

ĐỒ ÁN MÔN NHẬP MÔN ỨNG DỤNG DI ĐỘNG

Giảng viên hướng dẫn: Nguyễn Tấn Toàn

20521450

NHIỆM VỤ ĐỒ ÁN MÔN HỌC

Họ và tên SV 1: **Nguyễn Đức Long**..... MSSV: **20521566**.....

Họ và tên SV 1: **Nguyễn Tuấn Khôi**..... MSSV: **20521480**.....

Họ và tên SV 2: **Nguyễn Bá Khanh** MSSV: **20521450**.....

Lớp: **SE114.M21.PMCL**

Tên đề tài: **XÂY DỰNG ỨNG DỤNG NGHE NHẠC TRỰC TUYẾN**.....

Giảng viên giảng dạy **Nguyễn Tấn Toàn**

Thời gian thực hiện: **03/2022 đến 06/2022**

Nhiệm vụ đồ án môn học:

1. Xây dựng CSDL trong PhpMyAdmin.
2. Thiết kế giao diện phần mềm.
3. Lập trình xử lý phần mềm với các chức năng sau:
 - Đăng ký, đăng nhập và đăng xuất
 - Phát nhạc, chuyển bài, dừng nhạc, phát ngẫu nhiên, lặp bài hát, đặt giờ phát
 - Yêu thích bài hát, hiển thị bài hát đã yêu thích
 - Hiển thị thông tin nhạc sĩ, thể loại, tài khoản
 - Tìm kiếm bài hát
4. Nộp file nén (*.rar) lưu sản phẩm đề tài bao gồm:
 - File báo cáo word (*.docx)
 - File thuyết trình (*.pptx)
 - Thư mục chứa dự án (project), các class thư viện, CSDL, hình ảnh, ...)

Tp.HCM, ngày 13 tháng 6 năm 2022

GIẢNG VIÊN GIẢNG DẠY

(Ký và ghi rõ họ tên)

Nguyễn Tấn Toàn

BẢNG PHÂN CÔNG THỰC HIỆN ĐỒ ÁN MÔN HỌC

Họ tên SV1: NGUYỄN ĐỨC LONG MSSV: 20521566	Họ tên SV2: NGUYỄN TUẤN KHÔI MSSV: 20521480
1. Thiết kế: + Thiết kế các màn hình đăng nhập, đăng ký, trang chủ, danh sách nhạc, item bài hát, màn hình yêu thích	1. Thiết kế: + Logo ứng dụng + Thiết kế và làm cơ sở dữ liệu
2. Xử lý chức năng: + Màn hình đăng ký, logic màn hình, đăng ký, + Hiện thị danh sách bài hát theo yêu cầu + Màn hình trang chủ, tải quảng cáo, tải danh sách ca sĩ, danh sách thể loại danh sách +Xử lý màn hình danh sách bài hát yêu thích	2. Xử lý chức năng: + Đăng nhập, đăng ký, quên mật khẩu + Tìm kiếm bài hát theo tên bài hát hoặc ca sĩ + Thích/ bỏ thích bài hát + Truyền dữ liệu bài hát giữa các màn hình + Kết nối và lấy dữ liệu từ cơ sở dữ liệu
SV thực hiện 1 <i>(Ký tên)</i> NGUYỄN ĐỨC LONG	SV thực hiện 2 <i>(Ký tên)</i> NGUYỄN TUẤN KHÔI

Nhận xét và đánh giá của GV	Nhận xét và đánh giá của GV
------------------------------------	------------------------------------

Họ tên SV1: NGUYỄN BÁ KHANH MSSV: 20521450	
1. Thiết kế: + Màn hình User	
1. Xử lý chức năng: + Phát nhạc + Chuyển bài hát + Tạm dừng bài hát + Hiện thông báo khi phát nhạc + Xử lý các sự kiện trên thông báo phát nhạc + Phát lại bài hát + Phát ngẫu nhiên + Ẩn bài hát + Đặt giờ phát + Chỉnh sửa thông tin cá nhân + Liên hệ	

<p>SV thực hiện 3</p> <p><i>(Ký tên)</i></p> <p>NGUYỄN BÁ KHANH</p>	
<p>Nhận xét và đánh giá của GV</p>	

LỜI CẢM ƠN

Trước khi bước vào phần báo cáo đồ án, nhóm chúng em trân trọng gửi lời cảm ơn đến thầy **Nguyễn Tấn Toàn** đã tạo điều kiện cho chúng em có cơ hội được thực hiện đồ án và đã nhiệt tình giảng dạy trên lớp, cung cấp cho chúng em rất nhiều kiến thức hữu ích trong quá trình mới bắt đầu làm quen với môi trường Android Studio và cách thức lập trình tạo ra một ứng dụng với Android Studio. Đồng thời thầy cũng đã hỗ trợ những thông tin cần thiết, gợi mở những trường hợp sẽ gặp phải vấn đề khi lập trình trong thực tế, đưa ra những giải pháp tối ưu và giải đáp những thắc mắc không chỉ cho nhóm chúng em nói riêng và các bạn sinh viên khác trên lớp nói chung trong suốt quá trình học tập và thực hiện đồ án.

Với những kiến thức lý thuyết có đề cập tới trong các tuần học, nhóm em đã vận dụng được rất nhiều trong việc hoàn thành đề tài.

Trong vòng hơn 3 tháng qua, nhờ sự chỉ dẫn nhiệt tình của thầy, chúng em đã tiếp thu được những kiến thức quan trọng cùng những góp ý chân thành để có thể làm được một chương trình hoàn chỉnh.

Cũng xin cảm ơn thầy cô và bạn bè trong khoa Công nghệ phần mềm đã nhiệt tình hỗ trợ, tạo điều kiện cho nhóm em làm bài báo cáo này.

Trong quá trình học tập, cũng như là trong quá trình làm đồ án, nhóm em đã cố gắng hết sức mình, vì thời gian và kiến thức có hạn nên nhóm em không thể tránh khỏi những thiếu sót. Nhóm em mong nhận được sự thông cảm, đóng góp và xây dựng từ thầy để nhóm em có thêm nhiều kiến thức có ích và ngày càng hoàn thiện hơn.

Nhóm em xin chân thành cảm ơn ạ!

Nhóm thực hiện

Nguyễn Đức Long - Nguyễn Tuấn Khôi - Nguyễn Bá Khanh

Trường Đại học Công nghệ Thông tin, tháng 6 năm 2022

NHẬN XÉT CỦA GIẢNG VIÊN HƯỚNG DẪN

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

Tp.HCM, ngày 14 tháng 6 năm 2022

GVHD

Nguyễn Tấn Toàn

MỤC LỤC

BẢNG PHÂN CÔNG THỰC HIỆN ĐỒ ÁN MÔN HỌC	3
LỜI CẢM ƠN	6
NHẬN XÉT CỦA GIẢNG VIÊN HƯỚNG DẪN	8
Chương 1: GIỚI THIỆU ĐỀ TÀI	12
1.1 Giới thiệu đề tài	12
1.2 Mục tiêu đề tài	13
1.2.1 Lý thuyết	13
1.2.2 Thực tế	13
1.2.3 Các bước nghiên cứu	13
1.2.4 Bố cục đề tài	13
1.2.5 Các chức năng chính của đề tài	14
1.2.6 Công nghệ sử dụng	14
1.2.7 Môi trường lập trình	14
1.2.8 Công cụ hỗ trợ	14
Chương 2: CƠ SỞ LÝ THUYẾT	15
2.1. Hệ điều hành Android	15
2.1.1. Lịch sử Android	15
2.1.2. Kiến trúc của Android	16
2.1.3. Các thành phần của một ứng dụng Android	19
2.2. Hệ quản trị cơ sở dữ liệu SQLite	20
2.2.1. SQLite là gì?	20
2.2.2. Tính năng của SQLite	21
2.2.3. Ứng dụng chủ yếu của SQLite	21
2.3. Hệ quản trị cơ sở dữ liệu Firebase	22
2.3.1. Firebase là gì?	22
2.3.2. Cách thức hoạt động của Firebase	22
2.3.3. Ưu điểm của Firebase	23
2.3.4. Nhược điểm của Firebase	23
2.4. Công cụ mã nguồn mở PhpMyAdmin	24
2.4.1. PhpMyAdmin là gì?	24
2.4.2. Các tính năng của PhpMyAdmin	24

2.4.3.	Điểm yếu trong việc sao lưu dữ liệu của phpMyAdmin.....	24
2.4.4.	Ưu điểm của PhpMyAdmin.....	24
Chương 3: KHẢO SÁT VÀ XÁC ĐỊNH YÊU CẦU		26
3.1.	Phương pháp khảo sát	26
3.1.1.	Câu hỏi khảo sát.....	26
3.1.2.	Kết quả khảo sát.....	26
3.2.	Xác định yêu cầu	26
CHƯƠNG 4: PHÂN TÍCH YÊU CẦU		28
4.1.	Sơ đồ Use case	28
4.2.	Đặc tả Use-case.....	28
4.2.1.	Đăng kí	28
4.2.2.	Đăng nhập	30
4.2.3.	Đăng xuất	31
4.2.4.	Xem bài hát nổi bật	32
4.2.5.	Phát nhạc	32
4.2.8.	Chuyển bài kế tiếp.....	34
4.2.9.	Phát bài hát trước	35
4.2.10.	Phát lại	36
4.2.11.	Phát ngẫu nhiên.....	36
4.2.12.	Âm bài hát.....	37
4.2.13.	Hẹn giờ phát.....	38
4.2.14.	Yêu thích bài hát	38
4.2.15.	Tìm kiếm bài hát và ca sĩ	39
4.2.16.	Thay đổi ảnh đại diện.....	40
4.2.17.	Thay đổi thông tin người dùng	41
Chương 5: THIẾT KẾ DỮ LIỆU		43
5.1.	Khảo sát hiện trạng	43
5.2.	Mô tả tân từ	43
5.3.	Sơ đồ logic.....	43
5.4.	Mô tả chi tiết các kiểu dữ liệu	44
5.4.1.	Bảng bài hát	44
5.4.2.	Bảng nghệ sĩ.....	44

5.4.3	Bảng Album	45
5.4.4	Bảng chủ đề.....	45
5.4.5	Bảng thể loại	45
5.4.6	Bảng thích	45
5.4.7	Bảng quảng cáo.....	46
Chương 6: THIẾT KẾ KIẾN TRÚC		47
6.1.	Ứng dụng kiến trúc MVC để phát triển phần mềm	47
6.2	Mục đích lựa chọn.....	48
Chương 7: THIẾT KẾ GIAO DIỆN		49
7.1.	Sơ đồ liên kết các màn hình	49
7.2.	Danh sách các màn hình.....	49
7.3.	Mô tả các màn hình	50
7.3.1.	Màn hình chọn phương thức đăng nhập hoặc đăng ký	50
7.3.2.	Màn hình đăng nhập.....	50
7.3.3.	Màn hình quên mật khẩu.....	51
7.3.4.	Các màn hình đăng ký.....	52
7.3.5.	Màn hình trang chủ	53
7.3.6.	Màn hình danh sách bài hát	53
7.3.7.	Màn hình tìm kiếm.....	54
7.3.8.	Màn hình bài hát yêu thích.....	55
Chương 8: NHẬN XÉT VÀ KẾT LUẬN		58
8.1.	Ưu điểm đồ án	58
8.2.	Hạn chế của đồ án	58
8.3.	Hướng phát triển của đồ án	58
Tài liệu tham khảo		59

Chương 1: GIỚI THIỆU ĐỀ TÀI

1.1 Giới thiệu đề tài

Công nghệ đã thay đổi cách chúng ta sống và tương tác với những người khác theo nhiều cách. Ngày nay công nghệ dường như là một phần không thể thiếu trong cuộc sống của mỗi chúng ta. Nó đem đến cho ta vô số những lợi ích làm cho cuộc sống trở nên dễ dàng và vui vẻ hơn. Công nghệ luôn luôn thay đổi mang đến cho ta cảm giác mới mẻ. Một trong những thay đổi đáng chú ý nhất là cách mà âm nhạc được truyền tải đến người nghe. Sự ra đời và sự gia tăng của các dịch vụ phát trực tuyến mang đến cho người nghe cơ hội thưởng thức âm nhạc của họ bất cứ khi nào và bất cứ nơi nào họ muốn. Truyền phát nhạc cũng có giá cả phải chăng hơn cho những người hâm mộ thích nghe nhiều thể loại. Bạn không cần phải dừng lại ở một cửa hàng âm nhạc đặc biệt để chọn album mới nhất từ nghệ sĩ yêu thích của bạn. Thay vào đó, bạn có thể kéo nó lên ngay lập tức trên điện thoại thông minh hoặc máy tính xách tay.

