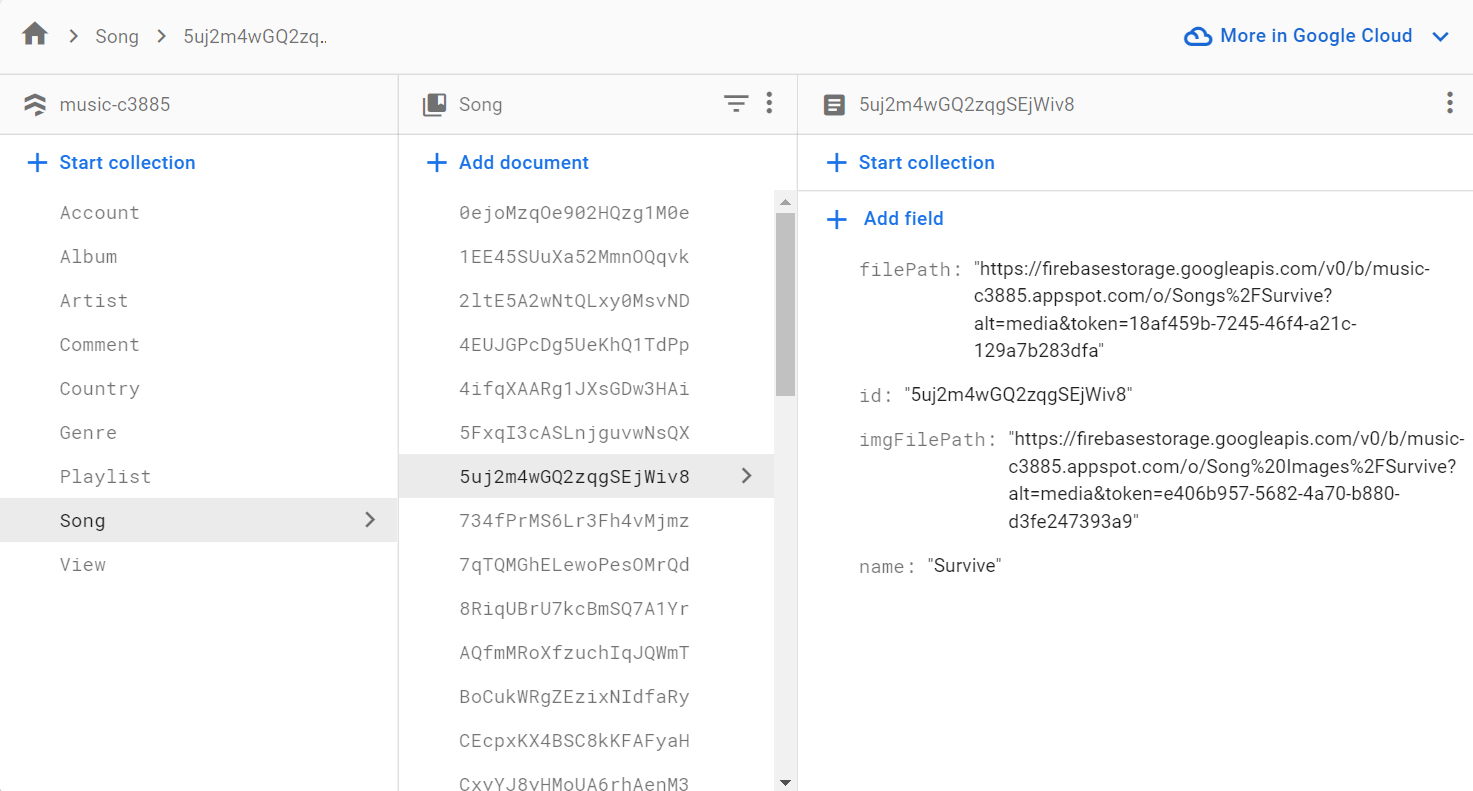
fun createMediaPlayer(song: OnlineSong){  
 currentSong = song  
 mediaPlayer = MediaPlayer()  
 *with*(mediaPlayer!!) **{** setAudioAttributes(  
 AudioAttributes.Builder()  
 .setContentType(AudioAttributes.*CONTENT\_TYPE\_MUSIC*)  
 .setUsage(AudioAttributes.*USAGE\_MEDIA*)  
 .build()  
 )  
 setDataSource(song.filePath)  
 prepare()  
 start()  
 **}** sendNotification(currentSong!!)  
}

* Hàm tìm kiếm bài hát

private fun filterSong(text: String) {  
 val filter: ArrayList<OnlineSong> = ArrayList()  
 for (item in onlineSongAdapter.songList) {  
 if (item.name!!.*lowercase*(Locale.getDefault()).*contains*(text.*lowercase*(Locale.getDefault()))) {  
 filter.add(item)  
 }  
 }  
 if (filter.isEmpty()) {  
 *toast*("Not found")  
 }  
 if (text.*isEmpty*()){  
 onlineSongAdapter.setData(initialList!!)  
 }  
 else {  
 onlineSongAdapter.setData(filter)  
 }  
}

