|  |  |
| --- | --- |
|  |  |
| NGUYỄN TÙNG DƯƠNG | **BỘ CÔNG THƯƠNG**  **TRƯỜNG ĐẠI HỌC CÔNG NGHIỆP HÀ NỘI**  **---------------------------------------** |
|  |
| ĐỒ ÁN TỐT NGHIỆP ĐẠI HỌC |
| NGÀNH CÔNG NGHỆ THÔNG TIN |
| **XÂY DỰNG ỨNG DỤNG NGHE NHẠC IMELODY** |
|  |
|  |
| **CBHD: Ths. Nguyễn Thanh Hải** |
| CÔNG NGHỆ THÔNG TIN | **Sinh viên: Nguyễn Tùng Dương** |
| **Mã số sinh viên: 2020602536** |
|  |
|  |
|  |
| Hà Nội – Năm 2024 |
|  |

|  |
| --- |
| **TRƯỜNG ĐẠI HỌC CÔNG NGHIỆP HÀ NỘI**  **KHOA CÔNG NGHỆ THÔNG TIN**  **---------------------------------------** |
|  |
| **ĐỒ ÁN TỐT NGHIỆP ĐẠI HỌC**  **NGÀNH CÔNG NGHỆ THÔNG TIN**  **XÂY DỰNG ỨNG DỤNG NGHE NHẠC IMELODY** |
| **CBHD:****Ths. Nguyễn Thanh Hải** |
| **Sinh viên: Nguyễn Tùng Dương**  **Mã sinh viên: 2020602536** |
|  |
| **Hà Nội – Năm 2024** |

# MỤC LỤC

[MỤC LỤC i](#_Toc166807181)

[DANH MỤC CÁC KÝ HIỆU, CÁC CHỮ VIẾT TẮT iii](#_Toc166807182)

[DANH MỤC BẢNG BIỂU iv](#_Toc166807183)

[DANH MỤC CÁC HÌNH VẼ, ĐỒ THỊ v](#_Toc166807184)

[LỜI CẢM ƠN 1](#_Toc166807185)

[MỞ ĐẦU 2](#_Toc166807186)

[1. Lý do chọn đề tài 2](#_Toc166807187)

[2. Mục tiêu nghiên cứu 2](#_Toc166807188)

[3. Nội dung nghiên cứu 3](#_Toc166807189)

[4. Bố cục đề tài 3](#_Toc166807190)

[CHƯƠNG 1: TỔNG QUAN VỀ DỰ ÁN 4](#_Toc166807191)

[1.1 Giới thiệu chung 4](#_Toc166807192)

[1.2 Giới thiệu dự án phần mềm 4](#_Toc166807193)

[1.3 Công cụ, kỹ thuật và phương pháp phát triển phần mềm 4](#_Toc166807194)

[1.3.1 Môi trường phát triển tích hợp 4](#_Toc166807195)

[1.3.2 Một số framework và nền tảng hỗ trợ 5](#_Toc166807196)

[CHƯƠNG 2: PHÂN TÍCH THIẾT KẾ HỆ THỐNG 10](#_Toc166807197)

[2.1 Khảo sát hệ thống 10](#_Toc166807198)

[2.1.1 Khảo sát 10](#_Toc166807199)

[2.1.2 Các yêu cầu chức năng 11](#_Toc166807200)

[2.1.3 Các yêu cầu phi chức năng 11](#_Toc166807201)

[2.2 Xây dựng biểu đồ Usecase 12](#_Toc166807202)

[2.2.1 Biểu đồ Usecase tổng quát 12](#_Toc166807203)

[2.2.2 Đặc tả các chức năng 12](#_Toc166807204)

[CHƯƠNG 3: XÂY DỰNG ỨNG DỤNG 40](#_Toc166807205)

[3.1 Biểu đồ Entity Relationship Diagram 40](#_Toc166807206)

[3.2 Các bảng trong cơ sở dữ liệu 40](#_Toc166807207)

[3.3 Kết quả giao diện màn hình 42](#_Toc166807208)

[3.3.1 Giao diện màn hình chính 42](#_Toc166807209)

[3.3.2 Giao diện màn hình yêu thích 44](#_Toc166807210)

[3.3.3 Giao diện màn hình tìm kiếm 44](#_Toc166807211)

[3.3.4 Giao diện màn hình thư viện 45](#_Toc166807212)

[3.3.5 Giao diện màn hình ca sĩ 46](#_Toc166807213)

[3.3.6 Giao diện màn hình đăng xuất 47](#_Toc166807214)

[3.3.7 Giao diện trình phát nhạc 48](#_Toc166807215)

[3.3.8 Giao diện đăng ký 49](#_Toc166807216)

[3.3.9 Giao diện đăng nhập 50](#_Toc166807217)

[3.3.10 Giao diện danh sách nhạc trong thư viện 52](#_Toc166807218)

[3.4 Kế hoạch kiểm thử 54](#_Toc166807219)

[3.5. Xây dựng kịch bản 54](#_Toc166807220)

[3.6 Kết quả kiểm thử 56](#_Toc166807221)

[KẾT LUẬN 57](#_Toc166807222)

[TÀI LIỆU THAM KHẢO 59](#_Toc166807223)

# DANH MỤC CÁC KÝ HIỆU, CÁC CHỮ VIẾT TẮT

|  |  |
| --- | --- |
| CSDL | Cơ sở dữ liệu |
| IDE | Integrated Development Environment |
| UWP | Universal Windows Platform |
| HTML | HyperText Markup Language |
| UI | User Interface |
| OTA | Over the air |
| GPS | Global Positioning System |
| API | Application Programming Interface |

# DANH MỤC BẢNG BIỂU

[Bảng 3.1 Bảng Users 40](#_Toc166797419)

[Bảng 3.2 Bảng Songs 41](#_Toc166797420)

[Bảng 3.3 Bảng Playlists 41](#_Toc166797421)

[Bảng 3.4 Bảng Artists 42](#_Toc166797422)

[Bảng 3.5 Bảng Favorites 42](#_Toc166797423)

# DANH MỤC CÁC HÌNH VẼ, ĐỒ THỊ

[Hình 2.1 Biểu đồ Usecase tổng quát 12](#_Toc166797429)

[Hình 2.2 Biểu đồ Usecase Đăng ký 12](#_Toc166797430)

[Hình 2.3 Biểu đồ trình tự Usecase Đăng ký 14](#_Toc166797431)

[Hình 2.4 Biểu đồ hoạt động Usecase Đăng ký 15](#_Toc166797432)

[Hình 2.5 Biểu đồ Usecase Đăng nhập 15](#_Toc166797433)

[Hình 2.6 Biểu đồ trình tự Usecase Đăng nhập 17](#_Toc166797434)

[Hình 2.7 Biểu đồ hoạt động Usecase Đăng nhập 18](#_Toc166797435)

[Hình 2.8 Biểu đồ Usecase Tìm kiếm 19](#_Toc166797436)

[Hình 2.9 Biểu đồ trình tự Usecase Tìm kiếm 20](#_Toc166797437)

[Hình 2.10 Biểu đồ hoạt động Usecase Tìm kiếm 21](#_Toc166797438)

[Hình 2.11 Biểu đồ Usecase Đăng xuất 22](#_Toc166797439)

[Hình 2.12 Biểu đồ trình tự Usecase Đăng xuất 23](#_Toc166797440)

[Hình 2.13 Biểu đồ hoạt động Usecase Đăng xuất 24](#_Toc166797441)

[Hình 2.14 Biểu đồ hoạt động Usecase Yêu thích 24](#_Toc166797442)

[Hình 2.15 Biểu đồ trình tự Usecase Yêu thích 26](#_Toc166797443)

[Hình 2.16 Biểu đồ hoạt động Usecase Yêu thích 27](#_Toc166797444)

[Hình 2.17 Biểu đồ hoạt động Usecase Chi tiết trình phát nhạc 28](#_Toc166797445)

[Hình 2.18 Biểu đồ trình tự Usecase Chi tiết trình phát nhạc 31](#_Toc166797446)

[Hình 2.19 Biểu đồ hoạt động Usecase Chi tiết trình phát nhạc 32](#_Toc166797447)

[Hình 2.20 Biểu đồ Usecase Phát nhạc 33](#_Toc166797448)

[Hình 2.21 Biểu đồ Usecase Phát nhạc 34](#_Toc166797449)

[Hình 2.22 Biểu đồ hoạt động Usecase phát nhạc 35](#_Toc166797450)

[Hình 2. 23 Biểu đồ Usecase Xem danh sách bài hát 35](#_Toc166797451)

[Hình 2.24 Biểu đồ trình tự Usecase xem danh sách bài hát 36](#_Toc166797452)

[Hình 2.25 Biểu đồ hoạt động Usecase xem danh sách bài hát 37](#_Toc166797453)

[Hình 2.26 Biểu đồ Usecase Thư viện 37](#_Toc166797454)

[Hình 2.27 Biểu đồ trình tự Usecase Thư viện 38](#_Toc166797455)

[Hình 2.28 Biểu đồ hoạt động Usecase Thư viện 39](#_Toc166797456)

[Hình 3.1 Biểu đồ Entity Relationship Diagram 40](#_Toc166797457)

[Hình 3.2 Giao diện màn hình chính 43](#_Toc166797458)

[Hình 3.3 Giao diện màn hình yêu thích 44](#_Toc166797459)

[Hình 3.4 Giao diện màn hình tìm kiếm 45](#_Toc166797460)

[Hình 3.5 Giao diện màn hình thư viện 46](#_Toc166797461)

[Hình 3.6 Giao diện màn hình ca sĩ 47](#_Toc166797462)

[Hình 3.7 Giao diện màn hình đăng xuất 48](#_Toc166797463)

[Hình 3.8 Giao diện trình phát nhạc 49](#_Toc166797464)

[Hình 3.9 Giao diện màn hình đăng ký 50](#_Toc166797465)

[Hình 3.10 Giao diện màn hình đăng nhập 51](#_Toc166797466)

[Hình 3.11Giao diện danh sách nhạc theo thư viện 52](#_Toc166797467)

[Hình 3.12 Giao diện danh sách nhạc theo ca sĩ 53](#_Toc166797468)

# LỜI CẢM ƠN

Đầu tiên, em xin gửi lời cảm ơn đến thầy Nguyễn Thanh Hải đã tận tâm nhiệt tình hướng dẫn, chỉ bảo và giúp đỡ em trong thời gian hoàn thành đề tài “Xây dựng ứng dụng nghe nhạc iMelody”.

Em cũng xin chân thành cảm ơn các thầy cô giáo trong trường Đại học Công Nghiệp Hà Nội nói chung, các thầy cô trong khoa Công Nghệ Thông Tin nói riêng đã dạy dỗ, cho em kiến thức về các môn đại cương cũng như các môn chuyên ngành, giúp em có được cơ sở lý thuyết vững vàng và tạo điều kiện giúp đỡ em trong suốt quá trình học tập.

Tài liệu đồ án này được hoàn thành theo đúng thời gian quy định của nhà trường cũng như của khoa không riêng là sự nỗ lực, cố gắng của em mà còn có sự giúp đỡ, chỉ bảo tận tâm của thầy.

Trong quá trình quá trình làm bài báo cáo đồ án, rất khó tránh khỏi sai sót. Em rất mong nhận được ý kiến góp ý của thầy, cô để em có thể trau dồi, tích lũy học thêm được nhiều kinh nghiệm và sẽ hoàn thành tốt hơn nữa!

Em xin chân thành cảm ơn!

# MỞ ĐẦU

## 1. Lý do chọn đề tài

Trong thời đại của internet phát triển mạnh mẽ, điện thoại thông minh không chỉ là một phương tiện giải trí mà còn là công cụ làm việc và kết nối không thể thiếu. Với mạng lưới internet liên tục mở ra không gian rộng lớn cho việc tiếp cận thông tin và giải trí, việc xây dựng ứng dụng cho điện thoại di động đang trở thành một lĩnh vực có tiềm năng lớn và hứa hẹn sẽ phát triển mạnh mẽ hơn trong tương lai.

Trong làn sóng đa dạng của các ứng dụng di động, hệ điều hành Android nổi lên với sự kết hợp của các công nghệ tiên tiến và tính linh hoạt. Android đã nhanh chóng chiếm lĩnh thị trường và trở thành hệ điều hành phổ biến nhất trong việc sử dụng điện thoại di động.

Với sự bùng nổ của nhu cầu giải trí thông qua điện thoại di động, việc xây dựng ứng dụng nghe nhạc trở thành một lựa chọn logic và hấp dẫn. Ứng dụng nghe nhạc không chỉ đáp ứng nhu cầu giải trí mà còn tạo ra trải nghiệm thú vị và thư giãn cho người dùng. "iMelody" là một dự án nghiên cứu và phát triển với mục tiêu cung cấp một nền tảng nghe nhạc tiện lợi và đa dạng trên nền tảng Android. Bằng cách này, em hy vọng mang lại trải nghiệm âm nhạc tốt nhất cho người dùng, giúp họ thư giãn và tận hưởng những khoảnh khắc đầy ý nghĩa cùng với âm nhạc.

Cùng với sự chỉ bảo tận tình của thầy Nguyễn Thanh Hải em đã hoàn thành ứng dụng này. Trong quá trình phân tích thiết kế không thể tránh khỏi những sai sót mong thầy cô đóng góp ý kiến để ứng dụng được hoàn thiện hơn.

## 2. Mục tiêu nghiên cứu

* Tìm hiểu và sử dụng các công cụ Android Studio, Expo… trong dự án phát triển ứng dụng nghe nhạc iMelody.
* Tìm hiểu về ngôn ngữ Typescript và framework React Native.
* Tìm hiểu về cách quản lí mã nguồn, dịch vụ cung cấp kho lưu trữ mã nguồn.
* Nghiên cứu quy trình phát triển phần mềm và vận dụng trong dự án xây dựng ứng dụng nghe nhạc iMelody.
* Thực hiện phân tích, thiết kế và viết được tài liệu phân tích, tài liệu thiết kế ứng dụng nghe nhạc iMelody.
* Cài đặt và kiểm thử được phần mềm.

## 3. Nội dung nghiên cứu

* Khảo sát hệ thống: Khảo sát các ứng dụng nghe nhạc trực tuyến nổi bật như Soundcloud, Spotify, Apple Music, … qua đó áp dụng để xây dựng một ứng dụng nhạc nghe nhạc của mình.
* Phân tích chức năng hệ thống: Vẽ và phân tích biểu đồ chức năng, biểu đồ tuần tự, biểu đồ lớp
* Phân tích thiết kế cơ sở dữ liệu: Phân tích thiết kế các bảng, kiểu dữ liệu lưu trữ, quan hệ giữa các bảng.
* Phân tích thiết kế chương trình

## 4. Bố cục đề tài

Báo cáo được bố cục thành 3 chương:

* *Chương 1:* Tổng quan về dự án

Giới thiệu về dự án xây dựng ứng dụng nghe nhạc iMelody.

* *Chương 2:* Phân tích thiết kế hệ thống

Dựa trên dữ liệu đã thu thập từ quá trình khảo sát, thực hiện việc vẽ biểu đồ chức năng tổng quát của hệ thống để mô tả chi tiết các chức năng cần có. Đồng thời tạo biểu đồ trình tự để minh họa quy trình hoạt động của các chức năng trong hệ thống. Sử dụng biểu đồ lớp để thiết kế cơ sở dữ liệu phù hợp cho ứng dụng, đảm bảo tính nhất quán và tương thích với các chức năng đã mô tả trước đó.

* *Chương 3:* Xây dựng ứng dụng

# TỔNG QUAN VỀ DỰ ÁN

## Giới thiệu chung

Với sự phát triển mạnh mẽ của internet và những ứng dụng vào đời sống, nhu cầu giải trí và làm việc có kết nối với internet ngày càng cao. Ứng dụng nghe nhạc được tạo ra với mục đích hỗ trợ nhu cầu giải trí của người dùng qua cách dễ dàng sử dụng, thao tác tiện lợi đơn giản.

## Giới thiệu dự án phần mềm

Mong muốn sơ bộ: Có được một ứng dụng cơ bản mong muốn đáp ứng nhu cầu nghe nhạc đơn giản với một số chức năng cơ bản như: Phát nhạc, tìm kiếm bài hát, danh sách nhạc yêu thích, nghe nhạc theo ca sĩ...

Ứng dụng cần có giao diện dễ dùng, hiệu năng tốt, đem lại trải nghiệm tốt với người dùng.

Những khách hàng yêu thích âm nhạc và muốn trải nghiệm toàn vẹn những bài hát một cách dễ dàng, giao diện đơn giản dễ dùng nhưng nhu cầu giải trí mỗi ngày.

## Công cụ, kỹ thuật và phương pháp phát triển phần mềm

### Môi trường phát triển tích hợp

#### Android Studio

Android Studio là một môi trường tích hợp phát triển (Integrated Development Environment – IDE) được phát triển bởi Google dành cho việc phát triển ứng dụng trên nền tảng Android. Nó cung cấp một loạt các công cụ và tính năng để giúp nhà phát triển xây dựng và triển khai ứng dụng Android một cách dễ dàng.

Công cụ này cung cấp một giao diện trực quan và các trình biên dịch, trình gỡ lỗi, trình tạo giao diện người dùng, trình quản lý phiên bản và nhiều công cụ khác. Nó hỗ trợ viết code trong ngôn ngữ Java hoặc Kotlin và tích hợp sẵn các thư viện và công cụ phát triển Android.

Một trong những tính năng quan trọng của công cụ này là khả năng sử dụng Gradle, một hệ thống quản lý dự án mạnh mẽ. Gradle cho phép bạn quản lý phụ thuộc, xây dựng, kiểm thử và đóng gói ứng dụng Android một cách linh hoạt và hiệu quả.

#### Visual Studio Code

Visual Studio Code là một trình soạn thảo mã nguồn được phát triển bởi Microsoft dành cho Windows, Linux và macOS. Nó hỗ trợ chức năng debug, đi kèm với Git, có chức năng nổi bật cú pháp (syntax highlighting), tự hoàn thành mã thông minh, snippets, và cải tiến mã nguồn. Nó cũng cho phép tùy chỉnh, người dùng có thể thay đổi theme, phím tắt, và các tùy chọn khác. Nó miễn phí và là phần mềm mã nguồn mở theo giấy phép MIT, mặc dù bản phát hành của Microsoft là theo giấy phép phần mềm miễn phí.

Visual Studio Code được dựa trên Electron, một nền tảng được sử dụng để triển khai các ứng dụng Node.js máy tính cá nhân chạy trên động cơ bố trí Blink. Mặc dù nó sử dụng nền tảng Electron nhưng phần mềm này không phải là một bản khác của Atom, nó thực ra được dựa trên trình biên tập của Visual Studio Online (tên mã là "Monaco").