Theo một khảo sát mới đây của Q&Me, 75% số người được khảo sát nghe nhạc hàng ngày, và là hình thức giải trí chính. 57% nghe vài lần một ngày, 19% nghe một lần một ngày, 12% nghe 4 – 6 lần/một tuần. Về hành vi nghe nhạc của người Việt Nam, khảo sát được thực hiện với 1.500 người từ độ tuổi 18 tới 44. Theo kết quả khảo sát, nghe nhạc là phương thức giải trí phổ biến nhất, các kênh nghe nhạc cũng trở nên linh hoạt và đa dạng hơn nhờ vào kết nối mạng ngày càng trở nên rộng rãi. Việc sử dụng các thiết bị điện tử để nghe nhạc cũng làm thay đổi thói quen lựa chọn kênh để nghe nhạc của người Việt. YouTube hẳn nhiên là kênh được nhiều người sử dụng nhất chiếm tới 86%, sau đó là Zing với 59%, Nhaccuatui chiếm 34%, Facebook chiếm 23%, Spotify một ứng dụng có bản quyền, người dùng phải trả tiền nổi lên ở Việt Nam khoảng 3 năm nay chiếm 14%, Apple chiếm 8%, sau đó là một số ứng dụng khác chiếm dưới 5%.

Điều gì khiến cho người sử dụng lựa chọn các ứng dụng để nghe nhạc? Ba yếu tố chính là chất lượng âm nhạc, sự đa dạng và việc cho tải xuống các bản nhạc là quan trọng nhất. Chất lượng âm nhạc là yếu tố hàng đầu được quan tâm chiếm tới 70%, sau đó là sự đa dạng của chủng loại âm nhạc chiếm tới 50%. Tuy hiện nay việc nghe nhạc online đã rất phổ biến, nhưng khả năng được tải nhạc, hay được nghe khi không kết nối cũng chiếm tới 43%. Tiếp theo đó là tính năng mang nhiều tính cá nhân hóa, tạo được danh sách nhạc cá nhân chiếm 26%, không hoặc ít quảng cáo chiếm 22%, tính phí hợp lý chiếm 18%, tốn ít dung lượng nghe nhạc chiếm 15% và cuối cùng là giao diện hấp dẫn chiếm 13%.

Cuộc sống của con người đang ngày càng được “số hóa” và ứng dụng nhiều công nghệ hơn. Trong số các quốc gia Đông Nam Á, Indonesia và Việt Nam là hai thị trường có lượng người dùng cao nhất. Việt Nam có 61,3 triệu người dùng, nằm trong top 10 quốc gia có số lượng người dùng smartphone lớn nhất thế giới. Khi điện thoại thông minh được “phổ cập”

thì sự phát triển của Mobile App – Ứng dụng di động cũng là một điều tất yếu. Ngày càng có nhiều ứng dụng di động ra đời.

Nắm bắt được xu thế hiện nay và trên cơ sở các kiến thức được học trong nhà trường, nhóm em đã quyết định chọn đề tài “Xây dựng ứng dụng nghe nhạc trực tuyến” trên nền tảng Android. Người dùng chỉ cần các thao tác đơn giản trên điện thoại di động là có thể trải nghiệm âm nhạc một cách vui vẻ.

1.2 Mục tiêu đề tài

1.2.1 Lý thuyết

- Nghiên cứu lập trình trên hệ điều hành Android.
- Nghiên cứu ngôn ngữ lập trình Java.
- Nghiên cứu xây dựng giao diện với XML trong Android
- Nắm rõ các thao tác trên môi trường phát triển tích hợp Android Studio.
- Nghiên cứu hệ quản trị cơ sở dữ liệu MySQL, Firebase.
- Tìm hiểu cách thức xây dựng ứng dụng trên nền tảng Android.

1.2.2 Thực tế

- Người dùng có thể lưu thông tin cá nhân.
- Liên kết đăng ký tài khoản với các mạng xã hội Facebook, Google.
- Ứng dụng hỗ trợ đầy đủ các tính năng của một ứng dụng nghe nhạc trực tuyến như chọn bài hát theo ca sĩ, theo thể loại, yêu thích bài hát ...
- Giao diện tiện lợi và dễ sử dụng.

1.2.3 Các bước nghiên cứu

- Phân tích yêu cầu đề tài.
- Lựa chọn công nghệ phù hợp với nhu cầu đề tài.
- Nghiên cứu cơ sở lý thuyết của công nghệ đã chọn
- Áp dụng lý thuyết vào xây dựng ứng dụng thực tiễn.
- Kiểm tra và khám phá các ứng dụng có chức năng tương ứng để tối ưu hóa và cải thiện các tính năng của ứng dụng.

1.2.4 Bố cục đề tài

- Cơ sở lý thuyết
- Phân tích đặc tả yêu cầu

- Phân tích hệ thống
- Thiết kế dữ liệu
- Thiết kế giao diện
- Cài đặt và thử nghiệm

1.2.5 Các chức năng chính của đề tài

- Đăng kí, đăng nhập, đăng xuất
- Thay đổi mật khẩu, ảnh, tên, tuổi, tên đăng nhập, ngày sinh, địa chỉ, số điện thoại, Email... nếu muốn.
- Quảng cáo bài hát trên phần trang chủ đưa người nghe hướng đến các bài hát của ứng dụng.
- Hiện thị thông tin bài hát như: Tên bài hát, tên ca sĩ ...
- Yêu thích bài hát, hiển thị danh sách yêu thích bài hát
- Phát nhạc, tạm dừng, tua bài hát, chuyển bài hát
- Hiện thị thông báo đang phát nhạc khi thoát ra khỏi ứng dụng
- Hiện thị danh sách ca sĩ, thể loại, album

1.2.6 Công nghệ sử dụng

- Sử dụng ngôn ngữ Java
- CSDL noSQL FireBase
- CSDL SQL phpMyAdmin

1.2.7 Môi trường lập trình

Android studio

1.2.8 Công cụ hỗ trợ

- GitHub: Quản lý source code
- Figma: Thiết kế giao diện
- Google Drive: Lưu trữ tài liệu
- Microsoft teams: Làm nhóm
- Android studio: Lập trình
- Microsoft Word: Viết báo cáo
- Microsoft Powerpoint: Làm slide báo cáo

Chương 2: CƠ SỞ LÝ THUYẾT

2.1. Hệ điều hành Android

2.1.1. Lịch sử Android

Ban đầu, Android là hệ điều hành cho các thiết bị Linux do công ty Android Inc. (California, Mỹ) thiết kế. Công ty này được Google mua lại vào năm 2005 và bắt đầu xây dựng Android Platform. Các thành viên chủ chốt ở Android Inc. gồm có: Andy Rubin, Rich Miner, Nick Sear, and Chris White.

Và sau tiếp, vào cuối năm 2007, thuộc Liên minh Thiết bị Cầm tay Mã Nguồn mở (Open Handset Alliance) gồm các thành viên nổi bật trong ngành viễn thông và thiết bị cầm tay như: Texas Instruments, Broadcom Corporation, Google, HTC, Intel, LG, Marvell Group, Motorola, Nvidia, Qualcomm, Samsung Electronics...

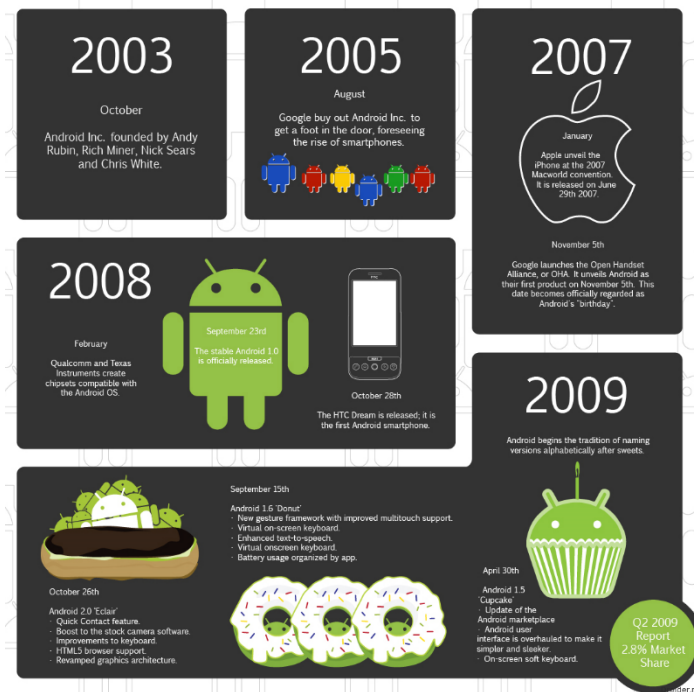
Mục tiêu của Liên minh này là nhanh chóng đổi mới để đáp ứng tốt hơn cho nhu cầu người tiêu dùng và kết quả đầu tiên của nó chính là nền tảng Android. Android được thiết kế để phục vụ nhu cầu của các nhà sản xuất thiết, các nhà khai thác và các lập trình viên thiết bị cầm tay.

Phiên bản SDK lần đầu tiên phát hành vào tháng 11 năm 2007, hãng T-Mobile cũng công bố chiếc điện thoại Android đầu tiên đó là chiếc T-Mobile G1, chiếc smartphone đầu tiên dựa trên nền tảng Android. Một vài ngày sau đó, Google lại tiếp tục công bố sự ra mắt phiên bản Android SDK release Candidate 1.0. Trong tháng 10 năm 2008, Google được cấp giấy phép mã nguồn mở cho Android Platform.

Khi Android được phát hành thì một trong số các mục tiêu trong kiến trúc của nó là cho phép các ứng dụng có thể tương tác được với nhau và có thể sử dụng lại các thành phần từ những ứng dụng khác. Việc tái sử dụng không chỉ được áp dụng cho các dịch vụ mà nó còn được áp dụng cho cả các thành phần dữ liệu và giao diện người dùng.

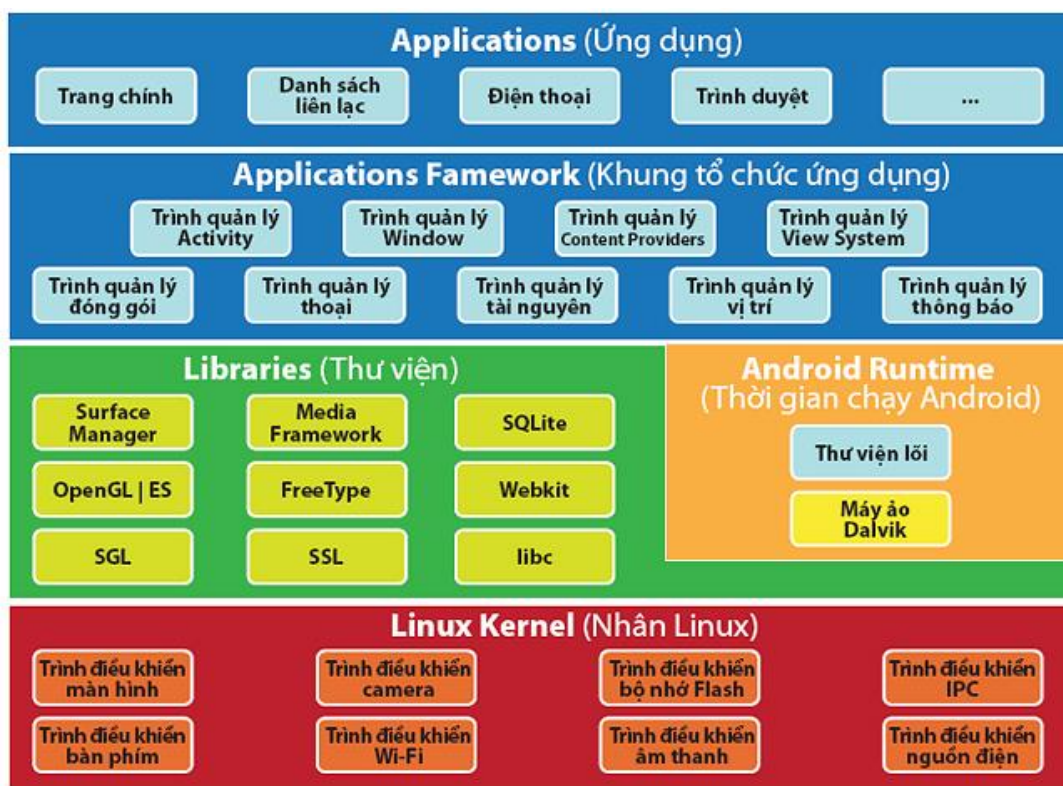
Vào cuối năm 2008, Google cho phát hành một thiết bị cầm tay được gọi là Android Dev Phone 1 có thể chạy được các ứng dụng Android mà không bị ràng buộc vào các nhà cung cấp mạng điện thoại di động. Mục tiêu của thiết bị này là cho phép các nhà phát triển thực hiện các cuộc thí nghiệm trên một thiết bị thực có thể chạy hệ điều hành Android mà không phải ký một bản hợp đồng nào. Vào. khoảng cùng thời gian đó thì Google cũng cho phát hành một phiên bản vá lỗi 1.1 của hệ điều hành này. Ở cả hai phiên bản 1.0 và 1.1 Android chưa hỗ trợ soft-keyboard mà đòi hỏi các thiết bị phải sử dụng bàn phím vật lý. Android