## Cài đặt cơ sở dữ liệu

### Trực tuyến



**Hình 4. Cơ sở dữ liệu trực tuyến**

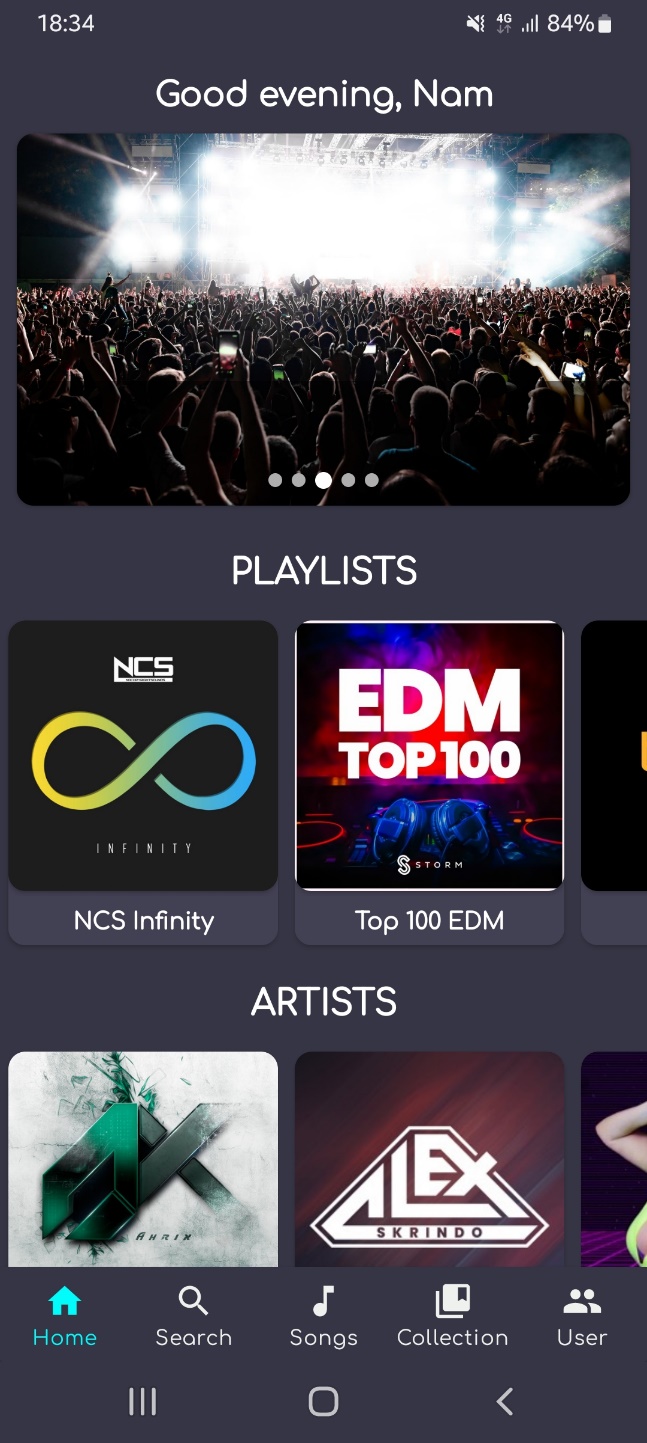
### Ngoại tuyến



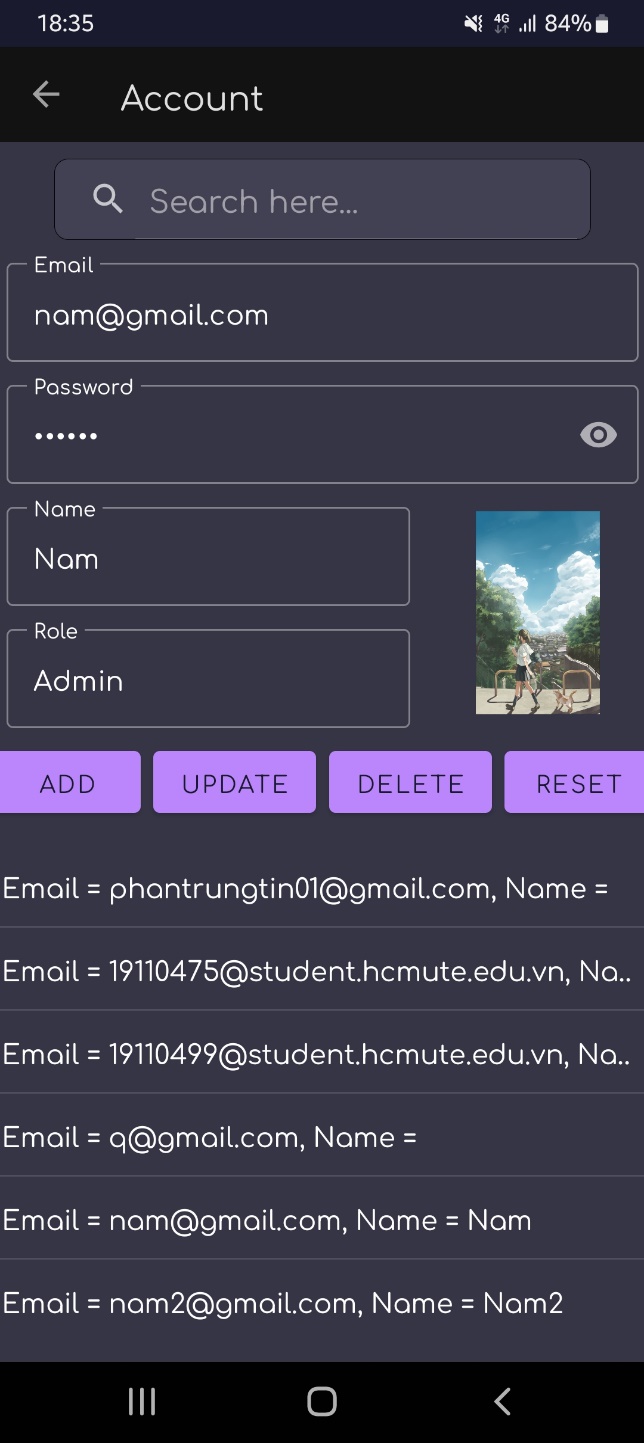
**Hình 5. Cơ sở dữ liệu ngoại tuyến**

## Thiết kế giao diện

Một số màn hình trong ứng dụng



**Hình 6. Màn hình chủ**



**Hình 7. Màn hình quản lý tài khoản (quyền quản trị)**

# TỔNG KẾT

## Kết quả đạt được

Có thêm kiến thức về việc áp dụng các công nghệ đã sử dụng và triển khai các công nghệ mới vào ứng dụng. Đồng thời xây dựng thành công một ứng dụng nghe nhạc trực tuyến với các chức năng cơ bản đã đề ra:

* Hỗ trợ phát nhạc trực tuyến và ngoại tuyến với các bài hát có sẵn trên máy.
* Tìm kiếm và phân loại bài hát theo tên, theo album, theo nghệ sĩ, theo thể loại, theo quốc gia, theo danh sách phát (tuỳ chỉnh hoặc có sẵn).
* Quản lý bài hát, danh sách phát, nghệ sĩ, album, thể loại, quốc gia, thông tin tài khoản, người dùng.
* Đăng nhập, đăng ký, đăng xuất, khôi phục mật khẩu.
* Để lại bình luận cho từng bài hát và tải bài hát về thiết bị.
* Tạo giao diện trực quan và dễ dàng thao tác.
* Bảo vệ thông tin khách hàng, hiệu suất nhanh và dễ dàng thao tác, bảo trì.

## Ưu điểm

Giao diện đơn giản, thân thiện và dễ sử dụng.

Tổ chức ứng dụng rõ ràng, chặt chẽ.

Các chức năng dễ tiếp cận và sử dụng.