### Một số framework và nền tảng hỗ trợ

#### React Native

React Native là một framework mã nguồn mở được sáng tạo bởi Facebook. Nó được sử dụng để phát triển ứng dụng di động Android, iOS, Web và UWP bằng cách cho phép các nhà phát triển sử dụng React cùng với môi trường ứng dụng gốc.

Lịch sử hình thành:

Năm 2012 [Mark Zuckerberg](https://vi.wikipedia.org/wiki/Mark_Zuckerberg) đã phát biểu, "Sai lầm lớn nhất của chúng tôi khi làm công ty là dựa trên quá nhiều HTML hơn là môi trường phát triển gốc". Ông hứa rằng Facebook sẽ sớm cung cấp trải nghiệm di động tốt hơn.

Kỹ sư Jordan Walke tại Facebook đã tìm ra cách xây dựng các thành phần [UI](https://vi.wikipedia.org/wiki/Giao_di%E1%BB%87n_ng%C6%B0%E1%BB%9Di_d%C3%B9ng) cho iOS bằng một luồng [JavaScript](https://vi.wikipedia.org/wiki/JavaScript). Họ quyết định tổ chức cuộc thi [Hackathon](https://vi.wikipedia.org/wiki/Hackathon) để hoàn thiện [nguyên mẫu hệ thống](https://vi.wikipedia.org/wiki/Nguy%C3%AAn_m%E1%BA%ABu_h%E1%BB%87_th%E1%BB%91ng) để có thể xây dựng các ứng dụng di động gốc (native app) bằng công nghệ này.

Sau nhiều tháng phát triển, [Facebook](https://vi.wikipedia.org/wiki/Facebook) đã phát hành phiên bản đầu tiên cho React Native vào năm 2015. Trong một cuộc hội thảo công nghệ, Christopher Chedeau cho biết [Facebook](https://vi.wikipedia.org/wiki/Facebook) đã sử dụng React Native trong phát triển ứng dụng nhóm và ứng dụng quản lí quảng cáo của họ.

Ưu điểm:

1. **Tích hợp dễ dàng**: React Native có thể tích hợp dễ dàng vào dự án hiện có sử dụng các thư viện JavaScript phổ biến như Redux hoặc MobX.
2. **Phát triển đa nền tảng**: Với React Native, bạn chỉ cần viết mã một lần và có thể triển khai ứng dụng trên cả iOS và Android.
3. **Cộng đồng lớn và hỗ trợ mạnh mẽ**: React Native có một cộng đồng lớn, đồng thời nhận được sự hỗ trợ từ Facebook và các công ty công nghệ khác.
4. **Hiệu suất tốt**: React Native sử dụng một số kỹ thuật để cải thiện hiệu suất, bao gồm cơ chế trình điều khiển nền (background thread) cho các tác vụ nặng.
5. **Cập nhật nhanh chóng**: Bạn có thể cập nhật ứng dụng một cách nhanh chóng bằng cách sử dụng cơ chế live reload hoặc hot reload trong quá trình phát triển.

Nhược điểm:

1. **Hiệu suất không nhất quán**: Mặc dù hiệu suất của React Native khá tốt, nhưng trong một số trường hợp, ứng dụng có thể không hoạt động mượt mà như ứng dụng native.
2. **Phụ thuộc vào bên thứ ba**: Đôi khi, việc phụ thuộc vào các thư viện và module bên thứ ba có thể tạo ra vấn đề về sự ổn định và bảo trì cho ứng dụng của bạn.
3. **Giới hạn về đa nền tảng**: Mặc dù React Native cho phép viết mã một lần và chạy trên nhiều nền tảng, nhưng vẫn có một số yếu tố có thể gây ra sự không nhất quán giữa các nền tảng.
4. **Khả năng tùy chỉnh hạn chế**: Trong một số trường hợp, việc tùy chỉnh giao diện người dùng có thể khó khăn hơn so với phát triển native.
5. **Kích thước ứng dụng lớn**: Ứng dụng React Native có thể có kích thước lớn hơn so với ứng dụng native do việc đóng gói nhiều mã JavaScript.

#### Expo

Expo là một công cụ mã nguồn mở và miễn phí được xây dựng xung quanh React Native để giúp bạn xây dựng các dự án iOS và Android bằng JavaScript và React.

Ưu điểm:

1. **Tích hợp dễ dàng**: Expo cung cấp một trải nghiệm phát triển dễ dàng và nhanh chóng. Bạn có thể bắt đầu phát triển ứng dụng mà không cần cài đặt và cấu hình môi trường phát triển địa phương.
2. **Cung cấp các API hữu ích**: Expo cung cấp các API hỗ trợ cho việc truy cập vào các tính năng thiết bị như camera, GPS, đèn flash, và nhiều tính năng khác mà không cần phải viết mã native.
3. **Cộng đồng lớn và hỗ trợ tốt**: Expo có một cộng đồng lớn và hoạt động tích cực, với tài liệu phong phú và sự hỗ trợ từ cộng đồng.
4. **Cập nhật OTA (Over-The-Air)**: Expo cho phép cập nhật ứng dụng của bạn "over-the-air" mà không cần phải tải xuống một bản cập nhật mới từ cửa hàng ứng dụng.
5. **Được quản lý bởi Expo team**: Việc có một nhóm chuyên nghiệp quản lý và phát triển Expo giúp đảm bảo tính ổn định và bảo trì cho các dự án.

Nhược điểm:

1. **Giới hạn về tùy chỉnh**: Expo cung cấp một số hạn chế về tùy chỉnh so với việc sử dụng React Native "nguyên bản". Điều này có thể là một vấn đề nếu bạn cần các tính năng tùy chỉnh hoặc tích hợp với các thư viện không được hỗ trợ bởi Expo.
2. **Phụ thuộc vào dịch vụ bên thứ ba**: Một số tính năng và dịch vụ trong Expo có thể phụ thuộc vào các dịch vụ bên thứ ba, điều này có thể là một rủi ro cho việc phát triển và bảo trì ứng dụng trong tương lai.
3. **Giới hạn về quyền truy cập thiết bị**: Mặc dù Expo cung cấp một số API cho việc truy cập vào các tính năng thiết bị, nhưng có một số giới hạn về quyền truy cập so với việc phát triển native.

#### Firebase

Firebase là dịch vụ cơ sở dữ liệu hoạt động trên nền tảng đám mây – cloud. Kèm theo đó là hệ thống máy chủ cực kỳ mạnh mẽ của Google. Chức năng chính là giúp người dùng lập trình ứng dụng bằng cách đơn giản hóa các thao tác với cơ sở dữ liệu.

Cụ thể là những giao diện lập trình ứng dụng API đơn giản. Mục đích nhằm tăng số lượng người dùng và thu lại nhiều lợi nhuận hơn.

Đặc biệt, còn là dịch vụ đa năng và bảo mật cực tốt. Firebase hỗ trợ cả hai nền tảng Android và IOS. Không có gì khó hiểu khi nhiều lập trình viên chọn Firebase làm nền tảng đầu tiên để xây dựng ứng dụng cho hàng triệu người dùng trên toàn thế giới

Lịch sử hình thành:

Gần một thập niên trước, Firebase ra đời với tiền thân là Envolve. Đây là một nền tảng đơn giản chuyên cung cấp những API cần thiết để tích hợp tính năng chat vào trang web. Bên cạnh ứng dụng nhắn tin trực tuyến, Envolve còn được người dùng sử dụng để truyền và đồng bộ hóa dữ liệu cho những ứng dụng khác như các trò chơi trực tuyến… Do đó, các nhà sáng lập đã tách biệt hệ thống nhắn tin trực tuyến và đồng bộ dữ liệu thời gian thực thành hai phần riêng biệt.

Trên cơ sở đó, năm 2012, Firebase ra đời với sản phẩm cung cấp là dịch vụ Backend-as-a-Service. Tiếp đến, vào năm 2014, Google mua lại Firebase và phát triển nó thành một dịch vụ đa chức năng được hàng triệu người sử dụng cho đến hiện nay.

# PHÂN TÍCH THIẾT KẾ HỆ THỐNG

## Khảo sát hệ thống

### Khảo sát

Phương pháp khảo sát: Sử dụng phiếu khảo sát cho người dùng để ghi nhận mong muốn của người dùng đối với một số chức năng và giao diện của ứng dụng.

Phiếu khảo sát:

Người khảo sát: Nguyễn Tùng Dương

Người được khảo sát: Lê Tuấn Minh

|  |  |
| --- | --- |
| Câu hỏi | Câu trả lời |
| Bạn thường nghe nhạc ở đâu? Thời gian trung bình mỗi lần nghe nhạc của bạn? | Trên điện thoại di động, mỗi lần nghe nhạc thường là 1-2 tiếng. |
| Bạn thấy trải nghiệm của mình như thế nào? (Tốt/Rất tốt/Kém/Bình thường/Rất kém) | Bình thường |
| Các chức năng bạn mong muốn trong ứng dụng nghe nhạc? | Khả năng tìm kiếm nâng cao: theo bài hát, ca sĩ, danh sách phát.  Thư viện cá nhân.  Equalizer đa dạng: Có thể điều chỉnh âm lượng, cân bằng âm thanh và hiệu ứng âm thanh cho phù hợp với sở thích cá nhân.  Tự quản lý danh sách phát nhạc của mình.  Phát nhạc ngẫu nhiên theo danh sách.  Thêm các bài hát yêu thích vào danh sách riêng. |
| Bạn có yêu cầu gì về giao diện của ứng dụng không? | Giao diện bắt mắt, đơn giản, dễ dùng, dễ tiếp cận, nhất quán. |

### Các yêu cầu chức năng

Các yêu cầu chức năng gồm:

* Chức năng đăng ký.
* Chức năng đăng nhập.
* Chức năng đăng xuất
* Chức năng tìm kiếm bài hát theo tên.
* Chức năng phân loại và tìm kiếm bài hát theo ca sĩ.
* Chức năng phát nhạc
* Chức năng xem chi tiết trình phát nhạc.
* Chức năng yêu thích.
* Chức năng hiển thị danh sách bài hát.
* Chức năng phát nhạc ngẫu nhiên.
* Chức năng xem và thêm vào, xóa khỏi thư viện.

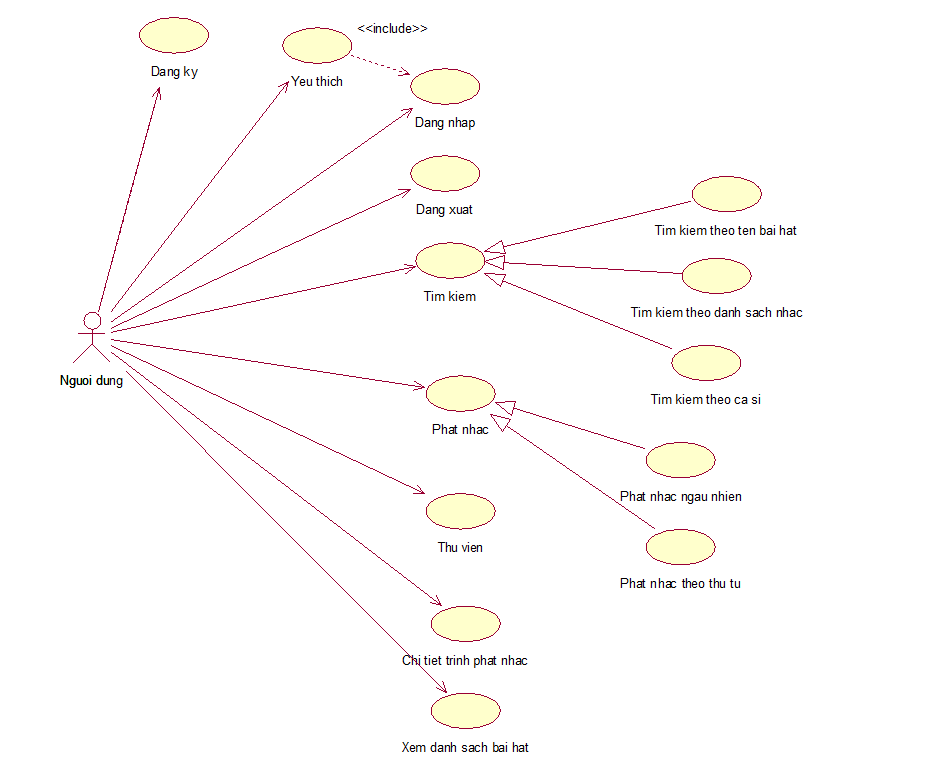
### Các yêu cầu phi chức năng

Giao diện ứng dụng thân thiện, hài hòa, dễ sử dụng, hình ảnh, màu sắc hài hòa, tạo sự thoải mái, dễ chịu.

Khả năng bảo trì, quản lý tốt.

## 2.2 Xây dựng biểu đồ Usecase

### 2.2.1 Biểu đồ Usecase tổng quát



Hình 2.1 Biểu đồ Usecase tổng quát

### 2.2.2 Đặc tả các chức năng

#### 2.2.2.1 Usecase Đăng ký

##### Biểu đồ Usecase Đăng ký



Hình 2.2 Biểu đồ Usecase Đăng ký

##### Đặc tả chi tiết Usecase Đăng ký

Mô tả vắn tắt: Use case cho phép khách đăng ký tài khoản trên hệ thống.

Luồng sự kiện:

+ Luồng cơ bản:

1. Use case này bắt đầu khi khách hàng kích vào nút “Don’t have account yet?” trên màn hình. Hệ thống sẽ hiển thị màn hình đăng ký.
2. Khách hàng nhập thông tin trên form đăng ký gồm email, mật khẩu, xác nhận mật khẩu rồi kích nút “Sign Up”. Hệ thống thêm một bản ghi mới vào trong bảng USERS trong cơ sở dữ liệu.
3. Use case kết thúc.

+ Luồng rẽ nhánh:

1. Tại bước 2 trong luồng cơ bản, khi nhập thông tin không hợp lệ, hệ thống thông báo lỗi. Use case kết thúc.
2. Tại bất kỳ thời điểm nào trong quá trình thực hiện use case nếu không kết nối được với cơ sở dữ liệu thì hệ thống sẽ hiển thị thông báo lỗi và use case kết thúc.

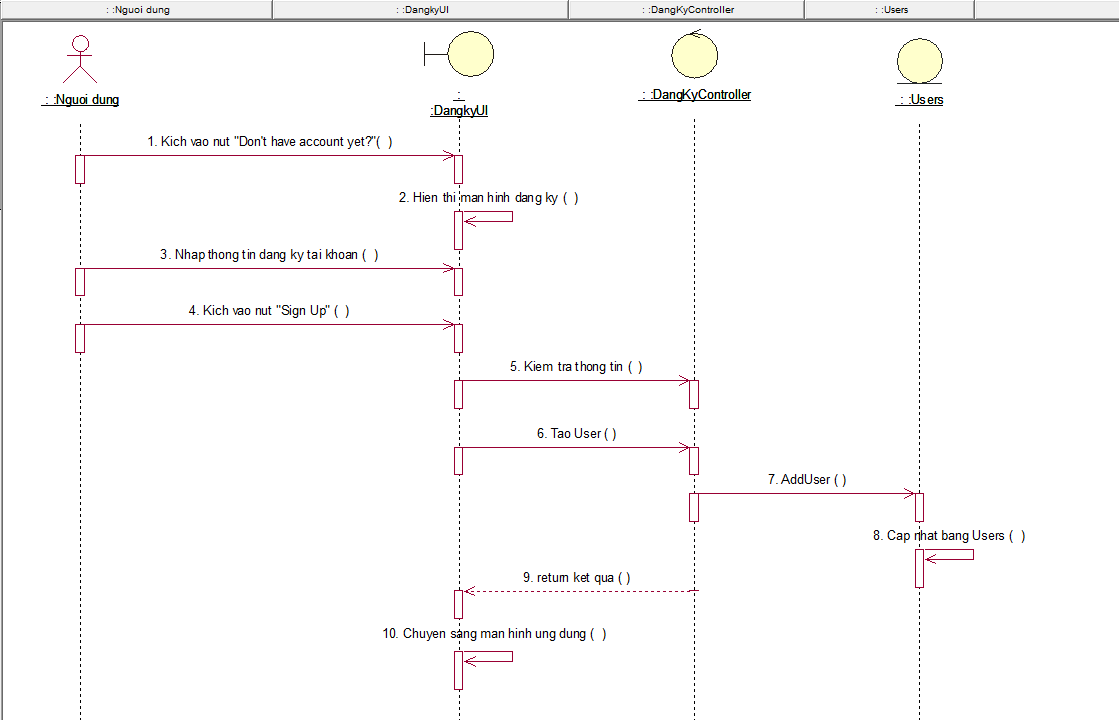
Các yêu cầu đặc biệt: Không có.

Tiền điều kiện: Không có.

Hậu điều kiện: Không có.

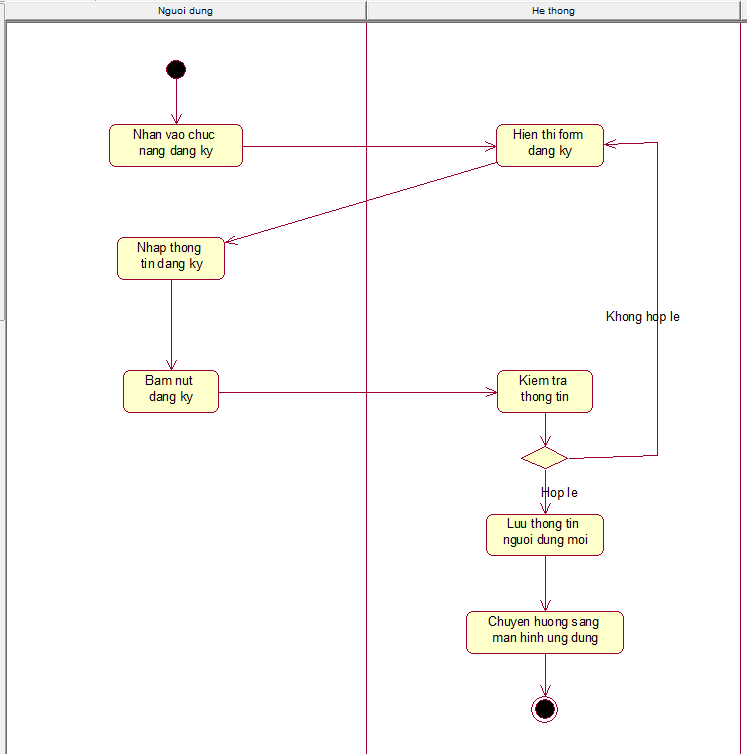
Điểm mở rộng: Không có.