cố định vấn đề này bằng cách phát hành SDK 1.5 vào tháng Tư năm 2009, cùng với một số tính năng khác. Chẳng hạn như nâng cao khả năng ghi âm truyền thông, vật dụng, và các live folder...



2.1.2. Kiến trúc của Android

Mô hình sau thể hiện một cách tổng quát các thành phần của hệ điều hành Android:



Linux Kernel

Dưới cùng là lớp Linux - Linux 3.6 cùng với khoảng 115 bản vá. Lớp này cung cấp 1 cấp độ trừu tượng giữa phần cứng của thiết bị và các thành phần điều khiển phần cứng thiết yếu như máy ảnh, bàn phím, màn hình hiển thị... Đồng thời, hạt nhân (kernel) còn xử lý tất cả các thứ mà Linux có thể làm tốt như mạng kết nối và 1 chuỗi các trình điều khiển thiết bị, giúp cho giao tiếp với các thiết bị ngoại vi dễ dàng hơn.

Các thư viện

Ở trên lớp nhân Linux là tập các thư viện bao gồm WebKit - trình duyệt Web mã nguồn mở, được biết đến như thư viện libc, cơ sở dữ liệu SQLite - hữu dụng cho việc lưu trữ và chia sẻ dữ liệu ứng dụng, các thư viện chơi và ghi âm audio, video, hay các thư viện SSL chịu trách nhiệm bảo mật Internet...

Các thư viện Android

Đây là các thư viện dựa trên Java phục vụ cho việc phát triển Android. Ví dụ của các thư viện này bao gồm các thư viện ứng dụng dùng để xây dựng giao diện người dùng, vẽ đồ họa hay truy cập cơ sở dữ liệu. 1 số thư viện chính của Android:

android.app - Cung cấp quyền truy cập vào ứng dụng và là nền tảng của tất cả ứng dụng Android.

android.content - Cung cấp quyền truy cập nội dung (content), truyền tải thông điệp giữa các ứng dụng hay các thành phần của ứng dụng.

android.database - Được sử dụng để truy cập dữ liệu của content provider và cơ sở dữ liệu SQLite

android.opengl - giao diện các phương thức Java để sử dụng OpenGL ES

android.os - Cung cấp các ứng dụng với quyền truy cập vào các dịch vụ của hệ điều hành bao gồm thông điệp, các dịch vụ hệ thống và các giao tiếp nội tại

android.text - Được sử dụng để hiển thị và điều chỉnh chữ trên màn hình thiết bị

android.view - Các thành phần cơ bản trong việc xây dựng giao diện người dùng của ứng dụng.

android.widget - Tập các thành phần giao diện người dùng đã được xây dựng sẵn như các nút, các nhãn (label), list view,....

Android.webkit - Tập các lớp cho phép xây dựng khả năng duyệt web.

Android Runtime

Đây là phần thứ 3 của kiến trúc và nằm ở lớp thứ 2 từ dưới lên. Phần này cung cấp 1 bộ phận quan trọng là Dalvik Virtual Machine - là 1 loại Java Virtual Machine được thiết kế đặc biệt để tối ưu cho Android.

Dalvik VM sử dụng các đặc trưng của nhân Linux như quản lý bộ nhớ và đa luồng, những thứ mà đã có sẵn trong Java. Dalvik VM giúp mở ứng dụng Android chạy trong tiến trình riêng của nó, với các thể hiện (instance) riêng của Dalvik virtual Machine.

Android Runtime cũng cung cấp 1 tập các thư viện chính giúp các nhà phát triển ứng dụng Android có thể viết ứng dụng Android bằng Java

Application Framework

Lớp Android Framework cung cấp các dịch vụ cấp độ cao hơn cho các ứng dụng dưới dạng các lớp Java. Các nhà phát triển ứng dụng được phép sử dụng các dịch vụ này trong ứng dụng của họ.

Android Framework bao gồm các dịch vụ chính sau:

Activity Manager - Kiểm soát tất cả khía cạnh của vòng đời ứng dụng và ngăn xếp các Activity.

Content Providers - Cho phép các ứng dụng chia sẻ dữ liệu với các ứng dụng khác.

Resource Manager - Cung cấp quyền truy cập vào các tài nguyên như các chuỗi, màu sắc, các layout giao diện người dùng...

Notifications Manager - Cho phép các ứng dụng hiển thị cảnh báo và các thông báo cho người dùng.

View System - Tập các thành phần giao diện (view) được sử dụng để tạo giao diện người dùng.

Application

Lớp trên cùng của kiến trúc là Application. Các ứng dụng bạn tạo ra sẽ được cài đặt trên lớp này. Ví dụ như: Danh bạ, nhắn tin, trò chơi...

2.1.3. Các thành phần của một ứng dụng Android

Trong Android, chúng ta có thể liệt kê ra 4 thành phần cơ bản như sau:

- **Activity**
- **Service**
- **Broadcast Receiver**
- **Content Provider**

Bây giờ, mình sẽ cùng các bạn tìm hiểu chi tiết và ứng dụng của các thành phần trên.

Activity

Trong ứng dụng Android, Activity đóng vai trò đặc biệt quan trọng, là nơi giúp người dùng tương tác trực tiếp với ứng dụng, ví dụ như gọi điện thoại, chụp ảnh, gửi email hoặc xem bản đồ. Activity được coi là xương sống của một ứng dụng Android, một ứng dụng có thể có một hoặc nhiều Activity (bất kỳ ứng dụng nào cũng cần có ít nhất 1 Activity).

Activity có thể hiển thị ở chế độ toàn màn hình, dạng cửa sổ hoặc với một kích thước nhất định. Một Activity có thể gọi đến một Activity khác, Activity được gọi đến sẽ tương tác với người dùng tại thời điểm được gọi tới. Một ứng dụng bên ngoài có thể gọi tới bất kỳ Activity nào trong ứng dụng (nếu được cấp quyền).

Ví dụ: Một ứng dụng chụp ảnh sau khi chụp ảnh xong, sẽ gửi yêu cầu để start một activity có chức năng soạn email trong ứng dụng email nhằm mục đích gửi ảnh vừa chụp đi.

Service

Service là một thành phần ứng dụng chạy ngầm trên hệ điều hành ví dụ như nghe nhạc, hoặc tương tác với một content provider. Service không tương tác trực tiếp với người dùng,

khi service chạy thì người dùng vẫn có thể tương tác với một thành phần khác trong ứng dụng hoặc có thể tương tác với một ứng dụng khác trong hệ thống.

Ví dụ: Chúng ta có thể vừa nghe nhạc, vừa lướt facebook là do ứng dụng nghe nhạc có một service chạy ngầm trong background để phát nhạc trong khi người dùng đang tương tác với ứng dụng facebook.

Theo trang chủ android, Service trong Android được chia thành 3 loại đó là: Foreground Service, Background Service và Bound Service.

Broadcast Receiver

Broadcast Receiver là một thành phần của ứng dụng giúp lắng nghe các sự kiện mà hệ thống phát ra thông qua Intent, hệ thống có thể truyền phát ngay cả khi app không chạy. Broadcast Receiver không có giao diện cụ thể nhưng nó có thể thực hiện thông báo thông qua thanh Notification. Có rất nhiều broadcast được phát ra từ hệ thống, chúng ta có thể lấy ví dụ như một broadcast thông báo rằng màn hình điện thoại đã tắt, hay điện thoại đang ở trạng thái “Battery Low”, “Power Connected”, “Power Disconnected” hoặc một bức ảnh đã được chụp. Cũng có những broadcast được phát ra từ ứng dụng như sau khi download một tệp, ví dụ: Sau khi hoàn thành download một tệp tin, ứng dụng A phát ra thông báo là dữ liệu đã download xong, tệp đã sẵn sàng cho các ứng dụng khác có thể sử dụng.

Content Provider

Content Provider là một thành phần giúp các ứng dụng có thể đọc và ghi dữ liệu từ một file hoặc từ SQLite của một ứng dụng khác trong cùng một hệ thống. Bất kỳ ứng dụng nào có quyền (permission) đều có thể truy xuất, chỉnh sửa dữ liệu của một ứng dụng khác.

Content Provider được chia thành 2 loại:

Native Content Provider: Là những Content Provider có sẵn, được tạo ra bởi hệ thống, ví dụ như Contacts, Message, ...

Custom Content Provider: Bao gồm các Content Provider được tạo ra bởi các developer phụ thuộc vào đặc điểm của từng ứng dụng.

2.2. Hệ quản trị cơ sở dữ liệu SQLite

2.2.1. SQLite là gì?

SQLite là hệ quản trị cơ sở dữ liệu (DBMS) quan hệ tương tự như Mysql, ... Đặc điểm nổi bật của SQLite so với các DBMS khác là gọn, nhẹ, đơn giản, đặc biệt không cần mô hình server-client, không cần cài đặt, cấu hình hay khởi động nên không có khái niệm user, password hay quyền hạn trong SQLite Database. Dữ liệu cũng được lưu ở một file duy nhất.

SQLite thường không được sử dụng với các hệ thống lớn nhưng với những hệ thống ở quy mô vừa và nhỏ thì SQLite không thua các DBMS khác về chức năng hay tốc độ. Vì không cần cài đặt hay cấu hình nên SQLite được sử dụng nhiều trong việc phát triển, thử nghiệm ... vì tránh được những rắc rối trong quá trình cài đặt.

2.2.2. Tính năng của SQLite

- Giao dịch trong SQLite tuân thủ theo nguyên tắc (ACID) ngay cả sau khi hệ thống treo và mất điện.
- Không cấu hình
- SQLite hỗ trợ với đầy đủ tính năng với các khả năng nâng cao như các chỉ mục 1 phần, các chỉ mục về các biểu thức, JSON và các biểu thức bảng chung.
- Một sở dữ liệu hoàn chỉnh được lưu trữ trong một tệp đa nền tảng duy nhất. Phù hợp với sử dụng dưới dạng định dạng tệp ứng dụng
- Hỗ trợ các cơ sở dữ liệu có kích thước terabyte và các chuỗi có kích thước gigabyte.
- API
 - Đơn giản dễ sử dụng
- Nhanh: Trong một số trường hợp, SQLite nhanh hơn hệ thống tệp tin trực tiếp I/O.
- Được viết bằng ANSI-C.
- Bindings cho hàng chục ngôn ngữ khác có sẵn 1 cách riêng biệt.
- Mã nguồn đầy, nguồn mở đủ có thể kiểm tra nhánh 100%.
- Nền tảng đa nền tảng:
- SQLite là có sẵn trên Android, *BSD, iOS, Linux, Mac, Solaris, Windows... Dễ dàng dịch chuyển sang các hệ thống khác.