Chạy được trên nhiều phiên bản Android.

## Nhược điểm

Ứng dụng chưa có các tính năng nâng cao (sử dụng AI và ML đề xuất cho người dùng dựa trên lịch sử nghe của họ, đăng nhập với tài khoản mạng xã hội phổ biến hiện nay).

Giao diện còn khá phức tạp và không mang tính thẩm mỹ cao.

## Khó khăn

Khó khăn trong công tác kiểm thử.

Khó khăn trong việc tiếp cận và thực hiện các chức năng nâng cao.

## Bài học kinh nghiệm

Biết cách xây dựng một ứng dụng nghe nhạc trực tuyến và hiểu được cách thức vận hành của một ứng dụng di động.

Hiểu hơn về cách thức phối hợp giữa các nền tảng, công nghệ lập trình.

## Hướng phát triển

Cải thiện giao diện ứng dụng trở nên đơn giản và đẹp, trực quan, dễ sử dụng hơn.

Xây dựng thêm phiên bản miễn phí và trả phí cho ứng dụng.

Thêm các tính năng nâng cao như tích hợp AI và Machine Learning trong việc đề xuất cho người dùng thể loại mà họ hay nghe vào ứng dụng.

# TÀI LIỆU THAM KHẢO

[1] https://viblo.asia/p/tim-hieu-ve-mo-hinh-mvvm-maGK7vW95j2

[2] https://viblo.asia/p/cung-tim-hieu-ve-kotlin-coroutines-bWrZnp7Q5xw

[3] https://codelearn.io/sharing/hieu-ro-ve-dependency-injection

[4] https://viblo.asia/p/thay-the-database-cua-ung-dung-bang-room-database-RQqKLvarl7z

[5] https://viblo.asia/p/firebase-realtime-database-cloud-firestore-la-gi-nen-chon-frd-hay-cf-cho-app-cua-ban-63vKj2xdK2R