##### Biểu đồ trình tự



Hình 2.3 Biểu đồ trình tự Usecase Đăng ký

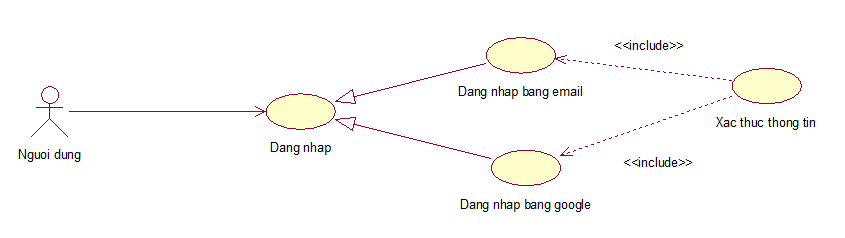
##### Biểu đồ hoạt động



Hình 2.4 Biểu đồ hoạt động Usecase Đăng ký

#### 2.2.2.2 Usecase Đăng nhập

##### Biểu đồ Usecase Đăng nhập



Hình 2.5 Biểu đồ Usecase Đăng nhập

##### Đặc tả Usecase Đăng nhập

Mô tả vắn tắt: Use case này cho phép người dùng đăng nhập để xác định quyền truy cập vào hệ thống.

Luồng sự kiện:

+ Luồng cơ bản:

1. Use case này bắt đầu khi người dùng bấm vào nút “Sign In” ở giữa màn hình.
2. Hệ thống yêu cầu người dùng nhập các thông tin bao gồm: email và mật khẩu.
3. Người dùng nhập các thông tin: email và mật khẩu và bấm nút “Sign In”.
4. Hệ thống kiểm tra tên đăng nhập và mật khẩu đã nhập và chuyển hướng sang trang chủ.
5. Use case kết thúc.

+ Luồng rẽ nhánh:

1. Sai tên đăng nhập hoặc mật khẩu: tại bước 3 trong luồng cơ bản nếu người dùng nhập tên đăng nhập hoặc mật khẩu sai, hệ thống hiển thị thông báo lỗi. Người dùng có thể nhập lại hoặc bỏ qua thao tác, khi đó use case kết thúc.
2. Đăng ký: tại bước 3 trong luồng cơ bản, nếu người dùng bấm nút “Sign Up” use case kết thúc.
3. Đăng nhập bằng google: tại bước 3 trong luồng cơ bản, nếu người dùng bấm biểu tượng “Google” người dùng sẽ chuyển hướng sang đăng nhập bằng google. Use case kết thúc.
4. Tại bất kỳ thời điểm nào trong quá trình thực hiện use case nếu không kết nối được với cơ sở dữ liệu thì hệ thống sẽ hiển thị thông báo lỗi và use case kết thúc.

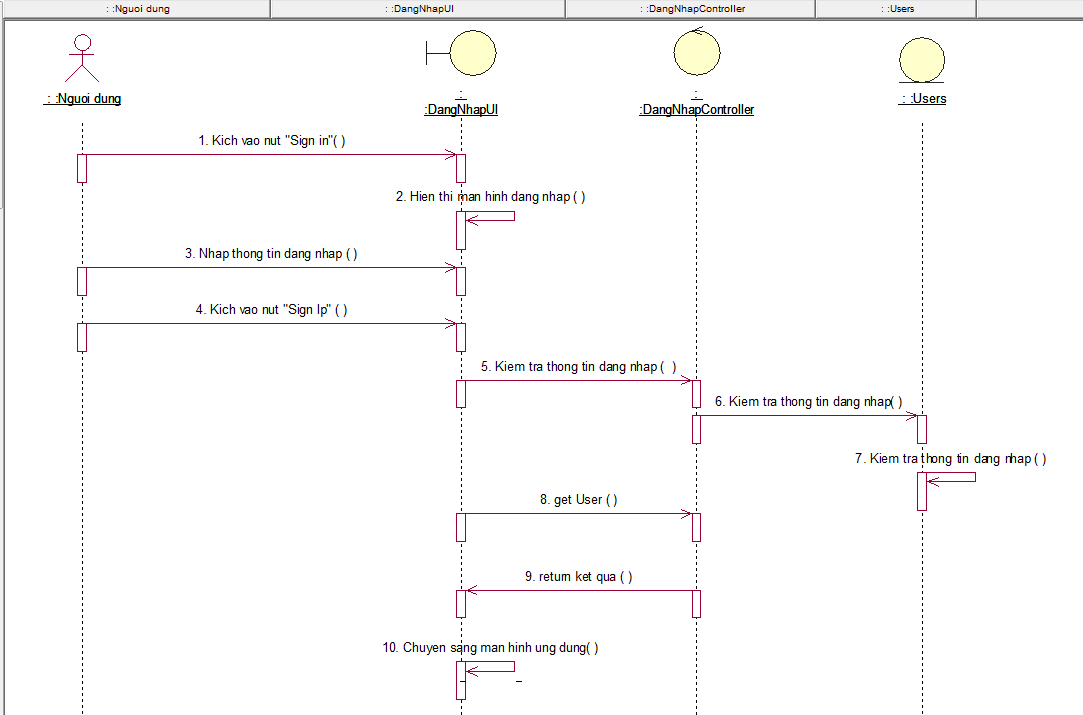
Các yêu cầu đặc biệt: Không có.

Tiền điều kiện: Không có.

Hậu điều kiện: Nếu use case thành công, người dùng sẽ đăng nhập được vào hệ thống. Nếu không trạng thái của hệ thống không thay đổi.

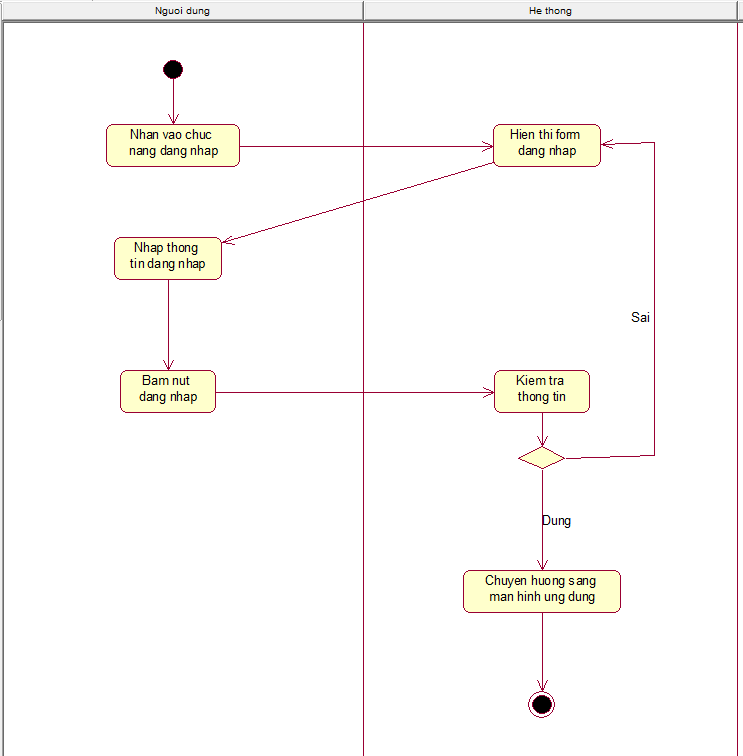
Điểm mở rộng: Không có.

##### Biểu đồ trình tự



Hình 2.6 Biểu đồ trình tự Usecase Đăng nhập

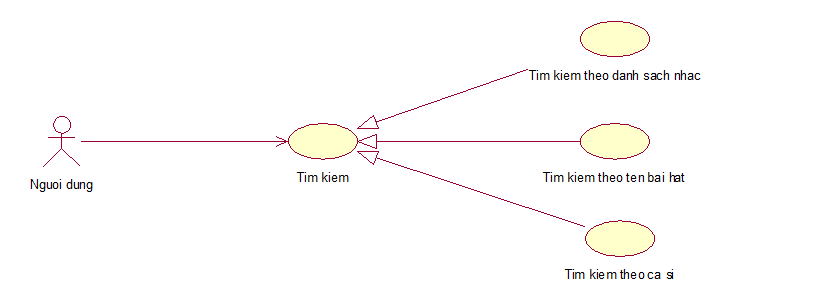
##### Biểu đồ hoạt động



Hình 2.7 Biểu đồ hoạt động Usecase Đăng nhập

#### 2.2.2.3 Usecase Tìm kiếm

##### Biểu đồ Usecase Tìm kiếm



Hình 2.8 Biểu đồ Usecase Tìm kiếm

##### Đặc tả chi tiết Usecase Tìm kiếm

Mô tả vắn tắt: Use case cho phép người dùng tìm kiếm bài hát.

Luồng sự kiện:

+ Luồng cơ bản:

1. Use case này bắt đầu khi khách hàng kích vào ô tìm kiếm.
2. Hệ thống yêu cầu người dùng nhập nội dung cần tìm kiếm.
3. Người dùng nhập nội dung cần tìm kiếm.
4. Hệ thống sẽ lấy danh sách những nội dung liên quan đến từ khóa tìm kiếm và hiển thị ra màn hình.
5. Use case kết thúc.

+ Luồng rẽ nhánh:

1. Tại bất kỳ thời điểm nào trong quá trình thực hiện use case nếu không kết nối được với cơ sở dữ liệu thì hệ thống sẽ hiển thị một thông báo lỗi và use case kết thúc.

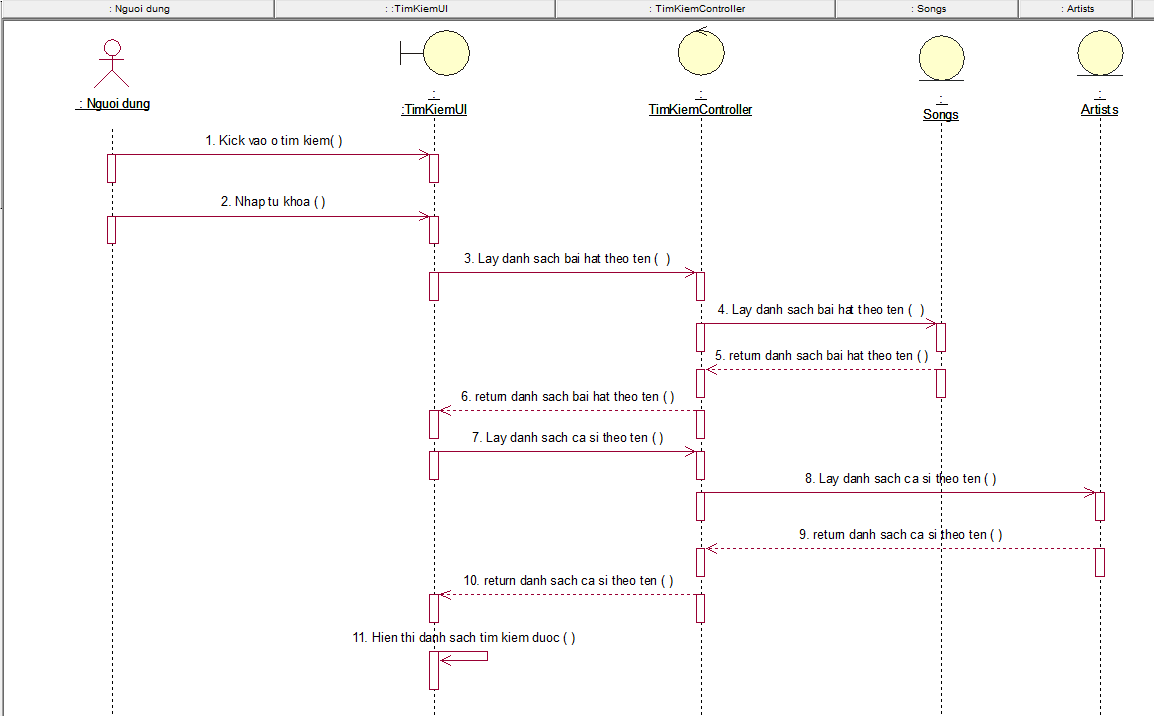
Các yêu cầu đặc biệt: Không có

Tiền điều kiện: Không có

Hậu điều kiện: Không có

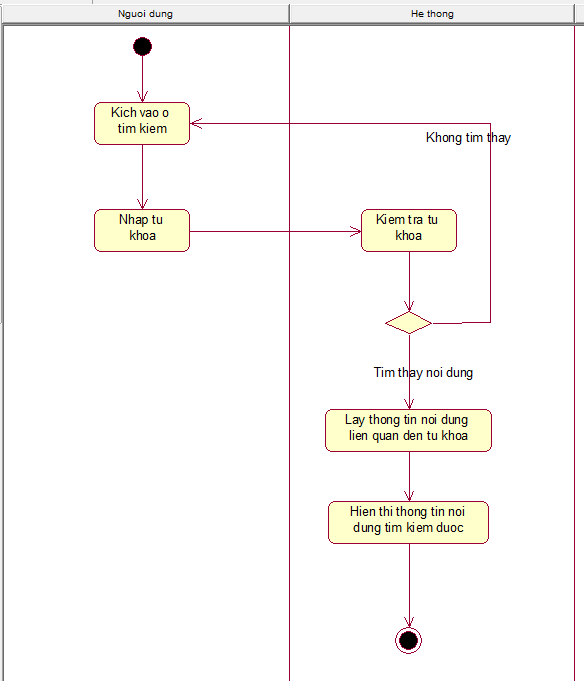
Điểm mở rộng: Không có

##### Biểu đồ trình tự



Hình 2.9 Biểu đồ trình tự Usecase Tìm kiếm

##### Biểu đồ hoạt động



Hình 2.10 Biểu đồ hoạt động Usecase Tìm kiếm

#### 2.2.2.4 Usecase Đăng xuất

##### Biểu đồ Usecase Đăng xuất



Hình 2.11 Biểu đồ Usecase Đăng xuất

##### Đặc tả chi tiết Usecase Đăng xuất

Mô tả vắn tắt: Use case cho phép người dùng đăng xuất tài khoản khỏi hệ thống.

Luồng sự kiện:

+ Luồng cơ bản:

1. Use case này bắt đầu khi khách hàng kích vào nút đăng xuất trên màn hình. Hệ thống sẽ hiển thị màn hình đăng nhập.
2. Use case kết thúc.

+ Luồng rẽ nhánh:

1. Tại bất kỳ thời điểm nào trong quá trình thực hiện use case nếu không kết nối được với cơ sở dữ liệu thì hệ thống sẽ hiển thị thông báo lỗi và use case kết thúc.

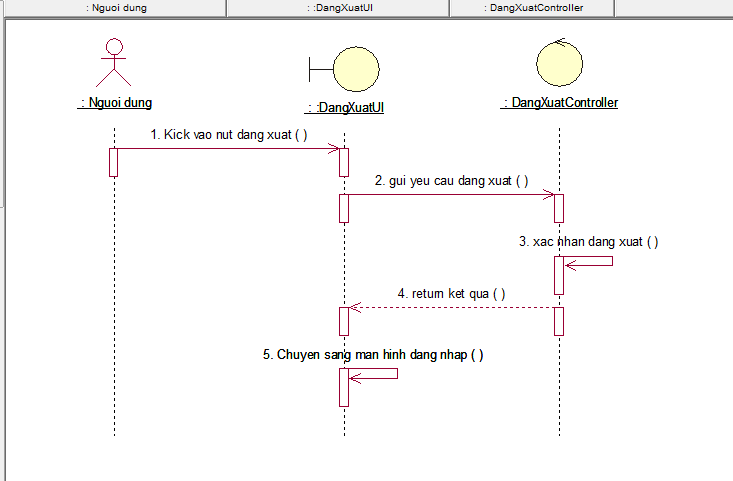
Các yêu cầu đặc biệt: Không có.

Tiền điều kiện: Không có.

Hậu điều kiện: Không có.

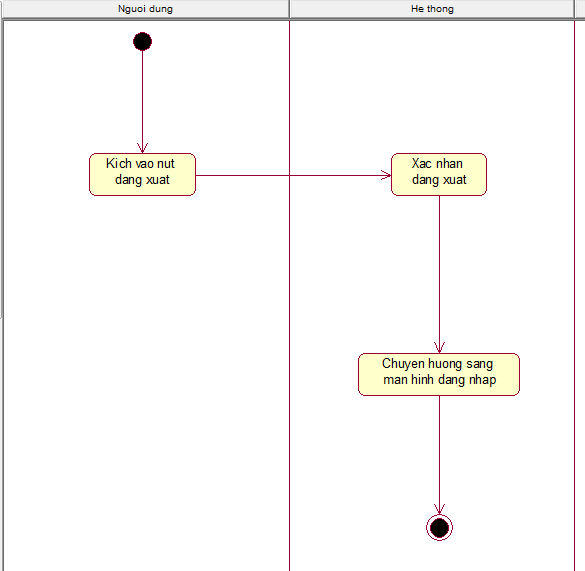
Điểm mở rộng: Không có.

##### Biểu đồ trình tự



Hình 2.12 Biểu đồ trình tự Usecase Đăng xuất

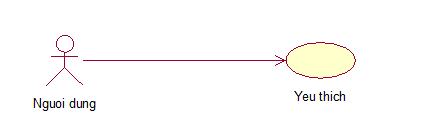
##### Biểu đồ hoạt động



Hình 2.13 Biểu đồ hoạt động Usecase Đăng xuất

#### 2.2.2.5 Usecase Yêu thích

##### Biểu đồ Usecase Yêu thích



Hình 2.14 Biểu đồ hoạt động Usecase Yêu thích

##### Đặc tả chi tiết

Mô tả vắn tắt: Use case cho phép người dùng thêm các bài hát yêu thích vào danh sách riêng.

Luồng sự kiện:

+ Luồng cơ bản:

1. Use case này bắt đầu khi người dùng chọn biểu tượng yêu thích trong trình phát nhạc.
2. Hệ thống lấy thông tin của bài hát, cập nhật thông tin yêu thích và hiển thị trong danh sách yêu thích. Hệ thống thêm một bản ghi mới vào bảng FAVORITES trong cơ sở dữ liệu.
3. Use case kết thúc.