2.2.3. Ứng dụng chủ yếu của SQLite

1. Cơ sở dữ liệu cho Internet of Things.

- SQLite là lựa chọn phổ biến cho các công cụ cơ sở dữ liệu trong điện thoại di động, PDA, máy nghe nhạc mp3, hộp set-top, và các tiện ích điện tử khác.

2. Định dạng tệp ứng dụng.

- Thay vì sử dụng fopen() để viết XML, JSON, CSV hoặc một số định dạng động quyền vào các tệp đĩa được ứng dụng của bạn sử dụng, hãy sử dụng SQLite.

3. Cơ sở dữ liệu cho web.

- Bởi vì SQLite không yêu cầu cấu hình và lưu trữ thông tin trong các tệp đĩa thông thường nên SQLite là lựa chọn phổ biến làm cơ sở dữ liệu để quay lại các trang web vừa và nhỏ.

4. Stand-in cho một doanh nghiệp RDBMS:

- SQLite được sử dụng như một RDBMS doanh nghiệp cho mục đích trình diễn hoặc thử nghiệm vì SQLite nhanh và không yêu cầu thiết lập.

2.3. Hệ quản trị cơ sở dữ liệu Firebase

2.3.1. Firebase là gì?

Firebase là một nền tảng mà Google cung cấp để phát triển các ứng dụng di động và trang web, hỗ trợ Developer trong việc đơn giản hóa các thao tác với dữ liệu và không cần tác động tới backend hay server.

2.3.2. Cách thức hoạt động của Firebase

Sau khi được Google mua lại và tiến hành phát triển, Firebase thực hiện nhiều hoạt động hỗ trợ. Dưới đây là 3 cách thức hoạt động nổi bật của firebase.

Firestore Realtime Database

Đầu tiên, để sử dụng nền tảng này tạo ứng dụng, lập trình viên cần đăng ký tài khoản Firebase. Đăng ký xong, bạn sẽ sở hữu một Realtime Database có dạng JSON được đồng bộ thời gian đến tất cả kết nối client.

Dữ liệu trong các database được tự động cập nhật liên tục khi lập trình viên tiến hành phát triển ứng dụng. Sau khi được cập nhật, những dữ liệu này sẽ được truyền tải thông qua kết nối SSL có chứng nhận 2048 bit.

Tại những ứng dụng đa nền tảng toàn bộ máy khách đều dùng chung một database. Trong trường hợp kết nối Internet bị gián đoạn, các dữ liệu sẽ được lưu trữ trên local và cập nhật khi đường truyền ổn định.

Firestore Authentication

Hoạt động kế tiếp của Firebase là tạo lập quy trình xác thực người dùng thông qua các phương tiện như Google, Email, GitHub, Facebook, Twitter và xác thực nặc danh đối với các ứng dụng. Việc xác thực giúp bảo vệ an toàn dữ liệu về thông tin cá nhân của người dùng mạng.

Firestore Hosting

Các thức hoạt động tiêu biểu cuối cùng, không thể không nhắc đến của Firebase là cung cấp hosting. Firebase hosting thông qua sự hỗ trợ của SSL đến từ mạng CDN. Nhờ vậy mà lập trình viên có thể tiết kiệm rất nhiều thời gian thiết kế cũng như phát triển ứng dụng.

2.3.3. Ưu điểm của Firebase

- Tốc độ cao: Firebase hỗ trợ phát triển ứng dụng với tốc độ nhanh chóng, giảm bớt thời gian phát triển và tiếp thị ứng dụng.
- Đơn giản, dễ dàng trong sử dụng: Người dùng có thể tạo tài khoản Firebase bằng tài khoản Google và cũng như sử dụng nền tảng này trong phát triển ứng dụng một cách dễ dàng.
- Một nền tảng, đa dịch vụ: Firebase cung cấp đầy đủ các dịch vụ hỗ trợ phát triển web, bạn có thể chọn database Firestore hoặc Realtime theo ý muốn.
- Là một phần của Google: Firebase đã được mua lại và hiện tại đang là một phần của Google. Nó khai thác triệt để sức mạnh và các dịch vụ sẵn có của Google.
- Tập trung phát triển chủ yếu về giao diện người dùng: Firebase cho phép lập trình viên tập trung chủ yếu vào phát triển phần giao diện người dùng nhờ kho Backend mẫu phong phú đa dạng.
- Không có máy chủ: Điều này giúp Firebase có khả năng tối ưu hóa về hiệu suất làm việc thông qua mở rộng cụm database.
- Học máy: Firebase cung cấp học máy cho lập trình viên, hỗ trợ tối đa cho việc phát triển ứng dụng
- Tạo ra lưu lượng truy cập: Với khả năng hỗ trợ tạo lập các chỉ mục, Firebase giúp nâng cao thứ hạng của ứng dụng trên bảng xếp hạng Google, từ đó tăng lượt traffic.
- Theo dõi lỗi: Firebase là công cụ phát triển, khắc phục lỗi tuyệt vời.
- Sao lưu: Firebase sao lưu thường xuyên, giúp đảm bảo tính sẵn có và bảo mật thông tin cho dữ liệu.

2.3.4. Nhược điểm của Firebase

- Không phải mã nguồn mở
- Chỉ hoạt động trên CSDL NoSQL
- Lập trình viên không được truy cập mã nguồn
- Truy vấn chậm
- Phạm vi hoạt động bị hạn chế ở một vài quốc gia
- Một vài dịch vụ cần trả phí, chi phí dành cho Firebase tương đối lớn và không ổn định
- Thiếu Dedicated Servers và cả hợp đồng doanh nghiệp
- Không cung cấp API GraphQL

- Chỉ có thể chạy trên Google Cloud
- Thiếu Dedicated Servers và cả hợp đồng doanh nghiệp.

2.4. Công cụ mã nguồn mở PhpMyAdmin

2.4.1. PhpMyAdmin là gì?

PhpMyAdmin là một công cụ nguồn mở được viết bằng ngôn ngữ PHP để giúp người dùng quản lý cơ sở dữ liệu của MySQL thông qua một trình duyệt web. Đây là công cụ quản trị MySQL phổ biến nhất được sử dụng bởi hàng triệu người dùng trên toàn thế giới, đặc biệt là các nhà quản trị cơ sở dữ liệu hay database administrator. Thay vì sử dụng giao diện cửa sổ dòng lệnh (command line interface), với trình duyệt web của mình, thông qua giao diện người dùng (user interface), phpMyAdmin có thể thực hiện nhiều tác vụ như tạo, cập nhật, sửa đổi hoặc xóa bỏ cơ sở dữ liệu, bảng hoặc bản ghi; thực hiện báo cáo SQL; phân quyền và quản lý người dùng...

2.4.2. Các tính năng của PhpMyAdmin

Một số tính năng chung thường được sử dụng trên **phpMyAdmin**:

- Quản lý user (người dùng): thêm, xóa, sửa(phân quyền).
- Quản lý cơ sở dữ liệu: tạo mới, xóa, sửa, thêm bảng, hàng, trường, tìm kiếm đối tượng.
- Nhập xuất dữ liệu (Import/Export): hỗ trợ các định dạng SQL, XML và CSV.
- Thực hiện các truy vấn MySQL, giám sát quá trình và theo dõi.
- Sao lưu và khôi phục (Backup/Restore): Thao tác thủ công.

2.4.3. Điểm yếu trong việc sao lưu dữ liệu của phpMyAdmin

- Chức năng export/import của phpMyAdmin thiếu rất nhiều tính năng mà bạn mong muốn:

- Lập kế hoạch (Scheduling): Với phpMyAdmin, không có cách nào để tự động xuất database data.

- Hỗ trợ lưu trữ phương tiện truyền thông (Storage media support): Vì phpMyAdmin là một phần mềm dựa trên web nên bạn chỉ có thể làm việc với nó thông qua trình duyệt. Điều này có nghĩa là bạn chỉ có thể lưu các bản sao lưu vào các local drive có sẵn trên hệ thống của mình, thông qua hộp thoại Save As... của trình duyệt.

- Nén, mã hóa và các tùy chọn khác: Các tệp được xuất bằng phpMyAdmin được lưu dưới dạng text files phổ biến, không cần xử lý thêm. Lưu trữ ở dạng ban đầu sẽ khiến chúng chiếm rất nhiều dung lượng đĩa và không an toàn.

2.4.4. Ưu điểm của PhpMyAdmin

Cộng đồng hỗ trợ rộng lớn: Người sử dụng sẽ nhận được sự hỗ trợ rất lớn từ cộng đồng vì công cụ phpMyAdmin có tính chất giống với mã nguồn mở được phát triển bởi các lập trình viên trên toàn cầu.

Tăng hiệu quả công tác quản lý cơ sở dữ liệu: phpMyAdmin mang đến giao diện xử lý các thao tác trên cơ sở dữ liệu một cách trực quan. Từ đó, tiết kiệm thời gian, thao tác so với việc thực hiện bằng dòng lệnh trên command line.

Đa ngôn ngữ: Hiện nay phpMyAdmin có sẵn đến 64 ngôn ngữ khác nhau và được duy trì bởi The phpMyAdmin Project.

Chi phí sử dụng: phpMyAdmin vẫn là công cụ được sử dụng miễn phí hoàn toàn.

Chương 3: KHẢO SÁT VÀ XÁC ĐỊNH YÊU CẦU

3.1. Phương pháp khảo sát

- Quá trình khảo sát được tiến hành online. Bằng google form.

3.1.1. Câu hỏi khảo sát

- Câu 1: Bạn thường làm gì vào lúc rảnh rỗi
- Câu 2: Bạn thường nghe nhạc trên thiết bị nào
- Câu 3: Bạn thường nghe nhạc trên ứng dụng nào
- Câu 4: Bạn thích điều gì ở ứng dụng nghe nhạc bạn đang sử dụng
- Câu 5: Bạn thích gam màu nóng hay lạnh
- Câu 6: Cách phối màu của hình ảnh nào dưới đây làm bạn hài lòng nhất (đính kèm hình ảnh)? *
- Câu 7: Bạn có sẵn sàng trả phí để có một trải nghiệm âm nhạc tốt hơn hay không?

3.1.2. Kết quả khảo sát

- Hơn 70% người được khảo sát trả lời rằng họ thường nghe nhạc vào những lúc rảnh rỗi tiếp theo là hơn 50% sẽ xem phim vào lúc rảnh rỗi
 - Hơn 75% người được khảo sát trả lời rằng họ thường nghe nhạc trên smartphone và 20% sử dụng laptop để nghe nhạc và 5% còn lại nghe nhạc trên các thiết bị khác
 - Hầu hết mọi người thường sử dụng spotify là ứng dụng nghe nhạc chính con số này chiếm tới 60% và 40% còn lại chia đều cho các nền tảng phổ biến như zingMp3, Youtube
- Hơn 70% trả lời rằng họ thích ứng dụng nghe nhạc phải có giao diện đẹp, trải nghiệm người dùng mượt mà và dễ sử dụng
- Có vẻ như gam màu lạnh được yêu thích nhiều hơn, kết quả là hơn 60% người được khảo sát thích màu lạnh hơn
 - Theo khảo sát thì mọi người thích những thứ đơn giản hơn trên một ứng dụng con số này là hơn 70%.
 - Mọi người có ngày càng có những xu hướng trả tiền để có những trải nghiệm để chịu không bị quảng cáo làm phiền hay để ủng hộ đội ngũ phát triển. Thật vậy con số hơn 80% đã trả lời cho phát biểu trên