+ Luồng rẽ nhánh:

1. Tại bất kỳ thời điểm nào trong quá trình thực hiện use case nếu không kết nối được với cơ sở dữ liệu thì hệ thống sẽ hiển thị thông báo lỗi và use case kết thúc.
2. Tại bước 1 trong luồng cơ bản, khi bài hát đã trong danh sách yêu thích rồi, hệ thống sẽ xóa bài hát khỏi danh sách yêu thích. Use case kết thúc

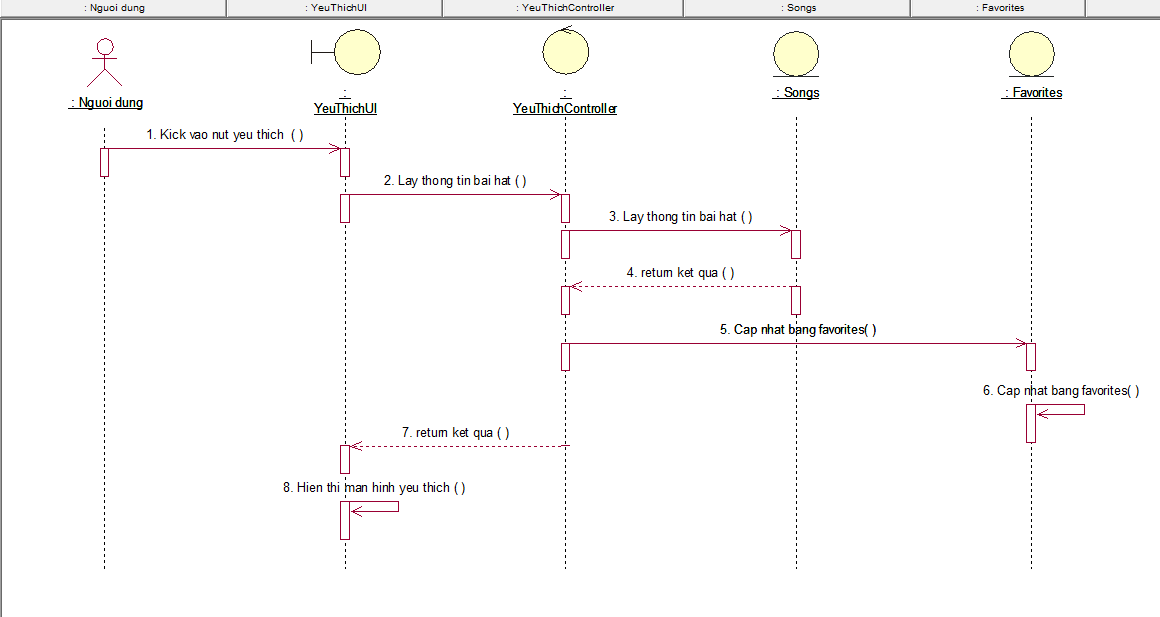
Các yêu cầu đặc biệt: Không có.

Tiền điều kiện: Không có.

Hậu điều kiện: Không có.

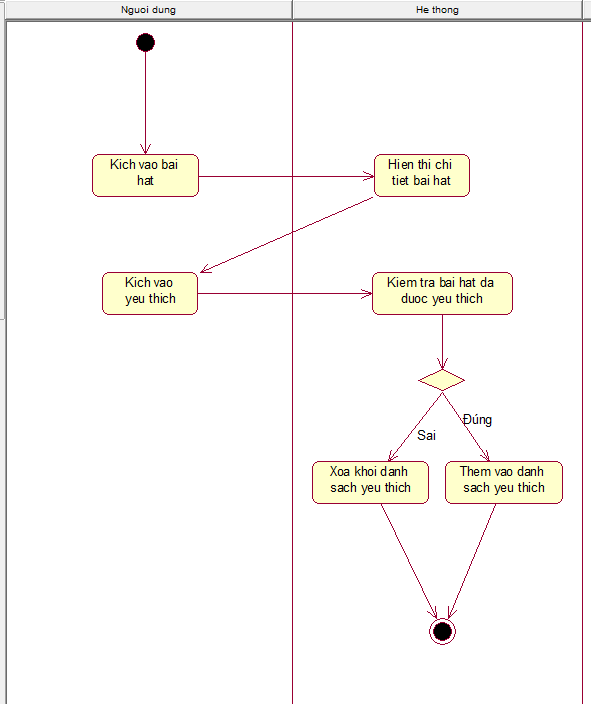
Điểm mở rộng: Không có.

##### Biểu đồ trình tự



Hình 2.15 Biểu đồ trình tự Usecase Yêu thích

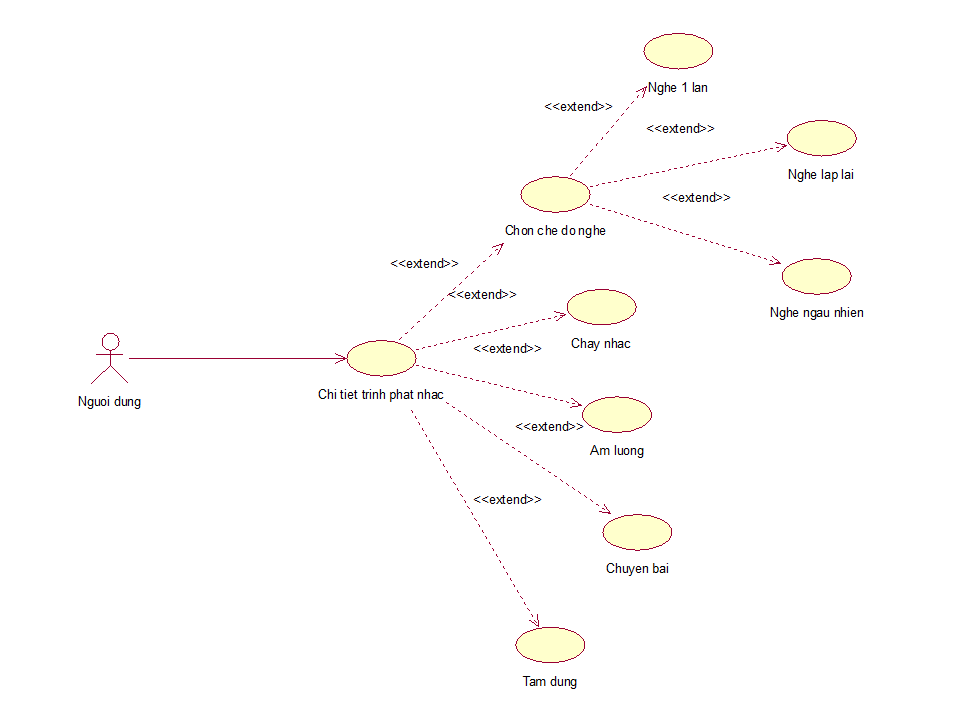
##### Biểu đồ hoạt động



Hình 2.16 Biểu đồ hoạt động Usecase Yêu thích

#### 2.2.2.6 Usecase Chi tiết trình phát nhạc

##### Biểu đồ Usecase Chi tiết trình phát nhạc



Hình 2.17 Biểu đồ hoạt động Usecase Chi tiết trình phát nhạc

##### Đặc tả chi tiết

Mô tả vắn tắt: Use case cho phép người dùng xem chi tiết trình phát nhạc, chọn chế độ nghe, chỉnh âm lượng, điều hướng bài hát.

Luồng sự kiện:

+ Luồng cơ bản:

1. Use case này bắt đầu khi khách hàng kích vào bài hát đang phát trên màn hình. Hệ thống lấy thông tin chi tiết bài hát từ bảng SONGS và hiển thị lên màn hình.
2. Điều hướng âm nhạc:
   1. Người dùng kích vào biểu tượng nút “Next” trên màn hình. Hệ thống lấy thông tin bài hát tiếp theo từ bảng SONGS và điều hướng đến bài hát tiếp theo.
   2. Người dùng kích vào biểu tượng nút “Previous” trên màn hình. Hệ thống lấy thông tin bài nhạc trước đó từ bảng SONGS và điều hướng đến bài hát tiếp theo.
3. Trạng thái âm nhạc:
   1. Người dùng kích vào biểu tượng nút “Pause” trên màn hình. Hệ thống lấy trạng thái bài hát và dừng bài hát đang phát.
   2. Người dùng kích vào biểu tượng nút “Play” trên màn hình. Hệ thống lấy trạng thái bài hát và phát bài hát.
   3. Người dùng kích vào thanh âm lượng trên màn hình. Hệ thống lấy trạng thái bài hát và tăng hoặc giảm âm lượng.
   4. Người dùng kích vào biểu tượng chế độ chơi trên màn hình. Hệ thống kiểm tra trạng thái chế độ chơi để thay đổi chế độ chơi thành lặp lại 1 bài, lặp lại ngẫu nhiên hoặc phát 1 lần.
   5. Người dùng kích vào biểu tượng yêu thích trên màn hình. Hệ thống kiểm tra trạng thái yêu thích của bài hát để thêm hoặc xóa bài hát khỏi danh sách yêu thích.
4. Use case kết thúc

+ Luồng rẽ nhánh:

1. Tại bất kỳ thời điểm nào trong quá trình thực hiện use case nếu không kết nối được với cơ sở dữ liệu thì hệ thống sẽ hiển thị thông báo lỗi và use case kết thúc.
2. Tại bước 2 của luồng cơ bản, nếu người dùng kích vào nút “Back” sẽ trở về màn hình chính. Use case kết thúc.

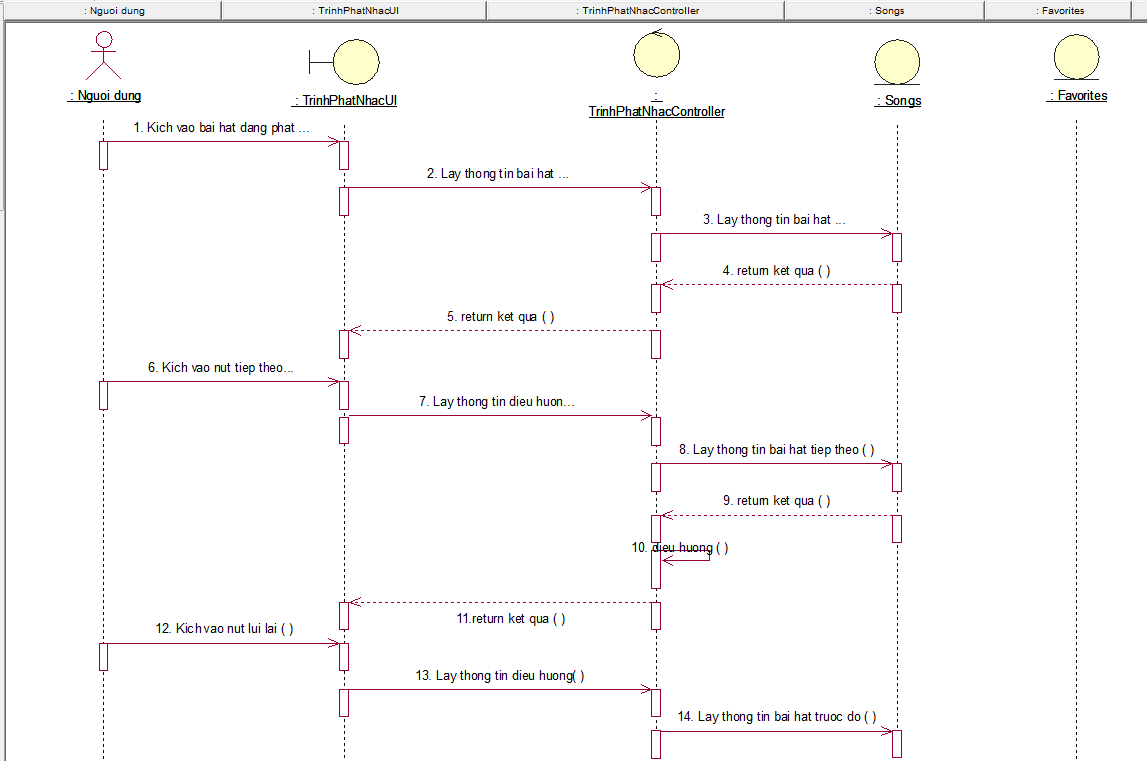
Các yêu cầu đặc biệt: Không có.

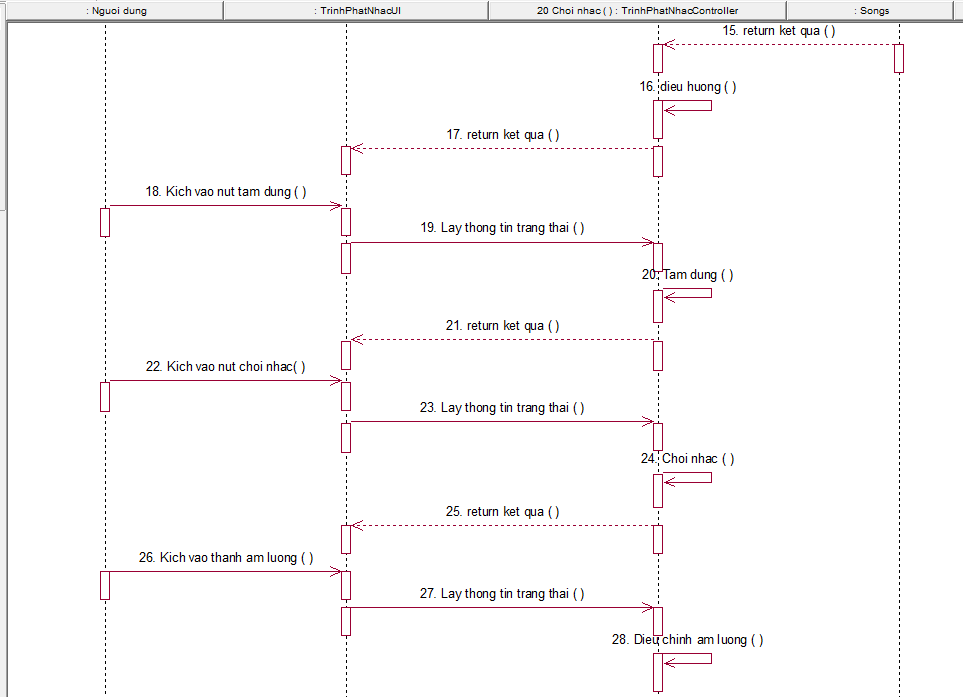
Tiền điều kiện: Không có.

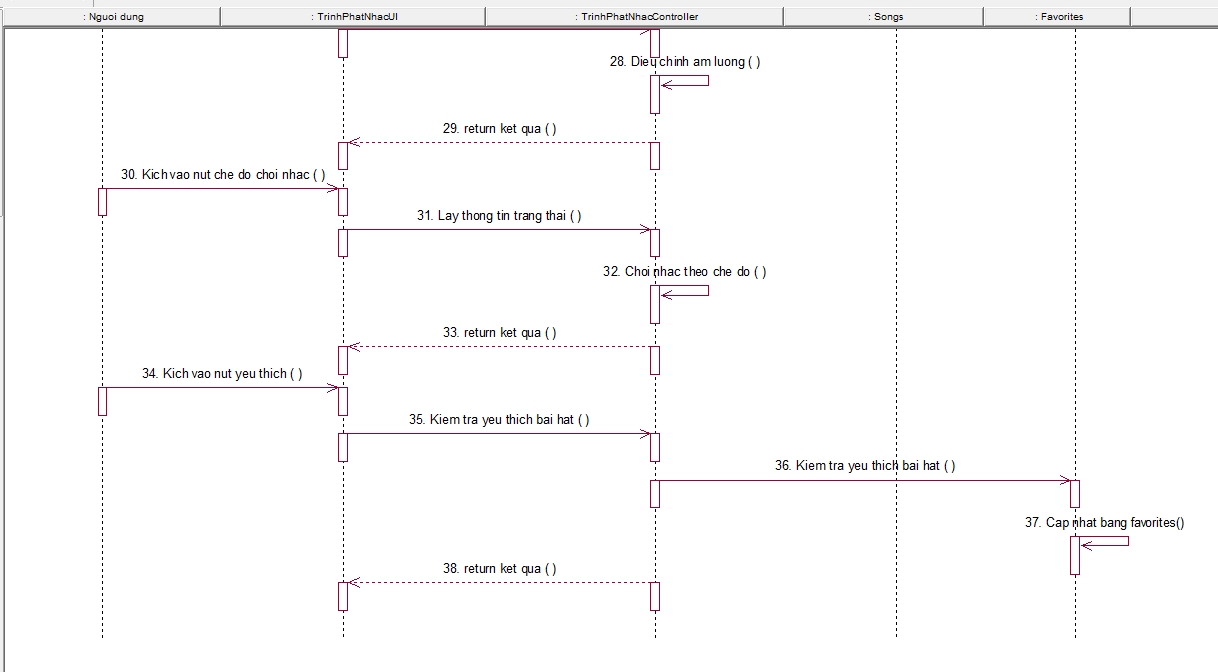
Hậu điều kiện: Không có.

Điểm mở rộng: Không có.

##### Biểu đồ trình tự

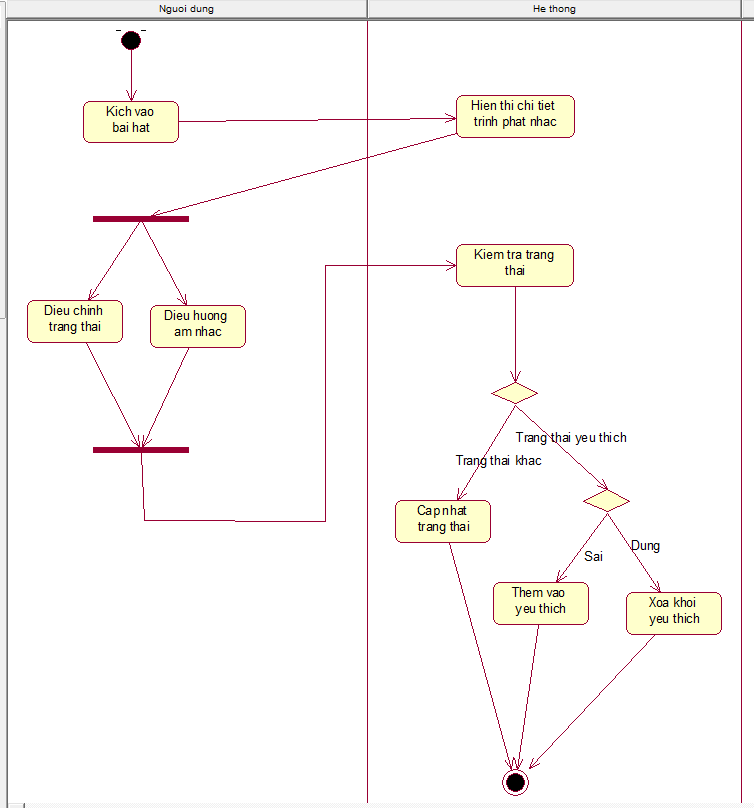






Hình 2.18 Biểu đồ trình tự Usecase Chi tiết trình phát nhạc

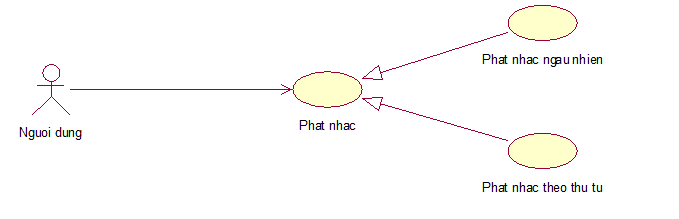
##### Biểu đồ hoạt động



Hình 2.19 Biểu đồ hoạt động Usecase Chi tiết trình phát nhạc

#### 2.2.2.7 Usecase Phát nhạc

##### Biểu đồ Usecase Phát nhạc



Hình 2.20 Biểu đồ Usecase Phát nhạc

##### Đặc tả chi tiết

Mô tả vắn tắt: Use case cho phép người phát nhạc theo tùy chọn.

Luồng sự kiện:

+ Luồng cơ bản:

1. Use case này bắt đầu khi người dùng một bài nhạc trong danh sách, nút “Play” hoặc “Shuffle”. Hệ thống lấy danh sách và bắt đầu phát bài hát.
2. Chế độ chơi nhạc:
3. Người dùng kích vào nút “Play” trên màn hình. Hệ thống lấy danh sách bài hát và bắt đầu phát bài hát.
4. Người dùng kích vào nút “Shuffle” trên màn hình. Hệ thống lấy danh sách trộn ngẫu nhiên và bắt đầu phát bài hát.
5. Use case kết thúc.

+ Luồng rẽ nhánh:

1. Tại bất kỳ thời điểm nào trong quá trình thực hiện use case nếu không kết nối được với cơ sở dữ liệu thì hệ thống sẽ hiển thị thông báo lỗi và use case kết thúc.

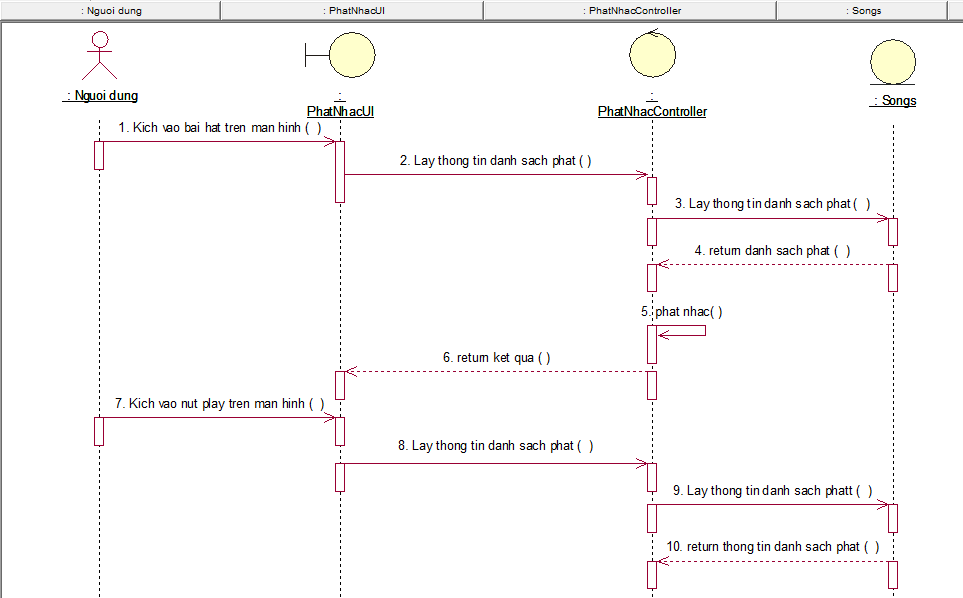
Các yêu cầu đặc biệt: Không có.

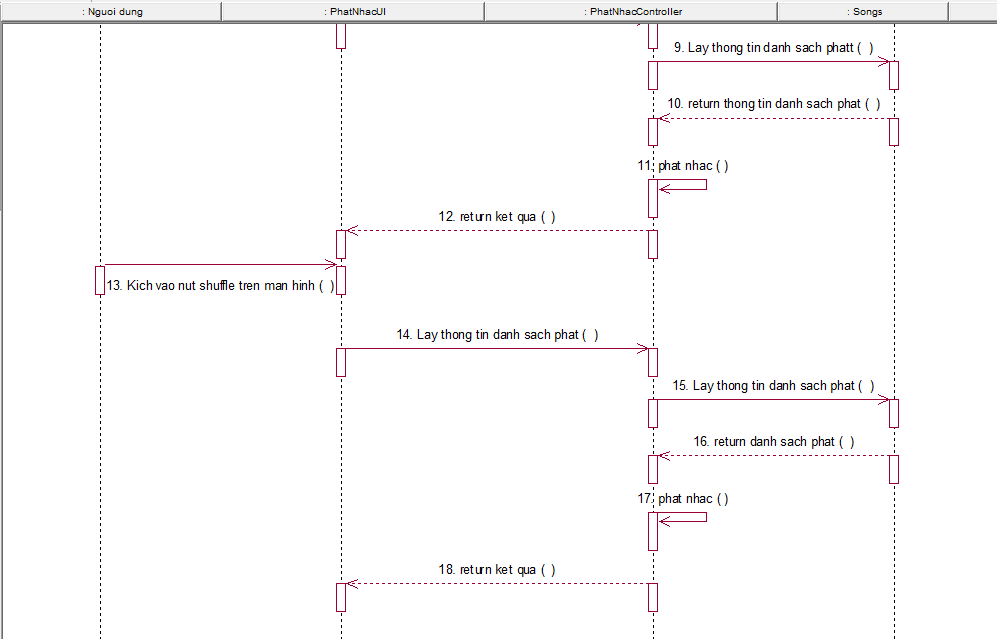
Tiền điều kiện: Không có.

Hậu điều kiện: Không có.

Điểm mở rộng: Không có.

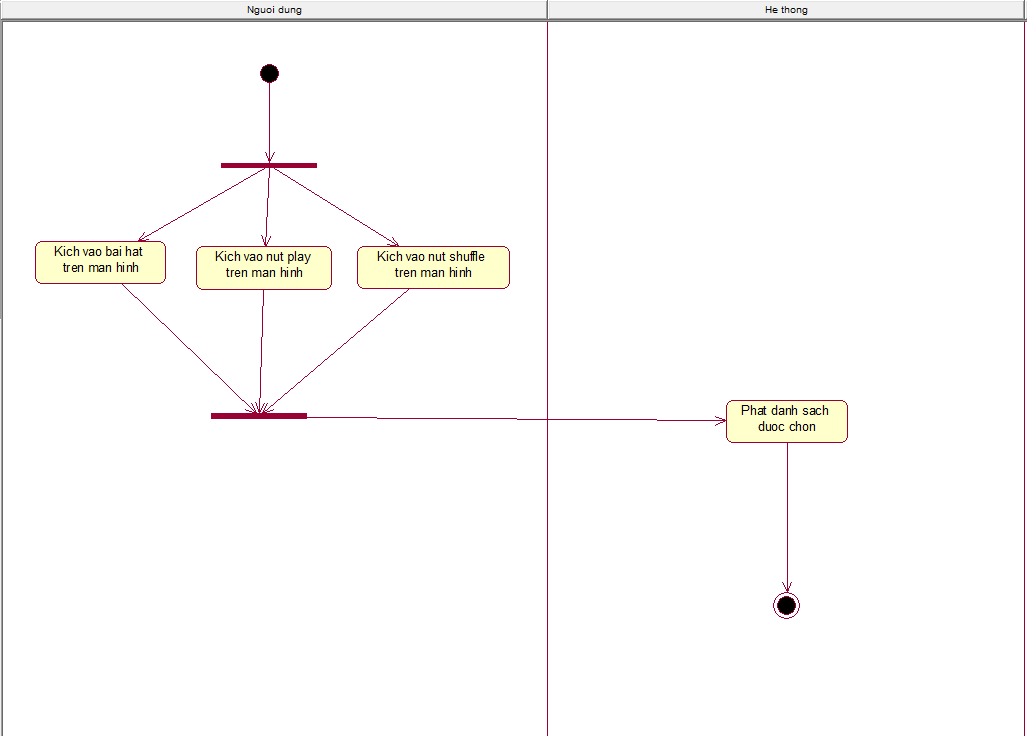
##### Biểu đồ trình tự





Hình 2.21 Biểu đồ Usecase Phát nhạc

##### Biểu đồ hoạt động



Hình 2.22 Biểu đồ hoạt động Usecase phát nhạc

#### 2.2.2.8 Usecase Xem danh sách bài hát

##### Biểu đồ Usecase Xem danh sách bài hát



Hình 2. 23 Biểu đồ Usecase Xem danh sách bài hát

##### Đặc tả chi tiết Usecase xem danh sách bài hát

Mô tả vắn tắt: Use case cho phép người dùng xem danh sách bài hát.

Luồng sự kiện:

+ Luồng cơ bản:

1. Use case này bắt đầu khi khách hàng kích vào biểu tượng “Song” hiển thị trên màn hình.
2. Hệ thống lấy thông tin chi tiết danh sách bài nhạc và hiển thị lên màn hình.
3. Use case kết thúc.

+ Luồng rẽ nhánh:

1. Tại bất kỳ thời điểm nào trong quá trình thực hiện use case nếu không kết nối được với cơ sở dữ liệu thì hệ thống sẽ hiển thị thông báo lỗi và use case kết thúc.

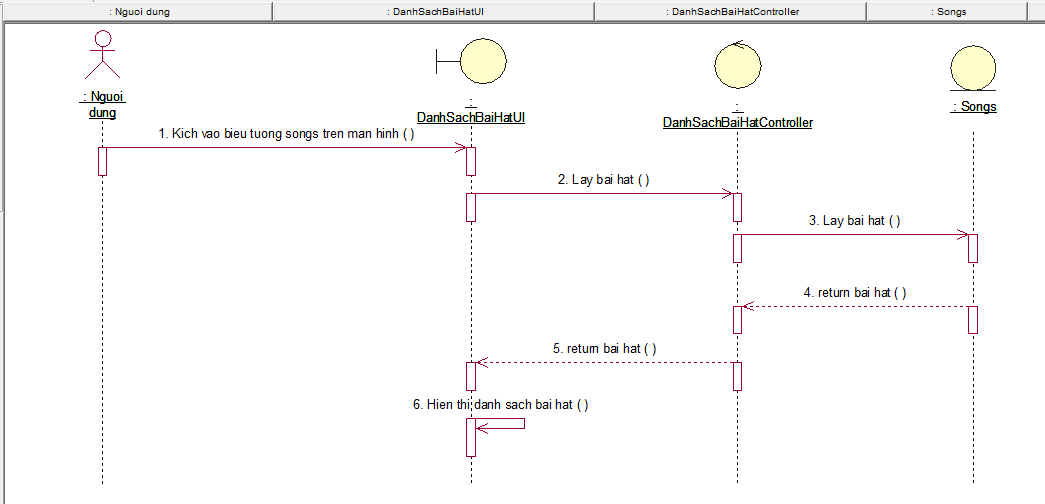
Các yêu cầu đặc biệt: Không có.

Tiền điều kiện: Không có.

Hậu điều kiện: Không có.

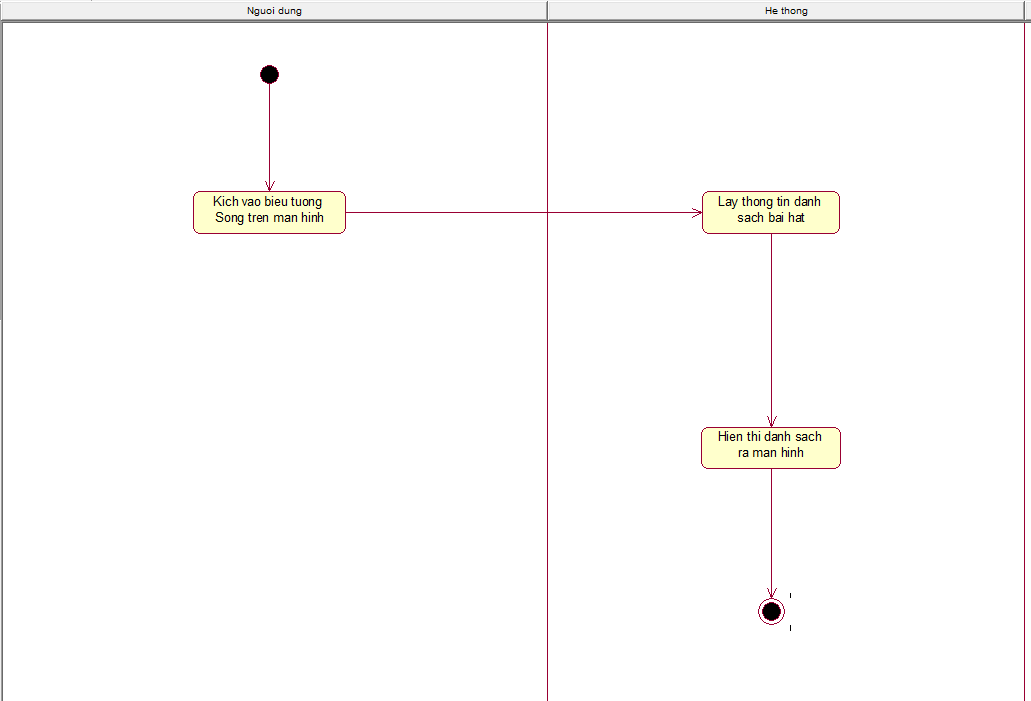
Điểm mở rộng: Không có

##### Biểu đồ trình tự



Hình 2.24 Biểu đồ trình tự Usecase Xem danh sách bài hát

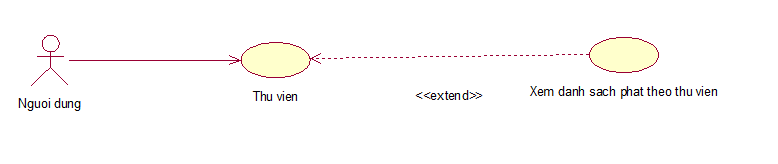
##### Biểu đồ hoạt động Usecase



Hình 2.25 Biểu đồ hoạt động Usecase Xem danh sách bài hát

#### 2.2.2.9 Usecase Thư viện

##### Biểu đồ Usecase Thư viện



Hình 2.26 Biểu đồ Usecase Thư viện

##### Đặc tả chi tiết Usecase thư viện

Mô tả vắn tắt: Use case cho phép người dùng xem danh sách thư viện.

Luồng sự kiện:

+ Luồng cơ bản:

1. Use case này bắt đầu khi khách hàng kích vào biểu tượng “Playlist” hiển thị trên màn hình.
2. Hệ thống lấy thông tin chi tiết danh sách nhạc và hiển thị lên màn hình.
3. Use case kết thúc.

+ Luồng rẽ nhánh:

1. Tại bất kỳ thời điểm nào trong quá trình thực hiện use case nếu không kết nối được với cơ sở dữ liệu thì hệ thống sẽ hiển thị thông báo lỗi và use case kết thúc.

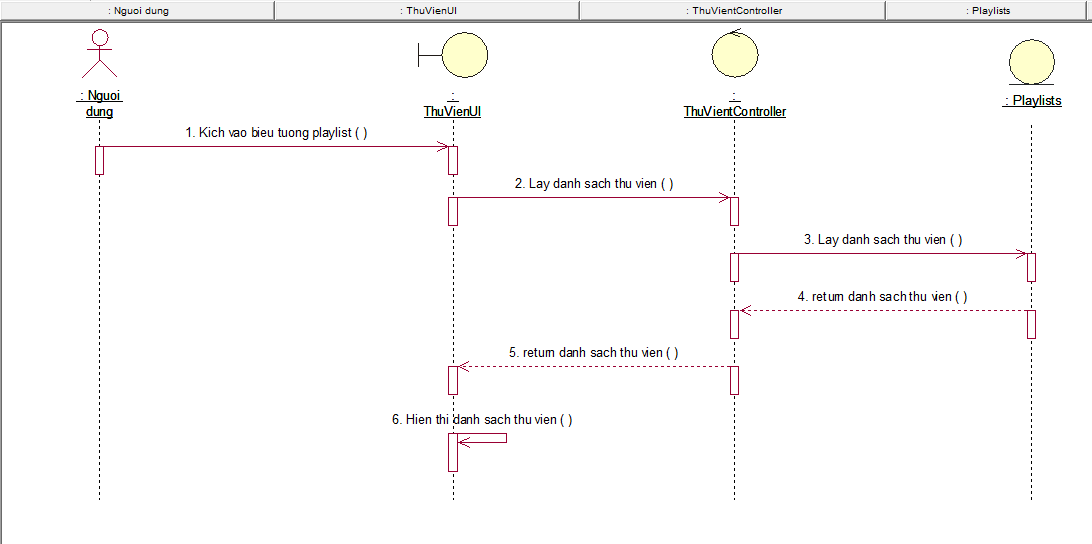
Các yêu cầu đặc biệt: Không có.

Tiền điều kiện: Không có.

Hậu điều kiện: Không có.

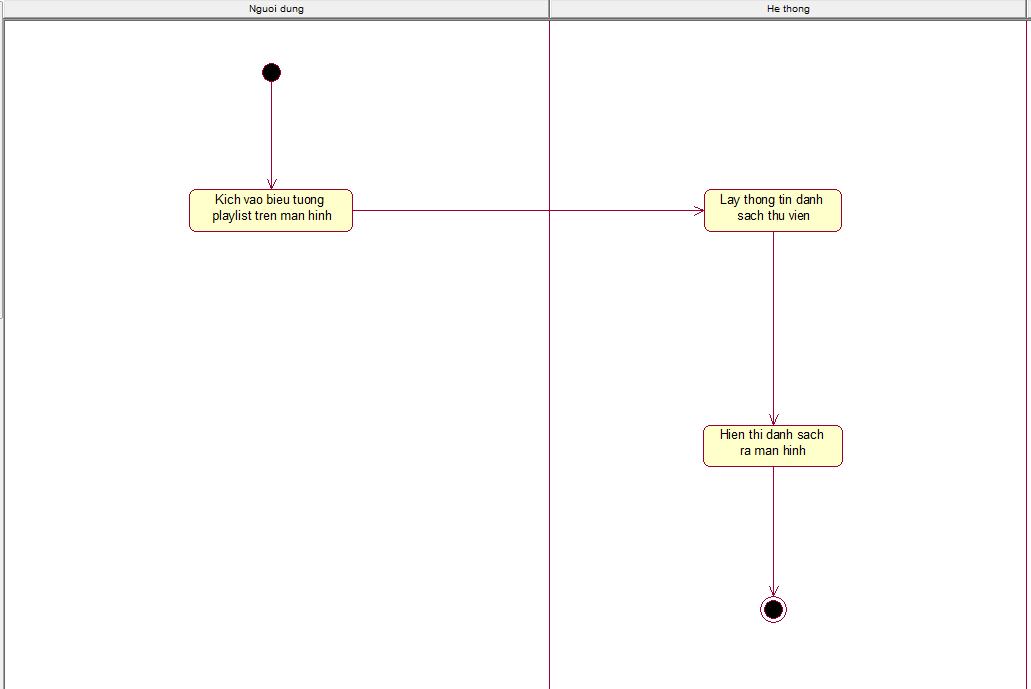
Điểm mở rộng: Không có.