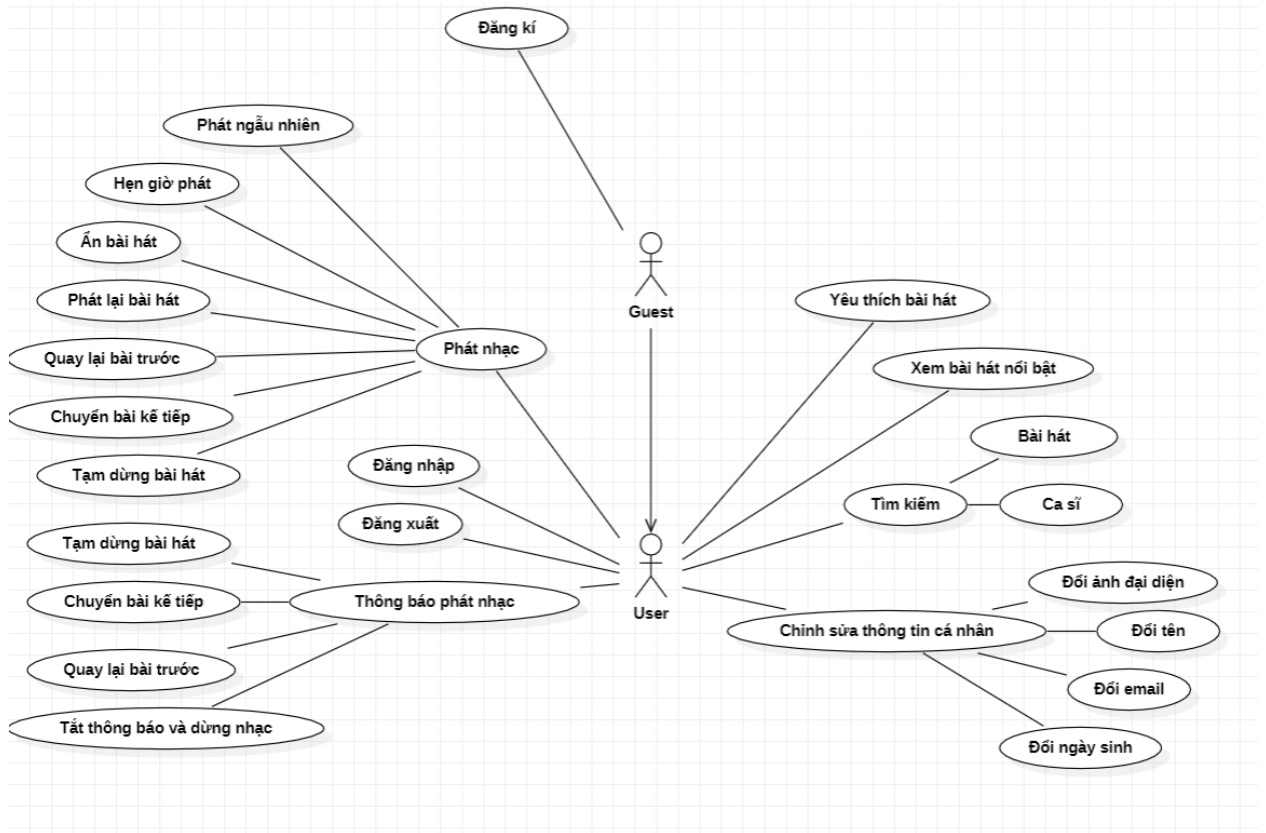
3.2. Xác định yêu cầu

- Xây dựng một ứng dụng nghe nhạc hỗ trợ nhiều tính năng đi kèm.

- Hướng đến một tone màu sáng chủ đạo, giao diện thân thiện không quá bắt mắt, tạo cho người dùng thói quen nghe nhạc trên ứng dụng
- Tạo cảm giác thoải mái dễ chịu khi thưởng thức âm nhạc trên ứng dụng

CHƯƠNG 4: PHÂN TÍCH YÊU CẦU

4.1. Sơ đồ Use case



4.2. Đặc tả Use-case

4.2.1. Đăng kí

Use Case ID	UC1
Tên	Đăng ký
Mô tả tóm tắt	Cho phép người dùng tạo tài khoản để sử dụng các chức năng khác của ứng dụng
Người thực hiện (Actors)	Khách (Guest)
Sự kiện kích hoạt (Trigger)	Bấm chọn nút Sign up
Điều kiện tiên quyết (Pre-condition)	Tài khoản mới và thông tin người dùng được lưu vào hệ thống

Điều kiện kết quả (Post-condition)	Ứng dụng điều hướng đến màn hình Home
Kịch bản chính (Basic flow)	<ol style="list-style-type: none"> 1. Người dùng bấm chọn nút Sign up (Đăng ký) 2. Hệ thống hiện bản nhập liệu email 3. Nhập email và bấm Next 4. Hệ thống kiểm tra tính hợp lệ của email và chuyển tiếp tới màn hình nhập mật khẩu và xác nhận mật khẩu 5. Nhập đầy đủ 2 trường mật khẩu và bấm Next 6. Hệ thống kiểm tra mật khẩu có đủ mạnh không và có trùng với nhau không và chuyển đến màn hình nhập ngày sinh 7. Nhập thông tin ngày sinh và bấm Next 8. Hệ thống kiểm tra thông tin ngày sinh hợp lệ và chuyển tiếp tới màn hình nhập giới tính 9. Chọn giới tính và bấm Next 10. Chuyển tiếp tới màn hình nhập tên người dùng 11. Nhập tên người dùng và nhấn Next 12. Kiểm tra tên người dùng hợp lệ 13. Hệ thống lưu lại thông tin tài khoản và chuyển đến màn hình Home

Kịch bản thay thế (Alternative flow)	<p>3a.1. Người dùng nhập sai hoặc để trống email và bấm Next</p> <p>3a.2. Hệ thống báo email không hợp lệ hoặc để trống</p> <p>3b.1. Người dùng nhập email đã được đăng ký trước đó</p> <p>3b.2. Hệ thống thông báo email đã tồn tại trong hệ thống và yêu cầu nhập lại</p> <p>5a.1. Người dùng nhập mật khẩu không đủ 8 ký tự</p> <p>5a.2. Hệ thống thông báo mật khẩu không đủ 8 ký tự</p> <p>5b.1. Người dùng nhập 2 trường mật khẩu khác nhau</p> <p>5b.2. Hệ thống thông báo mật khẩu không trùng khớp và yêu cầu nhập lại.</p> <p>6.1. Người dùng nhập ngày sinh lớn hơn ngày hiện tại</p> <p>6.2 Hệ thống thông báo yêu cầu nhập lại ngày sinh</p> <p>11.1. Người dùng nhập tên người dùng không hợp lệ (có lý tự đặc biệt)</p> <p>11.2. Hệ thống thông báo tên người dùng không hợp lệ và yêu cầu nhập lại.</p>
---	---

4.2.2. Đăng nhập

Use Case ID	UC2
Tên	Đăng nhập
Mô tả tóm tắt	Cho phép người dùng đăng nhập vào ứng dụng với tài khoản đã được tạo để sử dụng nhiều chức năng hơn của ứng dụng
Người thực hiện (Actors)	Người dùng (User).
Sự kiện kích hoạt (Trigger)	Bấm chọn nút Sign in
Điều kiện tiên quyết (Pre-condition)	Tài khoản đã được tạo và mật khẩu phải hợp lệ

Điều kiện kết quả (Post-condition)	Ứng dụng điều hướng đến màn hình Home
Kịch bản chính (Basic flow)	<ol style="list-style-type: none"> 1. Người dùng nhập thông tin email, mật khẩu rồi bấm Signin 2. Hệ thống kiểm tra tính hợp lệ của tài khoản 3. Ứng dụng điều hướng đến màn hình Home
Kịch bản thay thế (Alternative flow)	<ol style="list-style-type: none"> 2.1. Người dùng nhập sai thông tin đăng nhập 2.2. Hệ thống thông báo tài khoản hoặc mật khẩu không hợp lệ, đăng nhập lại
Mở rộng	Đăng nhập bằng tài khoản Google, Facebook

4.2.3. Đăng xuất

Use Case ID	UC3
Tên	Đăng xuất
Mô tả tóm tắt	Cho phép người dùng thoát khỏi tài khoản hiện tại trên ứng dụng
Người thực hiện (Actors)	Người dùng (User)
Sự kiện kích hoạt (Trigger)	Bấm chọn nút Sign out
Điều kiện tiên quyết (Pre-condition)	Người dùng phải đăng nhập trước đó
Điều kiện kết quả (Post-condition)	Ứng dụng điều hướng đến màn hình Đăng nhập
Kịch bản chính (Basic flow)	<ol style="list-style-type: none"> 1. Người dùng bấm nút Sign out tại màn hình ... 2. Hệ thống đăng xuất tài khoản và điều hướng đến màn hình Đăng nhập.

Kịch bản thay thế (Alternative flow)	
---	--

4.2.4. Xem bài hát nổi bật

Use Case ID	UC4
Tên	Xem bài hát nổi bật
Mô tả tóm tắt	Cho phép người dùng xem các bài hát nổi bật trên hệ thống
Người thực hiện (Actors)	Người dùng (User)
Sự kiện kích hoạt (Trigger)	Điều hướng đến màn hình Home
Điều kiện tiên quyết (Pre-condition)	Người dùng phải đăng nhập trước đó
Điều kiện kết quả (Post-condition)	Hệ thống hiển thị các bài hát nổi bật trên hệ thống

4.2.5. Phát nhạc


Use Case ID	UC5
Tên	Phát nhạc
Mô tả tóm tắt	Cho phép người dùng nghe bản nhạc đã chọn
Người thực hiện (Actors)	Người dùng (User)
Sự kiện kích hoạt (Trigger)	Bấm chọn 1 bài hát bất kì
Điều kiện tiên quyết (Pre-condition)	Người dùng phải đăng nhập trước đó

Điều kiện kết quả (Post-condition)	Ứng dụng điều hướng đến màn hình phát nhạc
Kịch bản chính (Basic flow)	<ol style="list-style-type: none"> 1. Người dùng chọn 1 danh sách phát trên màn hình ứng dụng sau đó chọn vào bài hát 2. Hệ thống lấy danh sách bài hát mà người dùng đã chọn và phát bài hát người dùng đã chọn
Kịch bản thay thế (Alternative flow)	


4.2.6. Hiện thông báo phát nhạc


Use Case ID	UC6
Tên	Hiện thông báo phát nhạc
Mô tả tóm tắt	Hiện thông báo phát nhạc trên thanh thông báo khi người dùng phát một bài hát
Người thực hiện (Actors)	Người dùng (User)
Sự kiện kích hoạt (Trigger)	Người dùng chọn phát một bài hát bất kì
Điều kiện tiên quyết (Pre-condition)	Người dùng đã đăng nhập
Điều kiện kết quả (Post-condition)	Hiện thông báo trên thanh thông báo
Mở rộng	<ol style="list-style-type: none"> 1. Chức năng tạm dừng/phát nhạc 2. Chức năng chuyển bài hát kế tiếp 3. Chức năng quay lại bài hát trước 4. Chức năng dừng phát nhạc và tắt thông báo

4.2.7. Dừng phát nhạc và tắt thông báo



Use Case ID	UC7
Tên	Dừng phát nhạc và tắt thông báo
Mô tả tóm tắt	Tắt thông báo trên thanh thông báo và dừng phát nhạc
Người thực hiện (Actors)	Người dùng (User)
Sự kiện kích hoạt (Trigger)	Người dùng chọn  trên thông báo phát nhạc
Điều kiện tiên quyết (Pre-condition)	Người dùng đã đăng nhập và nhạc đang phát hoặc tạm dừng
Điều kiện kết quả (Post-condition)	Tắt thông báo trên thanh thông báo và dừng phát nhạc

4.2.8. Chuyển bài kế tiếp



Use Case ID	UC8
Tên	Chuyển bài kế tiếp
Mô tả tóm tắt	Cho phép người dùng chuyển tới bài hát kế tiếp trong danh sách đã chọn
Người thực hiện (Actors)	Người dùng (User)
Sự kiện kích hoạt (Trigger)	Bấm chọn biểu tượng  (next) trên màn hình phát nhạc
Điều kiện tiên quyết (Pre-condition)	Người dùng phải đang phát nhạc
Điều kiện kết quả (Post-condition)	Hệ thống chuyển và phát bài hát kế tiếp trong danh sách

Kịch bản chính (Basic flow)	<ol style="list-style-type: none"> 1. Người dùng bấm nút  tại màn hình phát nhạc 2. Hệ thống dừng bài hát hiện tại và chuyển tới bài hát kế tiếp trong danh sách
Kịch bản thay thế (Alternative flow)	<ol style="list-style-type: none"> 1. Nếu bài hát hiện tại là bài hát cuối cùng trong danh sách 2. Phát bài hát đầu tiên trong danh sách