##### Biểu đồ trình tự



Hình 2.27 Biểu đồ trình tự Usecase Thư viện

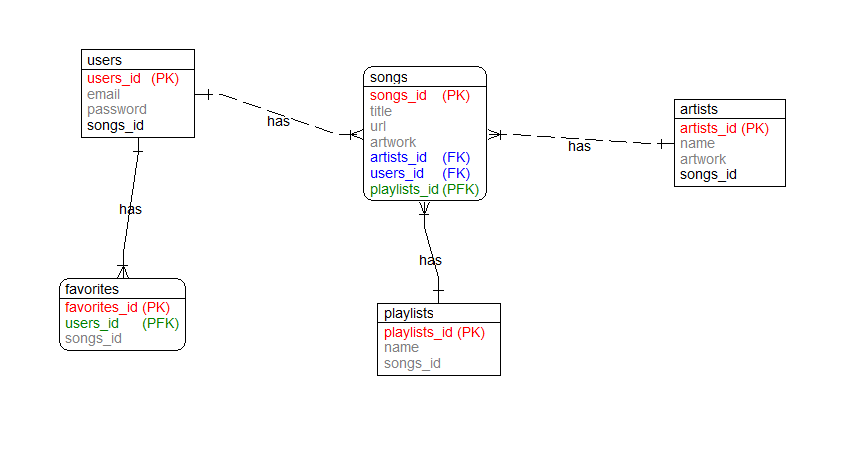
##### Biểu đồ hoạt động



Hình 2.28 Biểu đồ hoạt động Usecase Thư viện

# CHƯƠNG 3: XÂY DỰNG ỨNG DỤNG

## Biểu đồ Entity Relationship Diagram



Hình 3.1 Biểu đồ Entity Relationship Diagram

* 1. **Các bảng trong cơ sở dữ liệu**

**Bảng Users**

|  |  |  |
| --- | --- | --- |
| Thuộc tính | Kiểu dữ liệu | Ghi chú |
| users\_id (PK) | number | Định danh duy nhất cho mỗi người dùng |
| Email | string | Email của người dùng |
| password | string | Mật khẩu của người dùng |
| Songs\_id | array(number) | Danh sách songs\_id của người dùng. |

Bảng 3.1 Bảng Users

**Bảng Songs**

|  |  |  |
| --- | --- | --- |
| Thuộc tính | Kiểu dữ liệu | Ghi chú |
| songs\_id (PK) | number | Định danh duy nhất cho mỗi bài hát. |
| Title | string | Danh sách chứa user\_id của người gửi và người nhận. |
| url | string | Đường dẫn đến bài hát |
| Artwork | string | Đường dẫn đến hình ảnh của bài hát |
| Artists\_id | number | Artist\_id của ca sĩ |
| Playlists\_id | array(number) | Danh sách playlists\_id của bài hát |

Bảng 3.2 Bảng Songs

**Bảng Playlists**

|  |  |  |
| --- | --- | --- |
| Thuộc tính | Kiểu dữ liệu | Ghi chú |
| playlists\_id (PK) | number | Định danh duy nhất cho mỗi playlist |
| name | string | Tên của playlist |
| songs\_id | array(number) | Danh sách songs\_id có trong playlist. |

Bảng 3.3 Bảng Playlists

**Bảng Artists**

|  |  |  |
| --- | --- | --- |
| Thuộc tính | Kiểu dữ liệu | Ghi chú |
| artists\_id (PK) | number | Định danh duy nhất cho mỗi ca sĩ |
| name | string | Tên của ca sĩ |
| artwork | string | Đường dẫn hình ảnh của ca sĩ. |
| songs\_id | array(number) | Danh sách songs\_id của ca sĩ |

Bảng 3.4 Bảng Artists

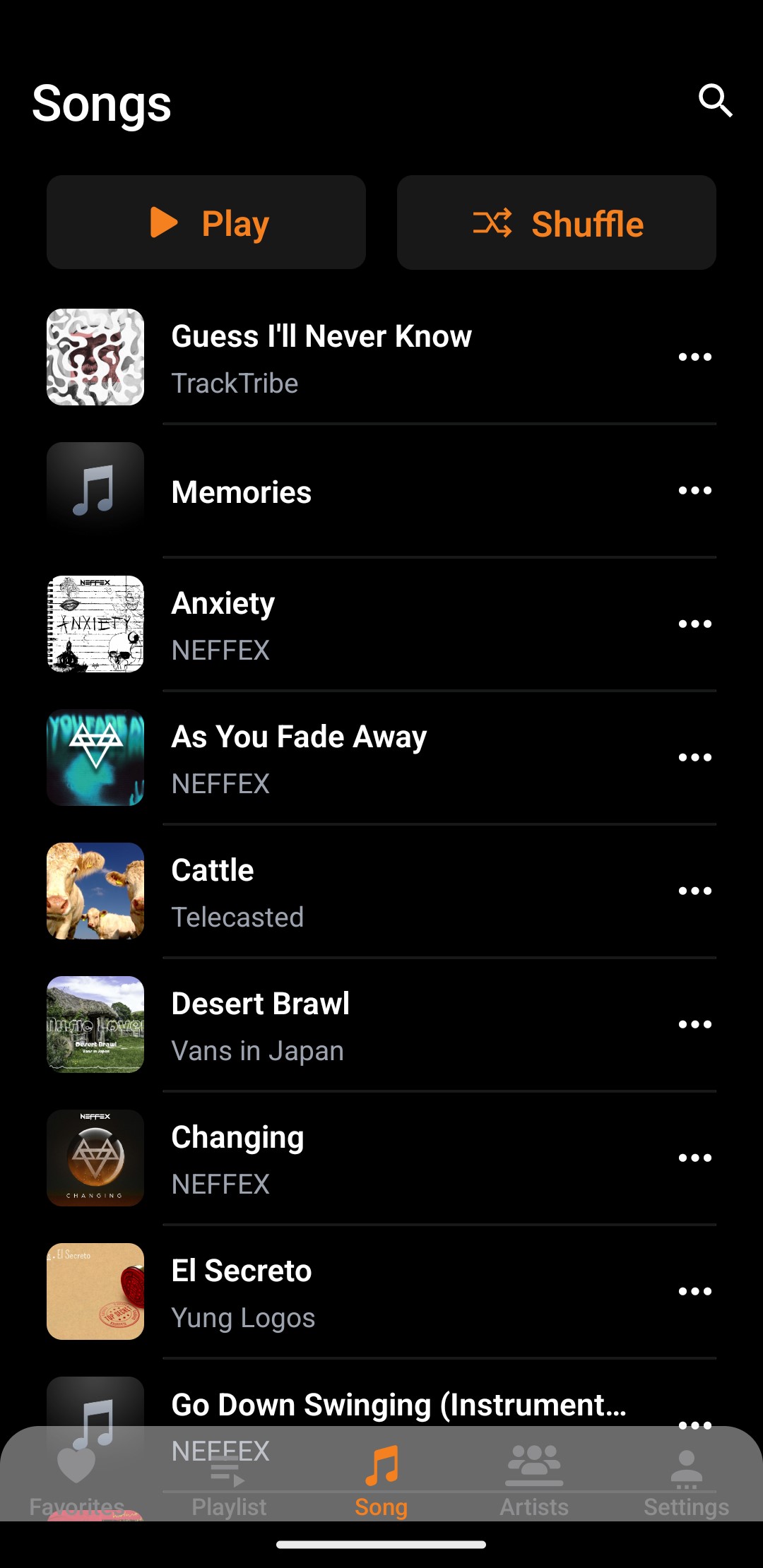
**Bảng Favorites**

|  |  |  |
| --- | --- | --- |
| Thuộc tính | Kiểu dữ liệu | Ghi chú |
| favorites\_id | number | Định danh duy nhất cho mỗi bảng yêu thích |
| users\_id | number | Định danh cho người dùng |
| songs\_id | array(number) | Danh sach songs\_id đã yêu thích của người dùng |

Bảng 3.5 Bảng Favorites

* 1. **Kết quả giao diện màn hình**
     1. **Giao diện màn hình chính**

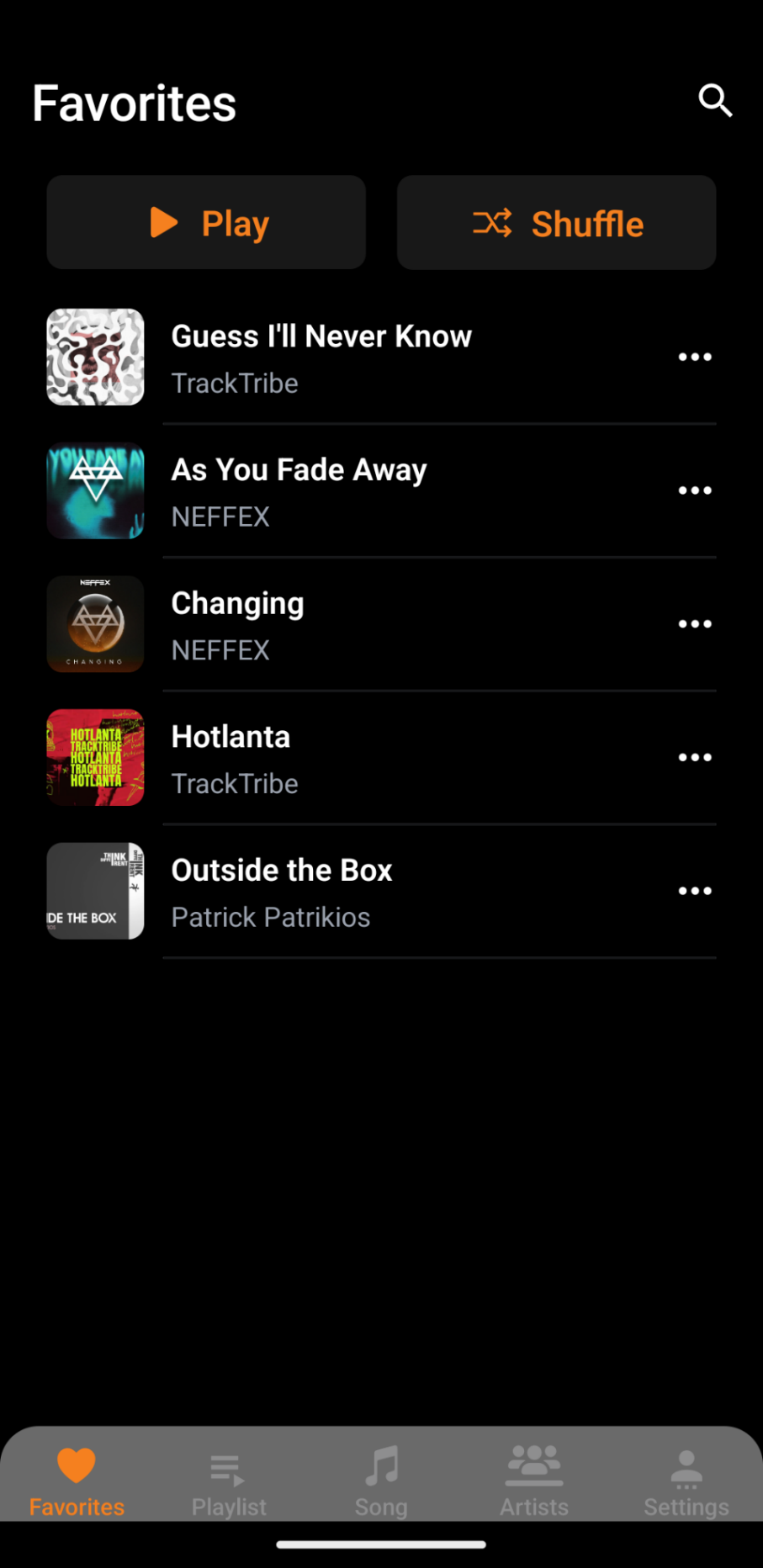
Sau khi thực hiện đăng nhập màn hình đăng nhập ứng dụng sẽ được chuyển sang màn hình chính, ở đây người dùng có thể thấy những thông tin cơ bản của một ứng dụng nghe nhạc như: danh sách bài hát, các tab chuyển hướng, tìm kiếm... Màn hình này chính là cơ sở để chuyển sang các màn hình khác trong ứng dụng.



Hình 3.2 Giao diện màn hình chính

* + 1. **Giao diện màn hình yêu thích**

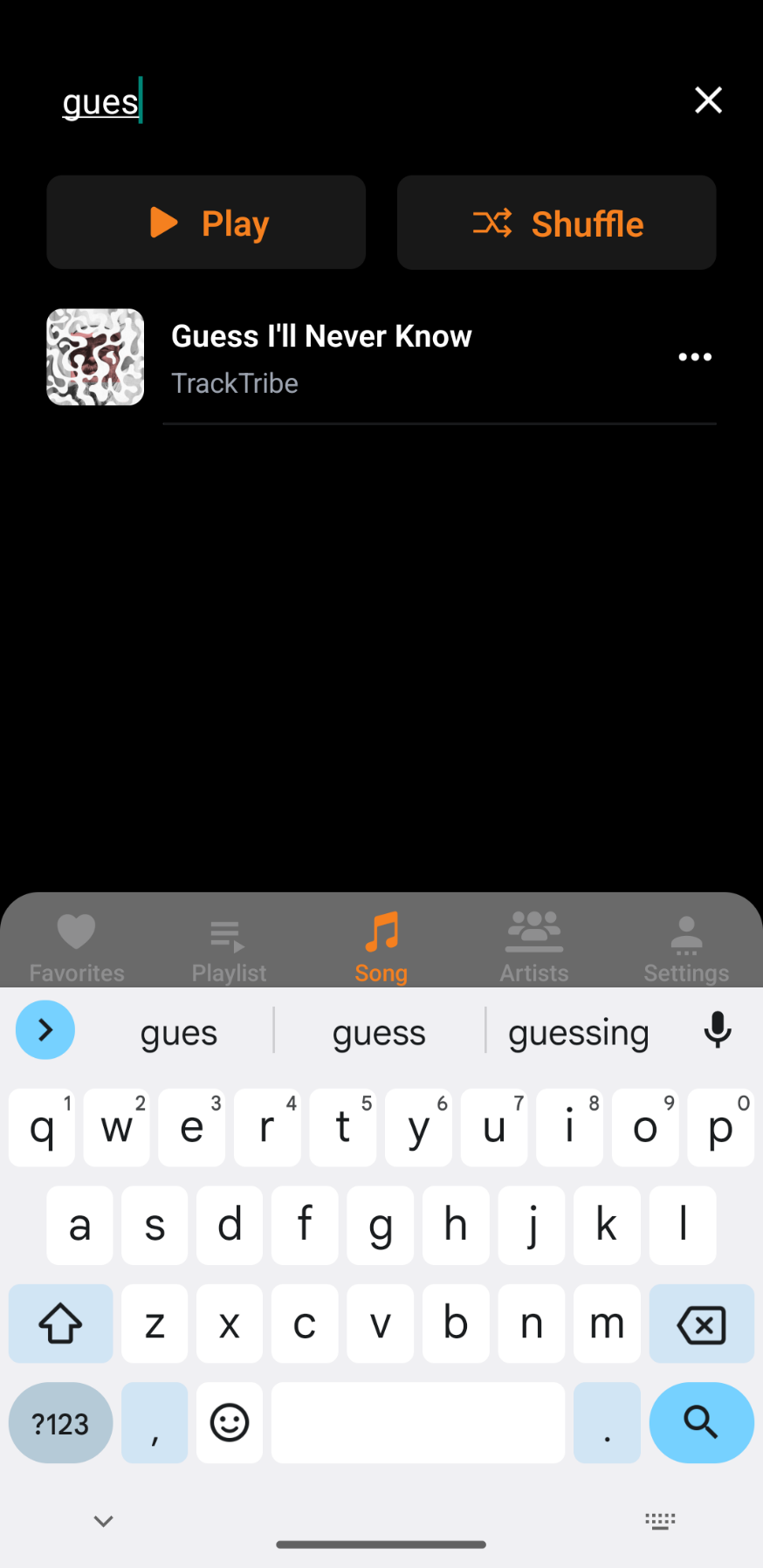
Màn hình thể hiện danh sách bài hát đã được yêu thích bởi người dùng.