4.2.9. Phát bài hát trước



Use Case ID	UC9
Tên	Chuyển bài trước
Mô tả tóm tắt	Cho phép người dùng chuyển tới bài hát trước trong danh sách đã chọn
Người thực hiện (Actors)	Người dùng (User)
Sự kiện kích hoạt (Trigger)	Bấm chọn biểu tượng  (previous) trên màn hình phát nhạc
Điều kiện tiên quyết (Pre-condition)	Người dùng phải đang phát nhạc
Điều kiện kết quả (Post-condition)	Hệ thống chuyển và phát bài hát trước trong danh sách
Kịch bản chính (Basic flow)	<ol style="list-style-type: none"> 1. Người dùng bấm nút  tại màn hình phát nhạc 2. Hệ thống dừng bài hát hiện tại và chuyển tới bài hát trước trong danh sách
Kịch bản thay thế (Alternative flow)	<ol style="list-style-type: none"> 1. Nếu bài hát hiện tại là bài hát đầu tiên trong danh sách 2. Phát bài hát cuối cùng trong danh sách

4.2.10. Phát lại

Use Case ID	UC10
Tên	Phát lại bài hát
Mô tả tóm tắt	Cho phép người dùng phát lại một bài hát cho đến khi người dùng tắt
Người thực hiện (Actors)	Người dùng (User)
Sự kiện kích hoạt (Trigger)	Bấm chọn biểu tượng  (repeat) trên màn hình phát nhạc
Điều kiện tiên quyết (Pre-condition)	Người dùng phải đang phát nhạc
Điều kiện kết quả (Post-condition)	Hệ thống phát lại bài hát hiện tại
Kịch bản chính (Basic flow)	<ol style="list-style-type: none"> 1. Người dùng bấm biểu tượng  (repeat) tại màn hình phát nhạc 2. Hệ thống tiếp tục phát bài hát hiện tại cho tới khi người dùng bấm lại biểu tượng này hoặc tắt nhạc
Kịch bản thay thế (Alternative flow)	


4.2.11. Phát ngẫu nhiên

Use Case ID	UC11
Tên	Phát ngẫu nhiên
Mô tả tóm tắt	Cho phép người dùng phát ngẫu nhiên các bài hát trong danh sách


Người thực hiện (Actors)	Người dùng (User)
Sự kiện kích hoạt (Trigger)	Bấm chọn biểu tượng  (shuffle) trên màn hình phát nhạc
Điều kiện tiên quyết (Pre-condition)	Người dùng phải đang phát nhạc
Điều kiện kết quả (Post-condition)	Hệ thống phát ngẫu nhiên các bài hát trong danh sách
Kịch bản chính (Basic flow)	<ol style="list-style-type: none"> 1. Người dùng bấm biểu tượng  (shuffle) tại màn hình phát nhạc 2. Hệ thống thay đổi thứ tự phát nhạc của danh sách theo ngẫu nhiên
Kịch bản thay thế (Alternative flow)	<ol style="list-style-type: none"> 1. Nếu danh sách đang được phát ngẫu nhiên 2. Chuyển lại phát theo thứ tự của danh sách

4.2.12. Ẩn bài hát

Use Case ID	UC12
Tên	Phát ngẫu nhiên
Mô tả tóm tắt	Cho phép người dùng phát ngẫu nhiên các bài hát trong danh sách
Người thực hiện (Actors)	Người dùng (User)
Sự kiện kích hoạt (Trigger)	Bấm chọn Hide
Điều kiện tiên quyết (Pre-condition)	Người dùng phải đang phát nhạc
Điều kiện kết quả (Post-condition)	Ẩn bài hát đang phát và chuyển tới bài hát kế tiếp






Kịch bản chính (Basic flow)	<ol style="list-style-type: none"> 1. Người dùng chọn  rồi chọn Hide 2. Hệ thống dừng và ẩn bài hát đang phát rồi chuyển tới bài hát kế tiếp trong danh sách
Kịch bản thay thế (Alternative flow)	

4.2.13. Hẹn giờ phát

Use Case ID	UC13
Tên	Hẹn giờ phát
Mô tả tóm tắt	Cho phép người dùng hẹn giờ phát nhạc
Người thực hiện (Actors)	Người dùng (User)
Sự kiện kích hoạt (Trigger)	Bấm chọn Sleep
Điều kiện tiên quyết (Pre-condition)	Người dùng phải đang phát nhạc
Điều kiện kết quả (Post-condition)	Dừng phát nhạc sau thời gian đã đặt
Kịch bản chính (Basic flow)	<ol style="list-style-type: none"> 1. Người dùng chọn  rồi chọn Sleep sau đó chọn thời gian 2. Hệ thống tạm dừng phát nhạc cho tới khi người dùng bật lại
Kịch bản thay thế (Alternative flow)	

4.2.14. Yêu thích bài hát

Use Case ID	UC14
-------------	------


Tên	Yêu thích bài hát
Mô tả tóm tắt	Cho phép người dùng yêu thích bài hát
Người thực hiện (Actors)	Người dùng (User)
Sự kiện kích hoạt (Trigger)	Bấm chọn  ở màn hình phát nhạc
Điều kiện tiên quyết (Pre-condition)	Đang ở trong màn hình phát nhạc
Điều kiện kết quả (Post-condition)	Thêm bài hát vào danh sách yêu thích
Kịch bản chính (Basic flow)	<ol style="list-style-type: none"> 1. Người dùng chọn  ở màn hình phát nhạc 2. Thêm bài hát vào danh sách yêu thích và đổi màu icon thành 
Kịch bản thay thế (Alternative flow)	<ol style="list-style-type: none"> 1. Nếu bài hát đã ở trong danh sách yêu thích 2. Bỏ thích bài hát và chuyển icon từ  thành 

4.2.15. Tìm kiếm bài hát và ca sĩ

Use Case ID	UC15
Tên	Tìm kiếm
Mô tả tóm tắt	Cho phép người dùng tìm kiếm bài hát trên hệ thống
Người thực hiện (Actors)	Người dùng (User)
Sự kiện kích hoạt (Trigger)	Nhập nội dung tìm kiếm ở màn hình Search

Điều kiện tiên quyết (Pre-condition)	Nội dung tìm kiếm có tồn tại trong hệ thống
Điều kiện kết quả (Post-condition)	Danh sách bài hát, ca sĩ có tên gần với từ khóa tìm kiếm
Kịch bản chính (Basic flow)	<ol style="list-style-type: none"> 1. Người dùng nhập từ khóa ở ô search trong màn hình search 2. Hệ thống tìm kiếm và xuất kết quả bài hát, ca sĩ có tên gần với từ khóa tìm kiếm
Kịch bản thay thế (Alternative flow)	

4.2.16. Thay đổi ảnh đại diện

Use Case ID	UC16
Tên	Thay đổi ảnh đại diện
Mô tả tóm tắt	Cho phép người dùng thay đổi ảnh đại diện
Người thực hiện (Actors)	Người dùng (User)
Sự kiện kích hoạt (Trigger)	Bấm chọn  ở màn hình Other
Điều kiện tiên quyết (Pre-condition)	Người dùng đã đăng nhập
Điều kiện kết quả (Post-condition)	Thay đổi ảnh đại diện

Kịch bản chính (Basic flow)	<ol style="list-style-type: none"> 1. Người dùng chọn icon ở màn hình Other 2. Người dùng chọn ảnh để thay đổi 3. Hệ thống hiện thông báo hỏi người dùng có xác nhận thay đổi ảnh đại diện không 4. Người dùng chọn Yes 5. Hệ thống lưu ảnh cho tài khoản hiện tại và thay đổi ảnh đại diện của người dùng
Kịch bản thay thế (Alternative flow)	4.1 Người dùng chọn No <ol style="list-style-type: none"> 1. Không thay đổi

4.2.17. Thay đổi thông tin người dùng

Use Case ID	UC17
Tên	Thay đổi thông tin
Mô tả tóm tắt	Cho phép người dùng thay đổi thông tin cá nhân
Người thực hiện (Actors)	Người dùng (User)
Sự kiện kích hoạt (Trigger)	Bấm chọn nút Edit Profile
Điều kiện tiên quyết (Pre-condition)	Người dùng đã đăng nhập vào ứng dụng
Điều kiện kết quả (Post-condition)	Thay đổi thông tin của người dùng
Kịch bản chính (Basic flow)	<ol style="list-style-type: none"> 1. Người dùng bấm chọn nút EditProfile 2. Hệ thống hiện màn hình thông tin user 3. Thay đổi các thông tin cần thay đổi 4. Hệ thống kiểm tra tính hợp lệ của các thông tin đã nhập và bấm Save 5. Hệ thống lưu lại thông tin người dùng và chuyển về màn hình Other

Kịch bản thay thế (Alternative flow)	4a.1 Các thông tin nhập chưa hợp lệ 4a.2 Hệ thống thông báo thông tin chưa hợp lệ và yêu cầu nhập lại 4b.1 Người dùng chọn cancel 4b.2 Quay lại màn hình Other
---	---

Chương 5: THIẾT KẾ DỮ LIỆU

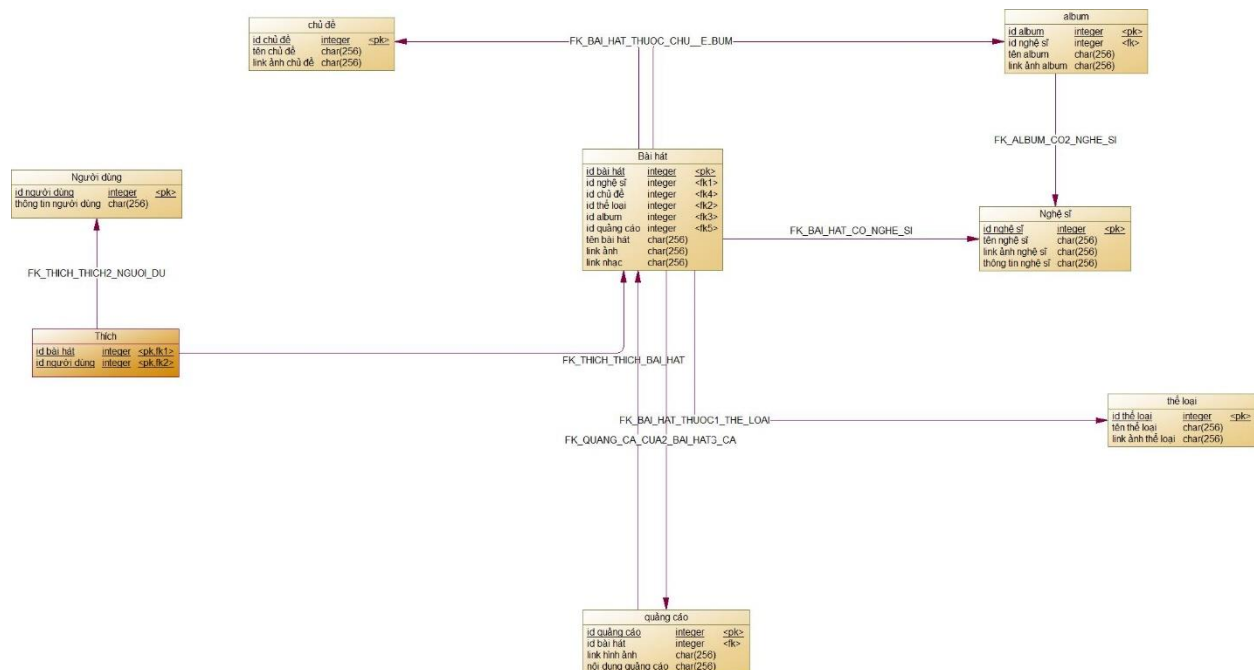
5.1. Khảo sát hiện trạng

- Ứng dụng nghe nhạc có bài hát. Mỗi bài hát có mỗi mã số và tên gọi để phân biệt
- Mỗi bài hát mô tả chi tiết tên bài hát, tên ca sĩ, thể loại, và link bài hát
- Ứng dụng cũng có quản lý các tài khoản riêng mỗi tài khoản bao gồm thông tin cá nhân như hình ảnh, tên, tuổi...
- Mỗi ứng dụng cũng có chức năng và quyền hạn riêng để tương tác với ứng dụng