Hình 3.3 Giao diện màn hình yêu thích

* + 1. **Giao diện màn hình tìm kiếm**

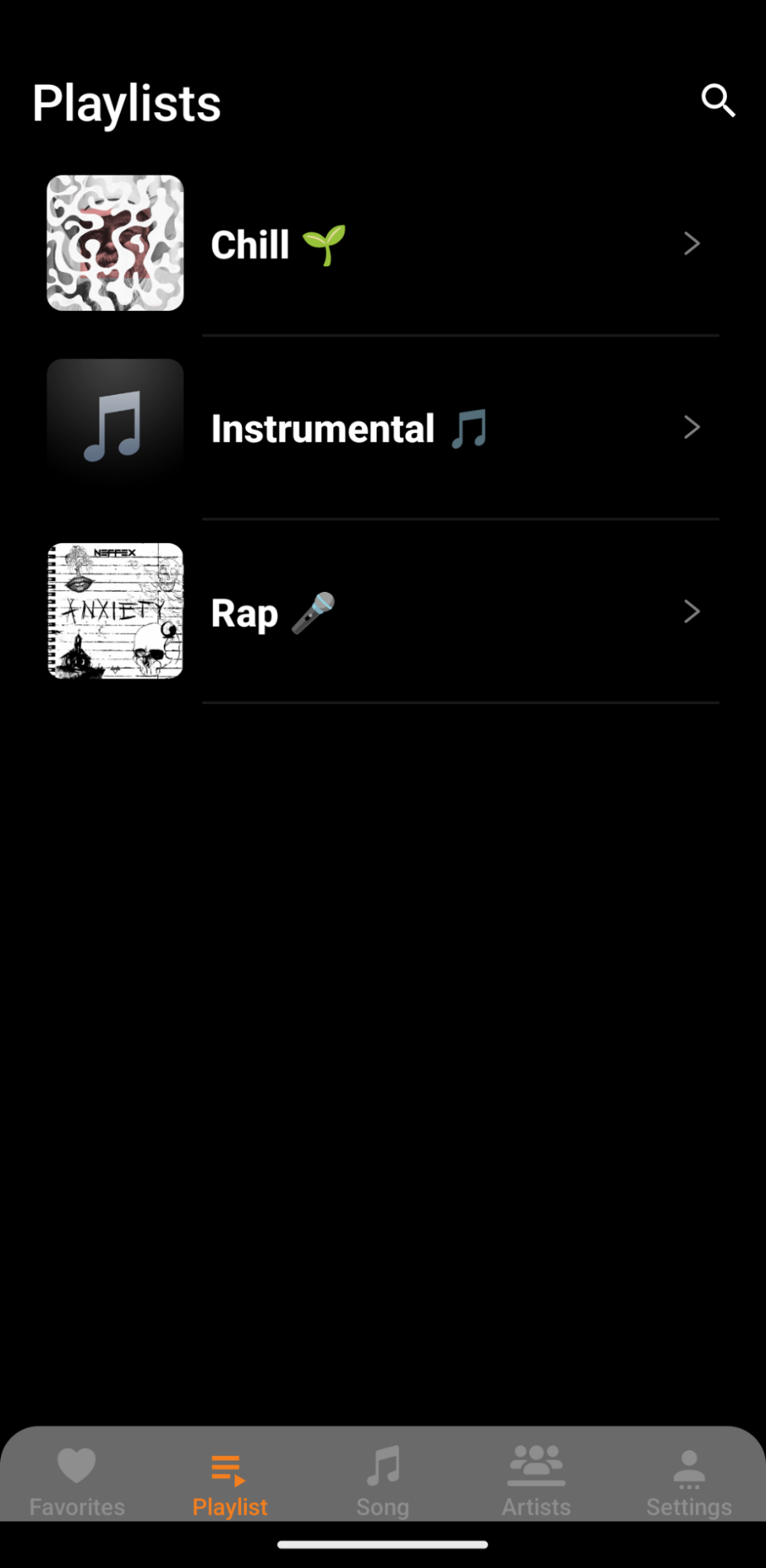
Màn hình giúp người dùng tìm kiếm những bài hát yêu thích của mình theo tên hoặc theo tên ca sĩ và nghe chúng.



Hình 3.4 Giao diện màn hình tìm kiếm

* + 1. **Giao diện màn hình thư viện**

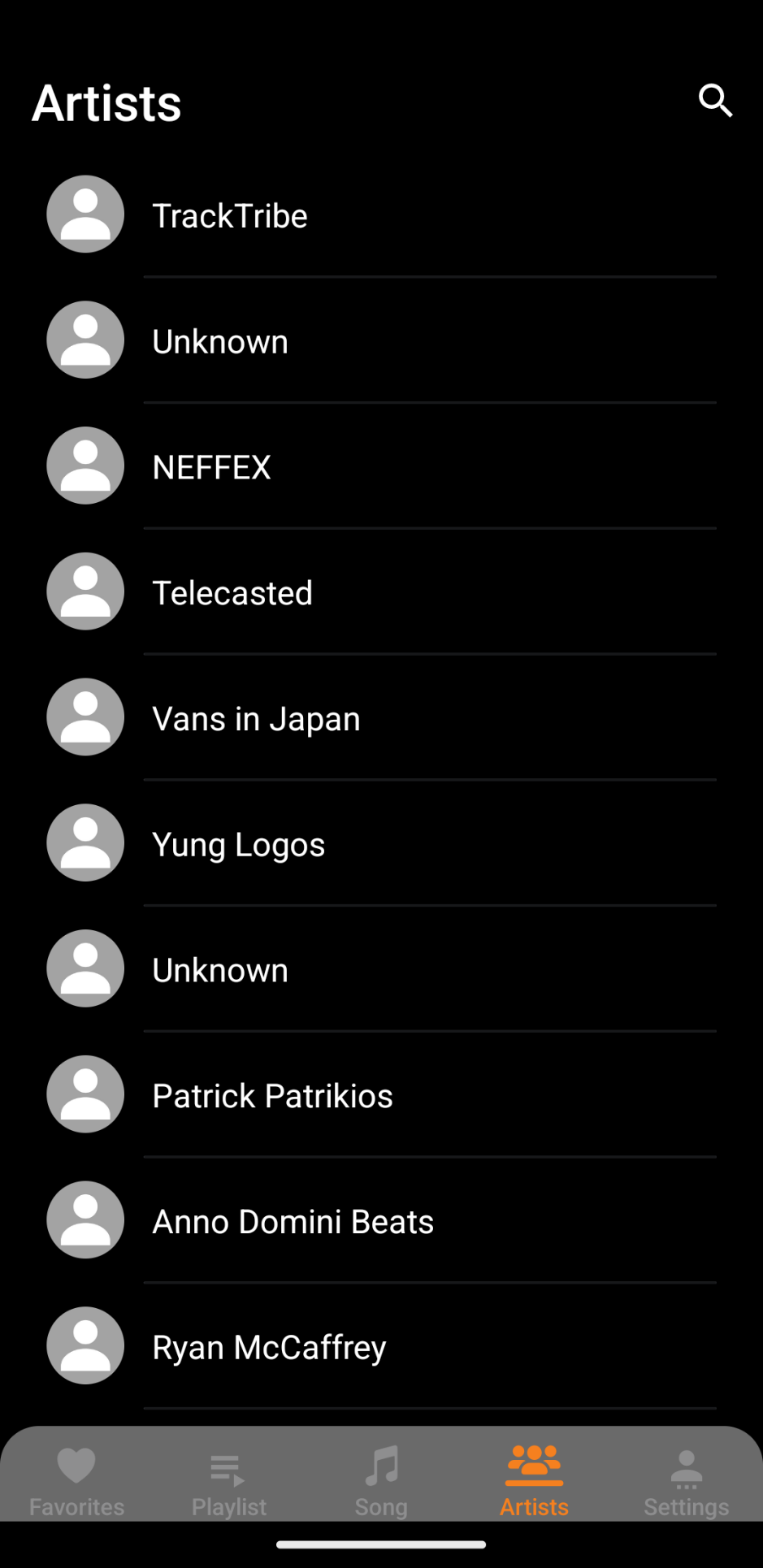
Màn hình thể hiện các danh sách thư viện.



Hình 3.5 Giao diện màn hình thư viện

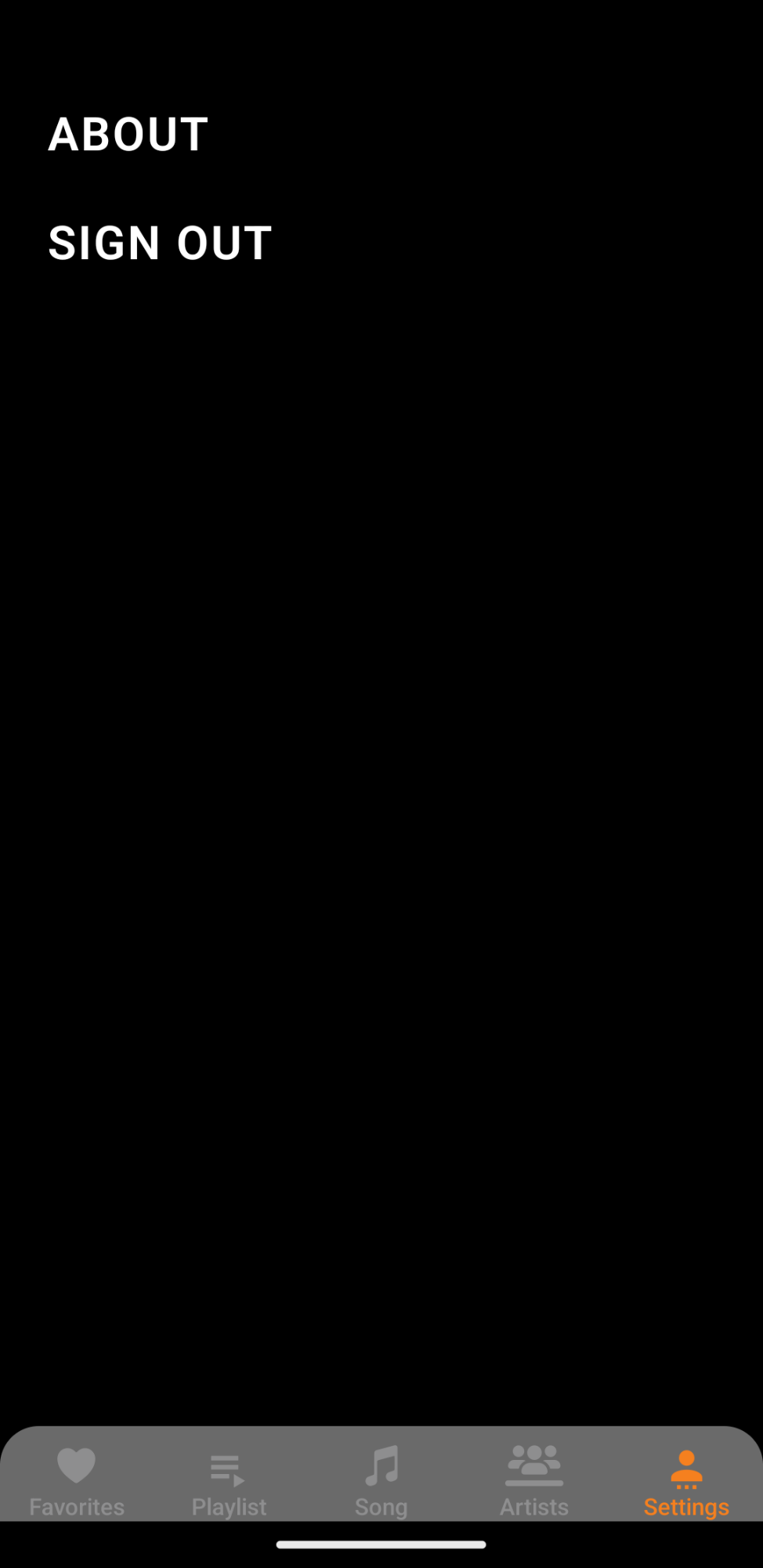
* + 1. **Giao diện màn hình ca sĩ**

Màn hình này thể hiện danh sách ca sĩ.



Hình 3.6 Giao diện màn hình ca sĩ

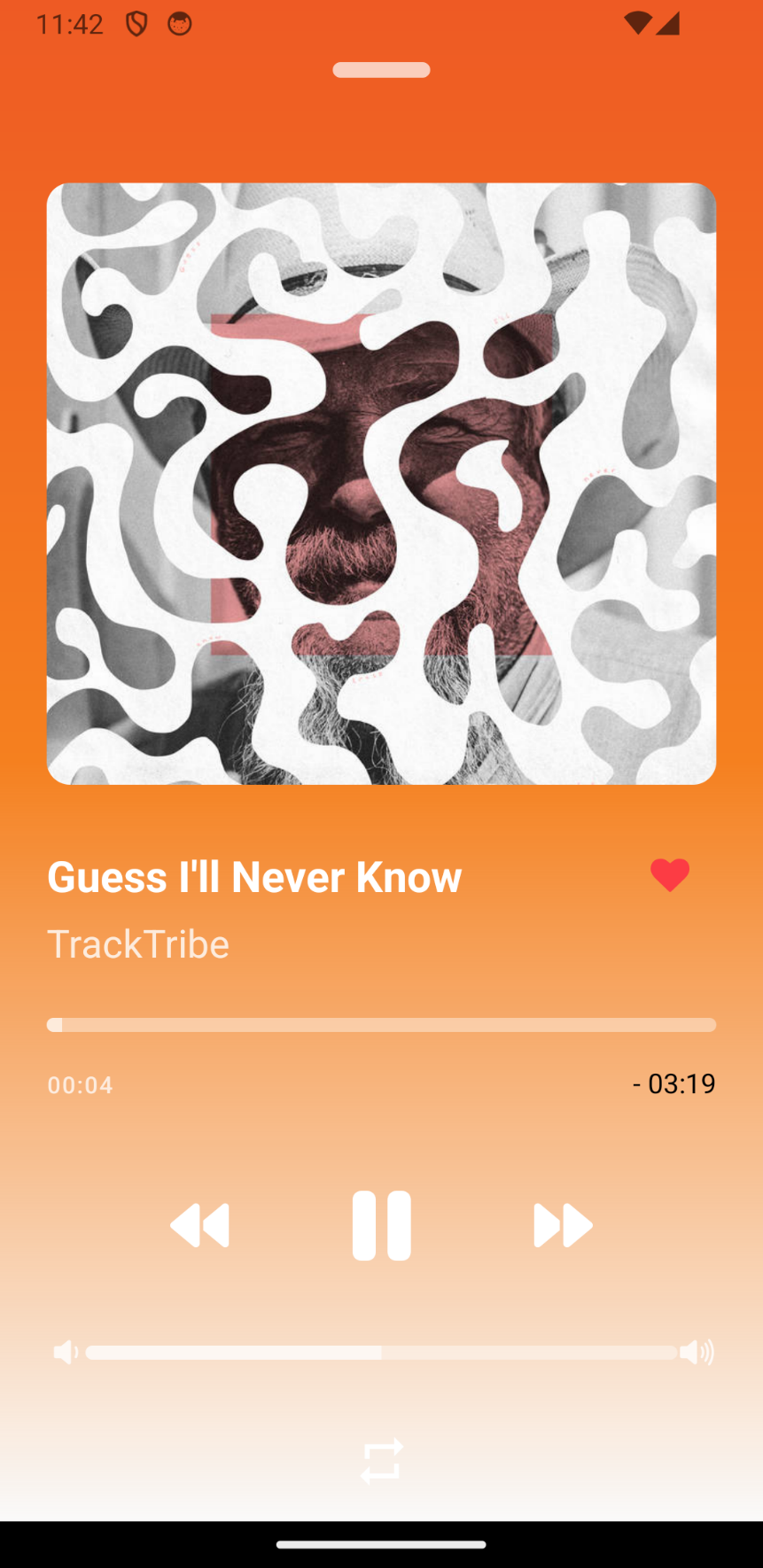
* + 1. **Giao diện màn hình đăng xuất**



Hình 3.7 Giao diện màn hình đăng xuất

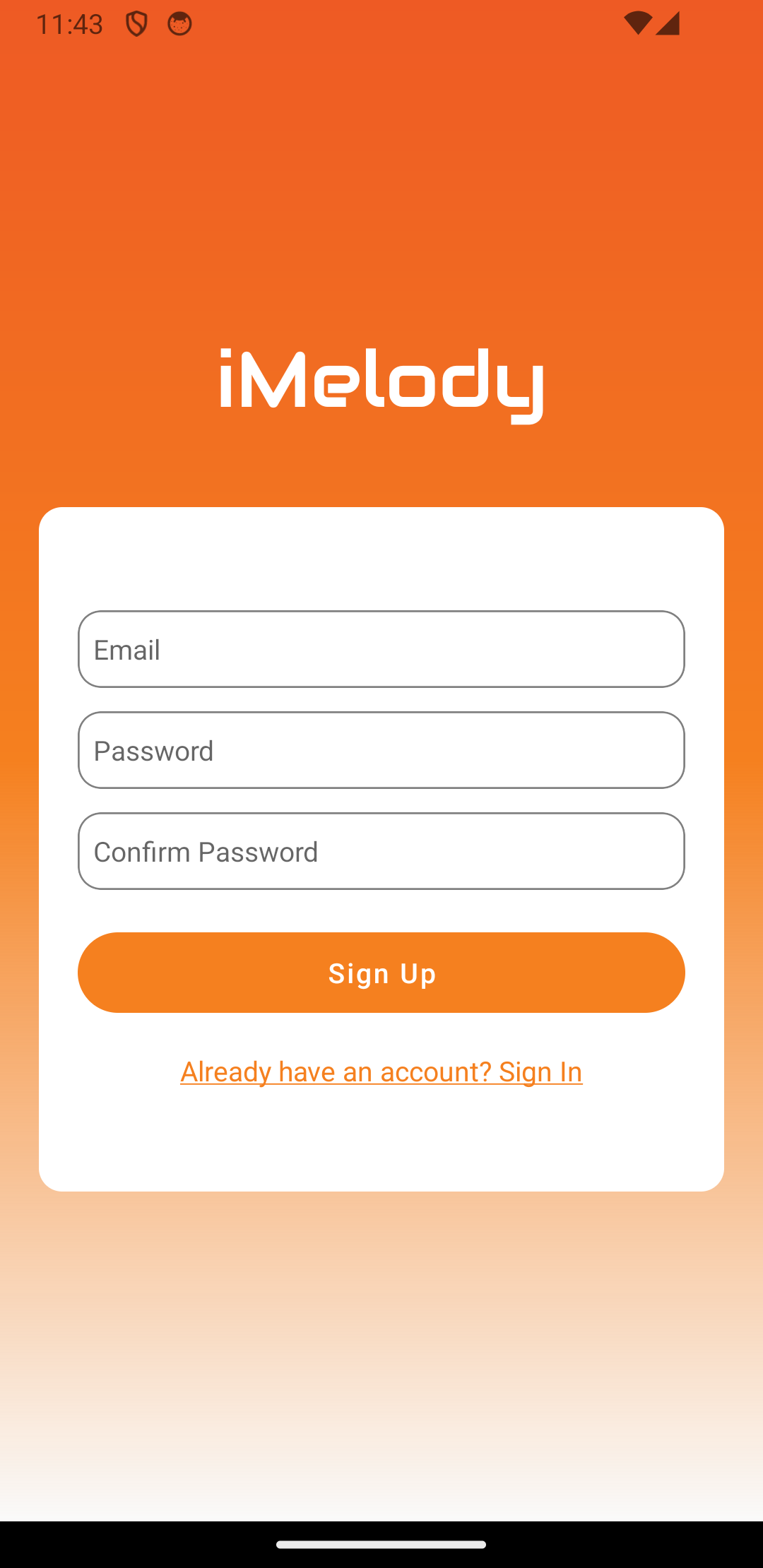
* + 1. **Giao diện trình phát nhạc**

Khi người dùng đã chọn được bài hát muốn nghe thì người dùng sẽ được chuyển sang màn hình trình phát nhạc, trong quá trình nghe nhạc có thể tăng giảm âm lượng, điều hướng bài hát, chuyển chế độ nghe.