5.2. Mô tả tân từ

- Mỗi tài khoản có một mã duy nhất để phân biệt. Mỗi mã tài khoản xác định tất cả các thuộc tính còn lại của tài khoản đó, bao gồm: Tên đăng nhập, mật khẩu, tên hiển thị và loại tài khoản. Mỗi tài khoản liên kết với một khách hàng
- Mỗi khách hàng có một mã khách hàng để phân biệt. Mỗi mã khách hàng xác định tất cả các thuộc tính còn lại của khách hàng đó, bao gồm: Tên khách hàng, số điện thoại, email, giới tính, ngày sinh, ảnh đại diện
- Mỗi thể loại bài hát đều có mã số để phân biệt. Mỗi mã thể loại bài hát cho ta biết được bài hát đó thuộc loại gì. Một bài hát có thể thuộc nhiều thể loại khác nhau.
- Một ca sĩ có một ID duy nhất để phân biệt. Mỗi ID ca sĩ xác định tất cả các thuộc tính còn lại của ca sĩ đó, bao gồm: Thông tin ca sĩ, tên ca sĩ, ảnh đại diện... Một ca sĩ có thể chứa một hay nhiều bài hát
- Một bài hát có một ID bài hát duy nhất để phân biệt. Mỗi ID bài hát xác định tất cả các thuộc tính còn lại của bài hát đó, bao gồm: Tên bài hát, link ảnh, link nhạc, ID nghệ sĩ, ID album, ID chủ đề, ID quảng cáo. Một sản phẩm có thể thuộc một hay nhiều hóa đơn tùy theo yêu cầu của người mua hàng.
- Một quảng cáo có một ID quảng cáo duy nhất để phân biệt. Mỗi ID quảng cáo xác định các thuộc tính còn lại của quảng cáo đó, bao gồm: Tên quảng cáo, ID bài hát, nội dung quảng cáo. Một quảng cáo chỉ có quảng cáo một bài hát duy nhất
- Mỗi người dùng có một tài khoản đăng nhập duy nhất để phân biệt. Mỗi tài khoản đăng nhập xác định tất cả các thuộc tính còn lại của người dùng đó, bao gồm: mật khẩu đăng nhập và quyền truy cập hệ thống

5.3. Sơ đồ logic



5.4. Mô tả chi tiết các kiểu dữ liệu

5.4.1. Bảng bài hát

STT	Tên thuộc tính	Kiểu	Ràng buộc	Ý nghĩa/ Ghi chú
1	IDBaiHat	Integer	>0	Khóa chính
2	IDNgheSI	Integer	>0	Id nghệ sĩ
3	IDChuDe	Integer	>0	Id chủ đề
4	IDTheLoai	Integer	>0	Id thể loại
5	IDQuangCao	Integer	>0	Id quảng cáo
6	IDAlbum	Integer	>0	Id Album
7	TenBaiHat	Char	Tối đa 256 kí tự	Tên bài hát
8	LinkAnh	Char	Tối đa 256 kí tự	Link ảnh bài hát
9	LinkNhac	Char	Tối đa 256 kí tự	Link nhạc

5.4.2 Bảng nghệ sĩ

STT	Tên thuộc tính	Kiểu	Ràng buộc	Ý nghĩa/Ghi chú
1	IDNgheSi	Integer	>0	Khóa chính

2	TenNgheSi	Char	Tối đa 256 kí tự	Tên nghệ sĩ
3	LinkAnh	Char	Tối đa 256 kí tự	Link ảnh nghệ sĩ
4	TTNgheSi	Char	Tối đa 256 kí tự	Thông tin nghệ sĩ

5.4.3 Bảng Album

STT	Tên thuộc tính	Kiểu	Ràng buộc	Ý nghĩa/Ghi chú
1	IDAlbum	Integer	>0	Khóa chính
2	IDNgheSi	Integer	>0	Id nghệ sĩ
3	TenAlbum	Char	Tối đa 256 kí tự	Tên album
4	LinkAnh	Char	Tối đa 256 kí tự	Link ảnh album

5.4.4 Bảng chủ đề

STT	Tên thuộc tính	Kiểu	Ràng buộc	Ý nghĩa/Ghi chú
1	IDChuDe	Integer	>0	Khóa chính
2	TenChuDe	Char	Tối đa 256 kí tự	Tên chủ đề
3	LinkAnh	Char	Tối đa 256 kí tự	Link ảnh chủ đề

5.4.5 Bảng thể loại

STT	Tên thuộc tính	Kiểu	Ràng buộc	Ý nghĩa/Ghi chú
1	IDTheLoai	Integer	>0	Khóa chính
2	TenTheLoai	Char	Tối đa 256 kí tự	Tên thể loại
3	LinkAnh	Char	Tối đa 256 kí tự	Link ảnh thể loại

5.4.6 Bảng thích

STT	Tên thuộc tính	Kiểu	Ràng buộc	Ý nghĩa/Ghi chú
1	IDBaiHat	Integer	>0	Khóa chính
2	IDNguoiDung	Char	Tối đa 256 kí tự	IdNguoiDung

5.4.7 Bảng quảng cáo

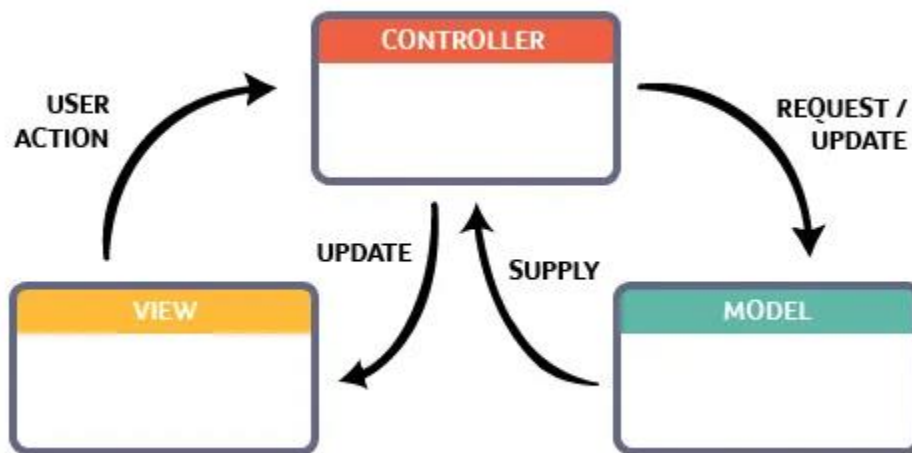
STT	Tên thuộc tính	Kiểu	Ràng buộc	Ý nghĩa/Ghi chú
1	IDQuangCao	Integer	>0	Khóa chính
2	IDBaiHat	Integer	>0	Id bài hát
3	LinkHinhAnh	Char	Tối đa 256 kí tự	Link hình ảnh quảng cáo
4	NoiDungQC	Char	Tối đa 256 kí tự	Nội dung quảng cáo

Chương 6: THIẾT KẾ KIẾN TRÚC

6.1. Ứng dụng kiến trúc MVC để phát triển phần mềm

MVC là viết tắt của cụm từ “Model-View-Controller”. Đây là mô hình thiết kế sử dụng trong kỹ thuật phần mềm. MVC là một mẫu kiến trúc phần mềm để tạo lập giao diện người dùng trên máy tính. MVC chia thành ba phần được kết nối với nhau như tên gọi: Model (dữ liệu), View (giao diện) và Controller (bộ điều khiển).

Đơn giản hơn, là mô hình này được chia thành 3 phần trong source code. Và mỗi phần đảm nhận vai trò và nhiệm vụ riêng biệt nhau và độc lập.



Mô hình MVC (MVC pattern) thường được dùng để phát triển giao diện người dùng. Nó cung cấp các thành phần cơ bản để thiết kế một chương trình cho máy tính hoặc điện thoại di động, cũng như là các ứng dụng web.

Mô hình MVC gồm 3 loại chính là thành phần bên trong không thể thiếu khi áp dụng mô hình này:

Model: Là bộ phận có chức năng lưu trữ toàn bộ dữ liệu của ứng dụng và là cầu nối giữa 2 thành phần bên dưới là View và Controller. Một model là dữ liệu được sử dụng bởi chương trình. Đây có thể là cơ sở dữ liệu, hoặc file XML bình thường hay một đối tượng đơn giản. Chẳng hạn như biểu tượng hay là một nhân vật trong game.

View: Đây là phần giao diện (theme) dành cho người sử dụng. View là phương tiện hiển thị các đối tượng trong một ứng dụng. Chẳng hạn như hiển thị một cửa sổ, nút hay văn bản trong một cửa sổ khác. Nó bao gồm bất cứ thứ gì mà người dùng có thể nhìn thấy được.

Controller: Là bộ phận có nhiệm vụ xử lý các yêu cầu người dùng đưa đến thông qua View. Một controller bao gồm cả Model lẫn View. Nó nhận input và thực hiện các update tương ứng.

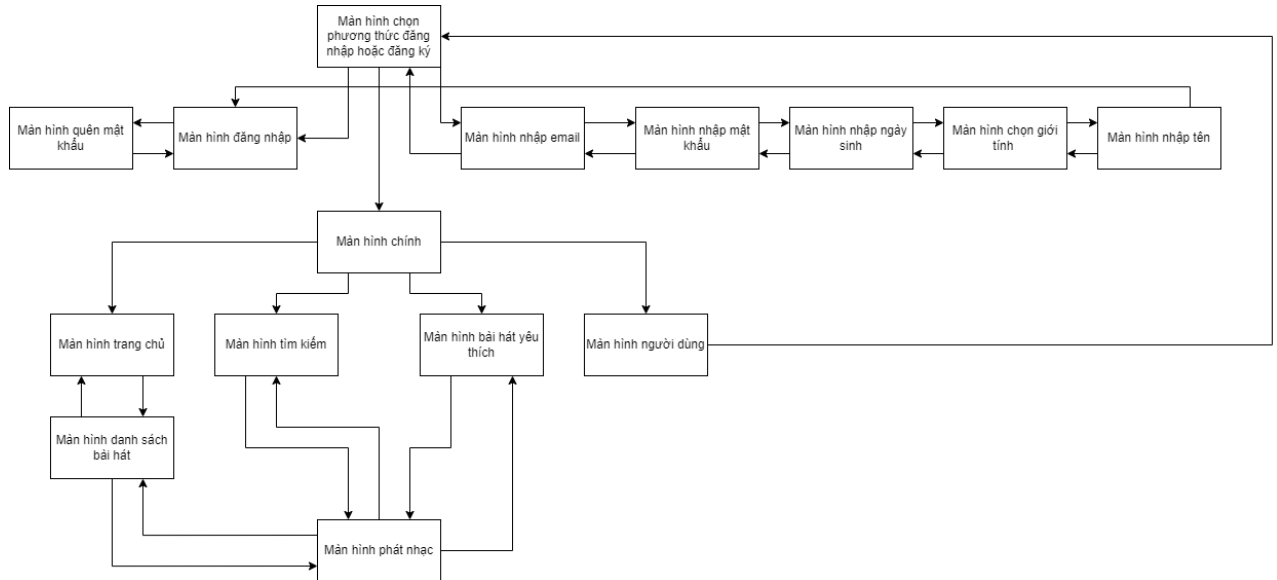
Ví dụ: một Controller có thể cập nhật một Model bằng cách thay đổi các thuộc tính của nhân vật trong game. Và nó có thể sửa đổi view bằng cách hiển thị nhân vật được cập nhật trong game đó.

6.2 Mục đích lựa chọn

- Trình tự xử lý rất rõ ràng
- Mô hình MVC quy hoạch các class/function vào các thành phần riêng biệt Controller - Model - View, việc đó làm cho quá trình phát triển - quản lý - vận hành - bảo trì diễn ra thuận lợi hơn, tạo ra được các chức năng chuyên biệt hoá đồng thời kiểm soát được luồng xử lý.
- Tạo mô hình chuẩn cho dự án, khi người có chuyên môn ngoài dự án tiếp cận với dự án dễ dàng hơn.
- Mô hình đơn giản, dễ hiểu, xử lý những nghiệp vụ đơn giản, và dễ dàng triển khai với các dự án nhỏ.