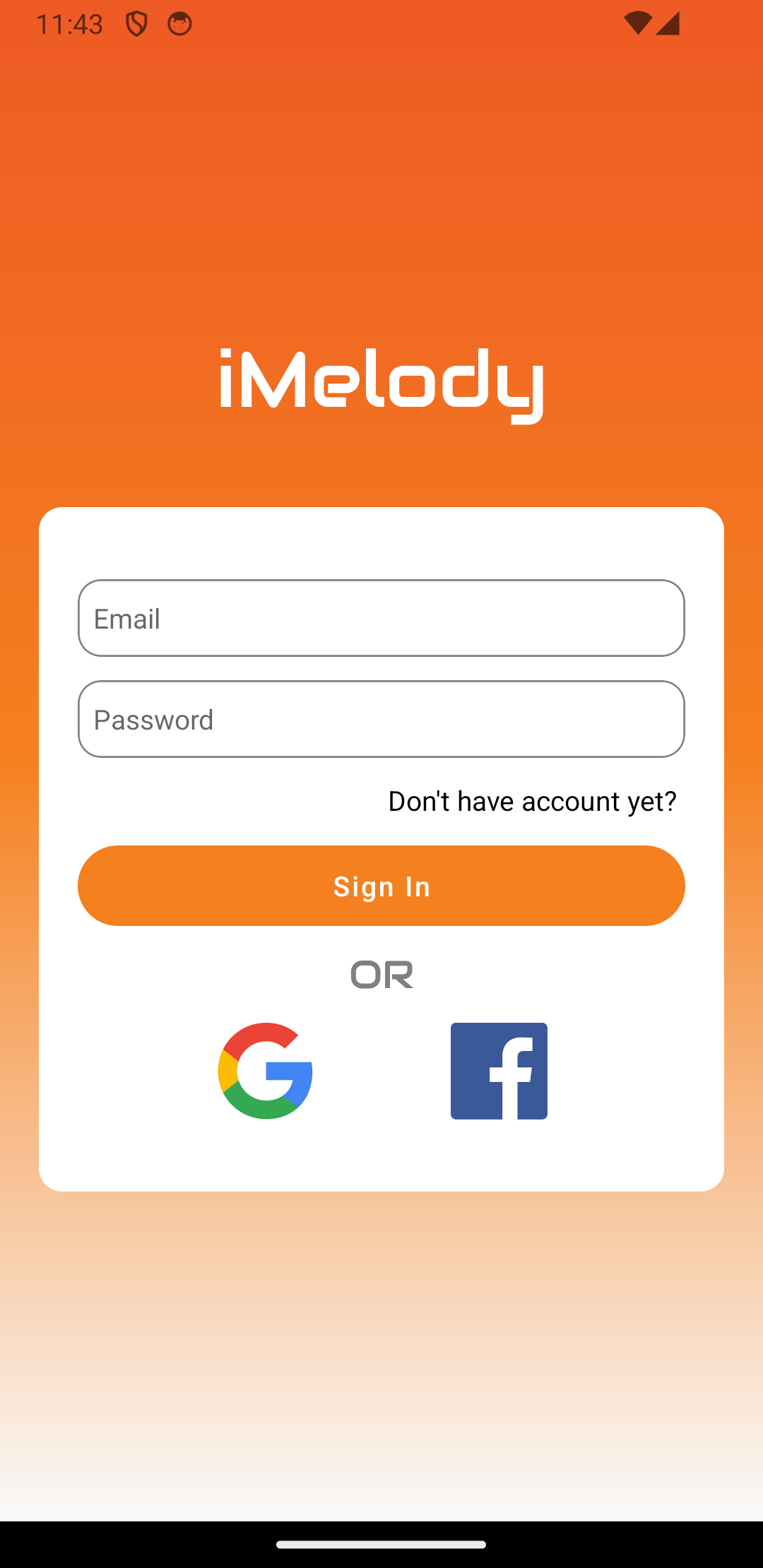
Hình 3.8 Giao diện trình phát nhạc

* + 1. **Giao diện đăng ký**



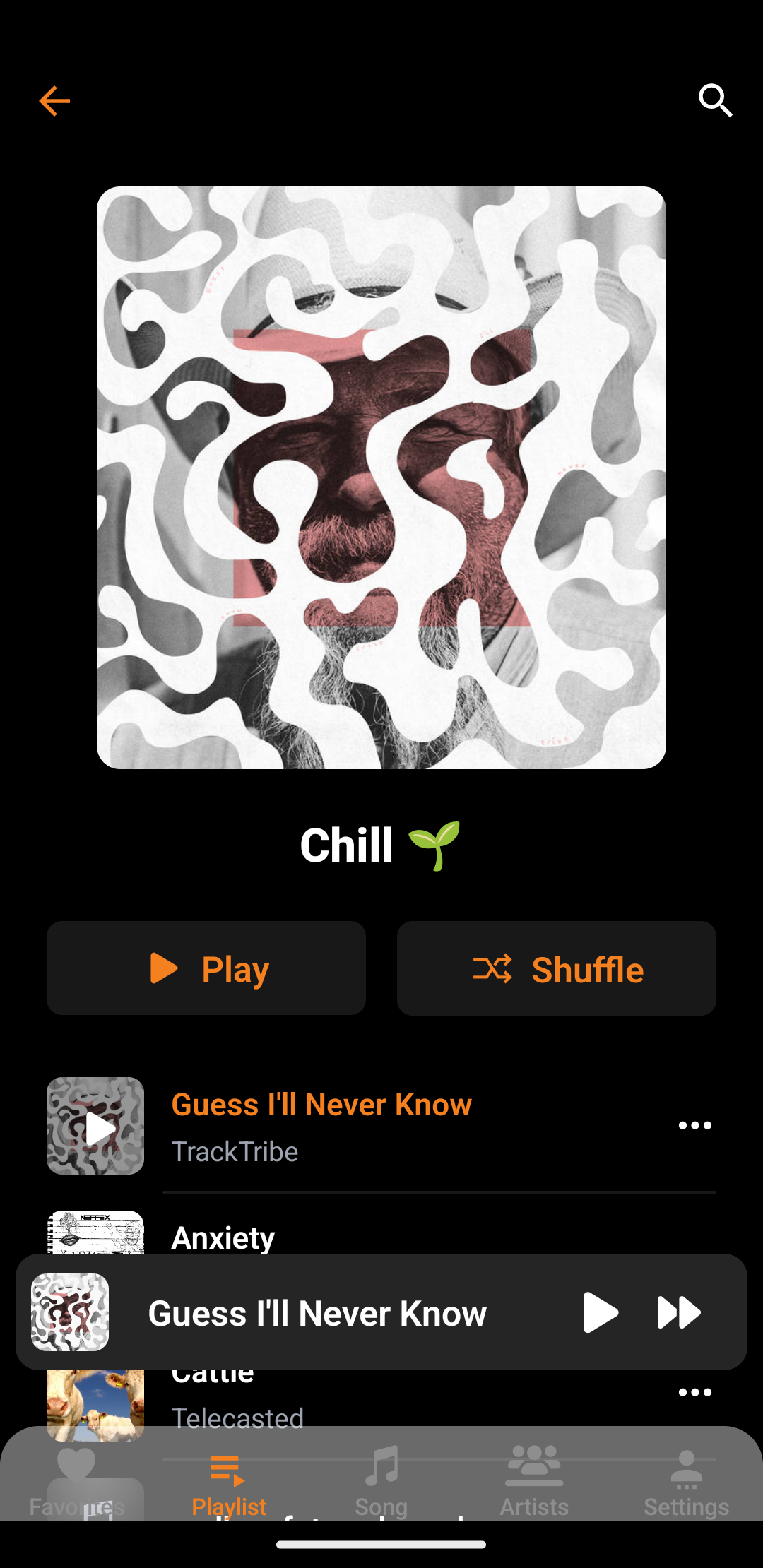
Hình 3.9 Giao diện màn hình đăng ký

* + 1. **Giao diện đăng nhập**



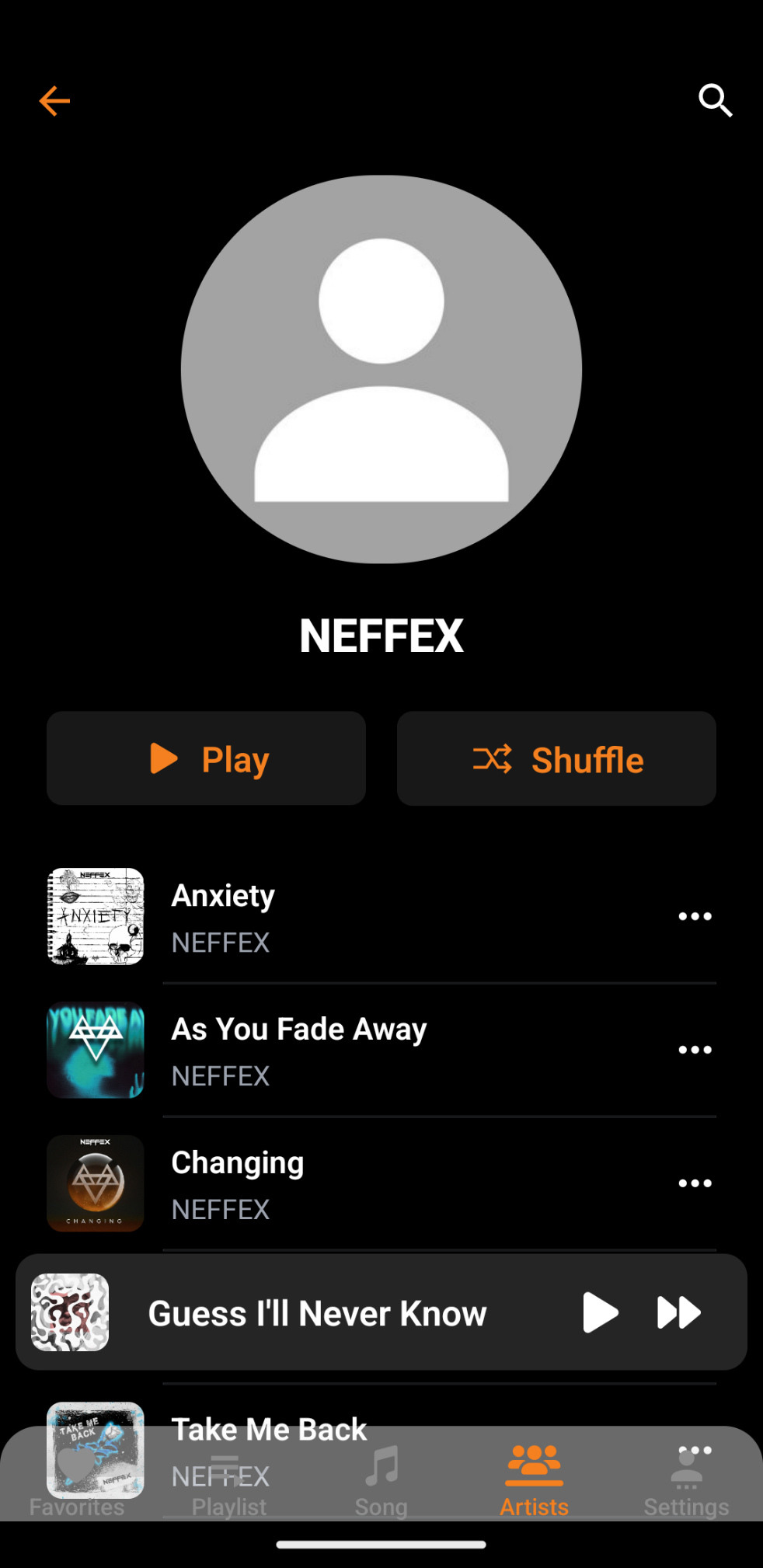
Hình 3.10 Giao diện màn hình đăng nhập

### Giao diện danh sách nhạc trong thư viện



Hình 3.11Giao diện danh sách nhạc theo thư viện

* + 1. **Giao diện danh sách nhạc theo ca sĩ**



Hình 3.12 Giao diện danh sách nhạc theo ca sĩ

* 1. **Kế hoạch kiểm thử**

Những chức năng được kiểm thử:

* Đăng nhập: Người dùng đăng nhập vào hệ thống.
* Đăng ký: Người dùng đăng ký tài khoản mới.
* Đăng xuất: Người dùng đăng xuất tài khoản ra khỏi hệ thống.
* Tìm kiếm
  + Tìm kiếm theo tên: Người dùng tìm kiếm theo tên bài hát
  + Tìm kiếm theo tên ca sĩ: Người dùng tìm kiếm theo tên ca sĩ
* Yêu thích: Người dùng thêm bài hát vào yêu thích.
* Trình phát nhạc: Người dùng sử dụng các chức năng của trình phát nhạc.
  1. **Xây dựng kịch bản**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **STT** | **Case** | **Đầu vào** | **Đầu ra mong muốn** | **Kết quả** |
| 1 | Kiểm tra chức năng đăng nhập | Nhập tài khoản & mật khẩu hợp lệ | Đăng nhập thành công và đưa người dùng sang trang chủ. | Pass |
| Nhập tài khoản & mật khẩu không hợp lệ | Đăng nhập không thành công và thông báo ra màn hình. | Pass |
| 2 | Kiểm tra chức năng đăng ký tài khoản | Đăng ký với thông tin hợp lệ | Đăng ký thành công và chuyển sang màn hình ứng dụng. | Pass |
| Đăng ký với thông tin không hợp lệ | Hệ thống thông báo không hợp lệ. | Pass |
| 3 | Kiểm tra chức năng đăng xuất | Đăng xuất tài khoản | Đăng xuất khỏi tài khoản, chuyển sang màn hình đăng nhập. | Pass |
| 4 | Kiểm tra chức năng tìm kiếm theo tên | Nhập từ khóa bài hát cần tìm có tồn tại | Hiển thị tất cả các bài hát liên quan đến từ khóa vừa nhập. | Pass |
| Nhập từ khóa bài hát cần tìm không tồn tại | Hiển thị thông báo trên màn hình | Pass |
| 5 | Kiểm tra chức năng tìm kiếm theo tên ca sĩ | Nhập từ khóa ca sĩ cần tìm có tồn tại | Hiển thị tất cả các bài hát liên quan đến từ khóa vừa nhập. | Pass |
| Nhập từ khóa ca sĩ cần tìm không tồn tại | Hiển thị thông báo trên màn hình | Pass |
| 6 | Kiểm tra chức năng yêu thích | Bài hát đã được yêu thích rồi | Xóa bài hát khỏi danh sách yêu thích | Pass |
| Bài hát chưa được yêu thích | Thêm bài hát vào danh sách yêu thích | Pass |
| 7 | Kiểm tra chức năng trình phát nhạc | Trình phát nhạc hoạt động bình thường | Trình phát nhạc có thể nghe được bình thường | Pass |
| Người dùng bấm vào nút Play | Bài hát được phát | Pass |
| 7 | Kiểm tra chức năng trình phát nhạc | Người dùng bấm vào nút Pause | Bài hát dừng lại | Pass |
| Người dùng tăng giảm âm lượng | Âm lượng của bài hát tăng giảm theo thanh âm lượng | Pass |
| Người dùng bấm các nút điều hướng bài hát | Trình phát nhạc mở bài hát kế tiếp hoặc trước đó theo nút điều hướng | Pass |
| Người dùng chọn chế độ chơi nhạc | Trình phát nhạc chơi nhạc theo chế độ người dùng chọn | Pass |

**3.6 Kết quả kiểm thử**

Tỉ lệ test case đạt (Passed): 100%

Tỉ lệ test case không đạt (Failed): 0%

Hệ thống chạy ổn định trên Emulator của Android Studio.

Giao diện dễ dàng sử dụng đối với người lần đầu tiếp cận.

Các chức năng thực hiện được thao tác dễ dàng.

**KẾT LUẬN**

Qua thời gian tìm hiểu và nghiên cứu đề tài “Xây dựng ứng dụng nghe nhacc iMelody” em đã học hỏi được cơ bản một số công nghệ: React Native, Firebase, Expo…. và áp dụng chúng vào chương trình của mình. Phân tích, thiết kế và hoàn thành cơ bản theo đúng những yêu cầu về nội dung và thời gian đã định.

Trong quá trình nghiên cứu và thực hiện đồ án do hạn chế về kinh nghiệm và kiến thức nên đề tài của em không thể tránh khỏi những thiếu sót. Em rất mong nhận được ý kiến đóng góp từ quý thầy cô và các bạn để ứng dụng có thể hoàn thiện hơn.

**Kết quả đạt được**

* Hoàn thiện khảo sát, phân tích, thiết kế ứng dụng nghe nhạc.
* Ứng dụng đã có một vài chức năng cơ bản như: đăng nhập, đăng ký, trình phát nhạc, quản lý danh sách, yêu thích, tìm kiếm.
* Ứng dụng đã có cơ sở dữ liệu đúng với yêu cầu phân tích, thiết kế.
* Ứng dụng đã được cập nhật giao diện đúng như mẫu thiết kế.

**Hạn chế của đề tài**

Do thời gian làm có hạn nên em chưa thể hoàn thiện đầy đủ chức năng của hệ thống, cũng như chưa xử lý được hết các lỗi phát sinh như:

* Ứng dụng vẫn còn hạn chế chức năng.
* Chưa kiểm soát được tất cả các lỗi.
* Chưa hoàn toàn hiểu rõ và áp dụng tốt được các công nghệ vào đề tài.

**Hướng phát triển tương lai**

* Thêm lời bài hát vào trình phát nhạc.
* Chế độ nghe nhạc ngoại tuyến.
* Tạo thêm các nội dung các ngoài bài hát ví dụ như Podcast, Radio….
* Tìm kiếm phương pháp kỹ thuật tự đề xuất bài hát theo sở thích người dùng.
* Phát hiện bài hát dựa trên giai điệu.

Cuối cùng xin gửi lời cảm ơn đến thầy Nguyễn Thanh Hải đã hướng dẫn em hoàn thành đề tài đồ án tốt nghiệp này!

# TÀI LIỆU THAM KHẢO

[1] Hoàng Quang Huy, Phùng Đức Hòa, Trịnh Bá Quý, Nhập môn công nghệ phần mềm, NXB Đại học Công nghiệp Hà Nội.

[2] Nguyễn Thị Thanh Huyền, Ngô Thị Bích Thúy, Phạm Thị Kim Phượng, Giáo trình phân tích thiết kế hệ thống, NXB Giáo dục VN.

[3] Trang tài liệu React Native: <https://reactnative.dev/>

[4] Trang tài liệu Expo: <https://expo.dev/>

[5] Trang tin tức về lập trình: <https://viblo.asia>

[6] Trang tài liệu Firebase: <https://firebase.google.com/>

[7] Tài liệu về quản lý thư viện trong Nodejs: <https://www.npmjs.com/>