Chương 7: THIẾT KẾ GIAO DIỆN

7.1. Sơ đồ liên kết các màn hình



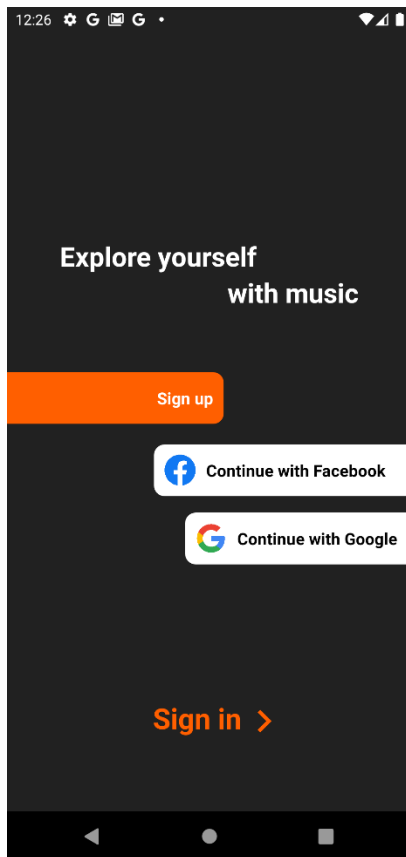
7.2. Danh sách các màn hình

STT	Màn hình	Ý nghĩa/ ghi chú
1	Màn hình chọn phương thức đăng nhập hoặc đăng ký	Chọn các phương thức đăng nhập vào ứng dụng hoặc tạo tài khoản
2	Màn hình đăng nhập	Đăng nhập vào ứng dụng
3	Màn hình quên mật khẩu	Nhập email đã đăng ký để đổi lại mật khẩu mới
4	Màn hình nhập Email	Nhập Email của tài khoản
5	Màn hình nhập mật khẩu	Nhập mật khẩu của tài khoản
6	Màn hình chọn giới tính	Chọn giới tính
7	Màn hình nhập tên	Nhập tên
8	Màn hình trang chủ	Hiển thị quảng cáo bài hát, các tuyển tập bài hát theo chủ đề, thể loại, ca sĩ
9	Màn hình tìm kiếm	Gợi ý 1 số bài hát và tìm kiếm bài hát
10	Màn hình bài hát yêu thích	Hiển thị các bài hát mà người dùng đã thích

- | | | |
|----|----------------------------|--|
| 11 | Màn hình người dùng | Hiển thị và cho phép chỉnh sửa thông tin người dùng, hiển thị thông tin của ứng dụng, đăng xuất tài khoản. |
| 12 | Màn hình danh sách bài hát | Hiển thị danh sách các bài hát theo chủ đề, thể loại, ca sĩ |
| 13 | Màn hình phát nhạc | Hiển thị thông tin bài hát, cho phép người dùng phát nhạc hoặc tạm dừng, chuyển bài hát, thích bài hát, ẩn bài hát, hẹn thời gian dừng |

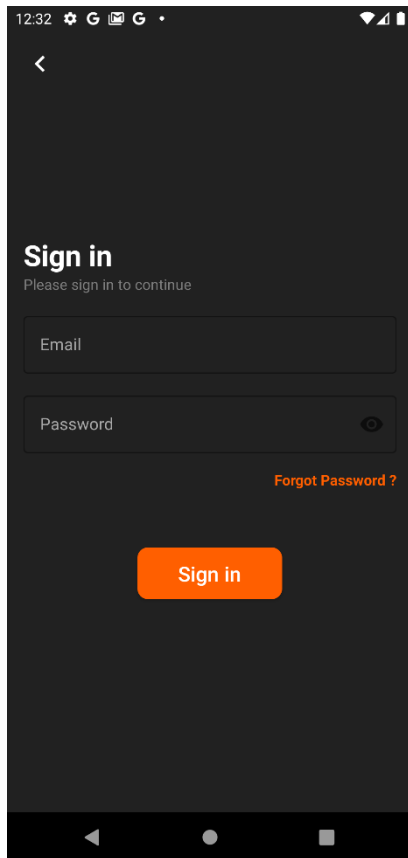
7.3. Mô tả các màn hình

7.3.1. Màn hình chọn phương thức đăng nhập hoặc đăng ký



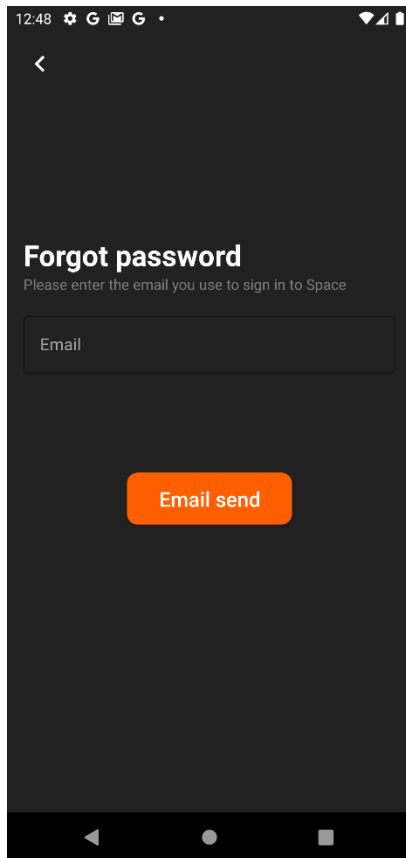
- Bấm nút Sign up để chuyển qua màn hình đăng ký tài khoản
- Bấm nút Facebook để đăng nhập bằng Facebook
- Bấm nút Google để đăng nhập bằng Google
- Bấm Sign in để chuyển qua màn hình đăng nhập

7.3.2. Màn hình đăng nhập



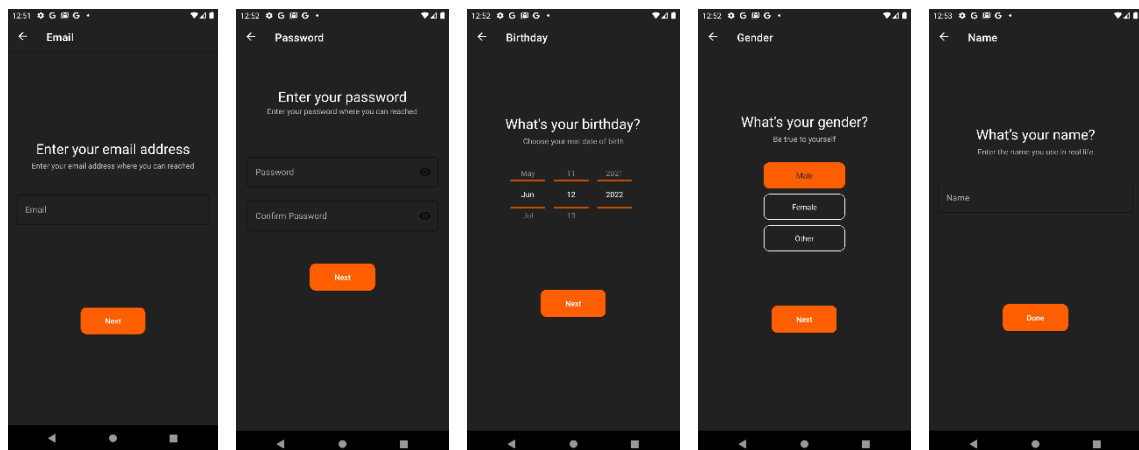
- Điền thông tin Email mà mật khẩu vào TextInput
- Bấm nút Sign in để vào màn hình trang chủ
- Bấm Forgot Password để chuyển qua màn hình quên mật khẩu
- Bấm nút < để trở về màn hình trước đó

7.3.3. Màn hình quên mật khẩu



- Điền thông tin Email vào TextInput
- Bấm nút Email send để hệ thống gửi mail đổi mật khẩu
- Bấm nút < để trở về màn hình trước đó

7.3.4. Các màn hình đăng ký



-Nhập email vào TextInput
-Bấm nút Next để chuyển qua màn hình

- Nhập mật khẩu và xác nhận lại mật khẩu vào TextInput

- Chọn ngày, tháng, năm sinh
- Bấm nút Next để chuyển qua

- Bấm nút Male hoặc Female hoặc Other để chọn giới tính

- Nhập tên vào TextInput
- Bấm nút Done để chuyển qua

nhập mật khẩu

- Bấm nút < để trở về màn hình trước đó

- Bấm nút Next để chuyển sang màn hình chọn ngày sinh

- Bấm nút < để trở về màn hình trước đó

màn hình chọn giới tính

- Bấm nút < để trở về màn hình trước đó

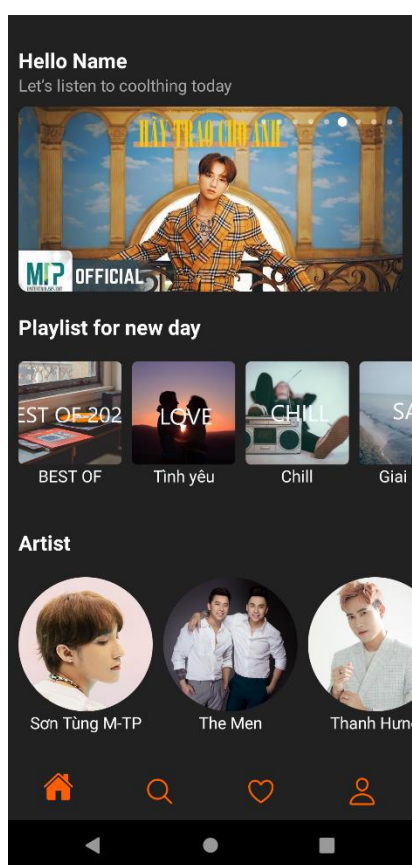
- Bấm nút Next để chuyển qua màn hình nhập tên

- Bấm nút < để trở về màn hình trước đó

màn hình đăng nhập

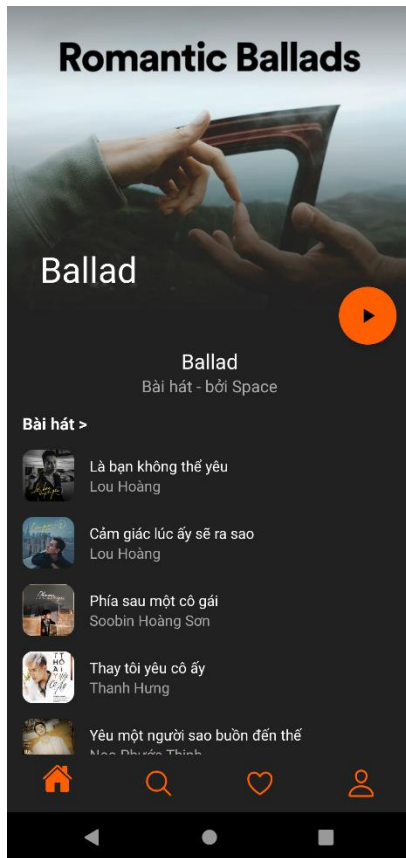
- Bấm nút < để trở về màn hình trước đó

7.3.5. Màn hình trang chủ



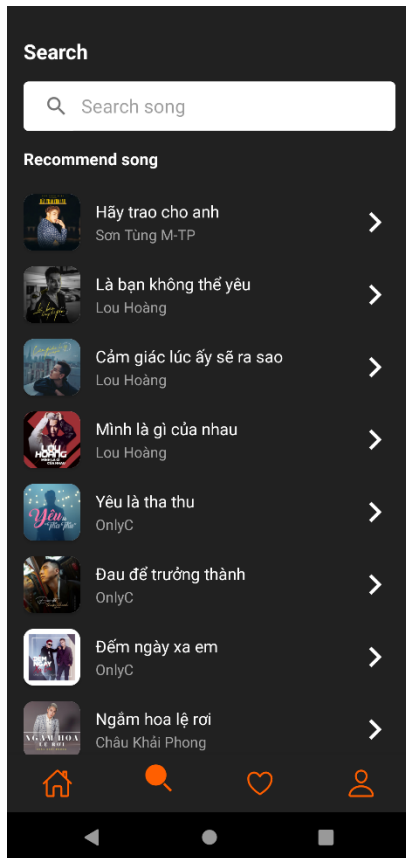
- Bấm vào quảng cáo bài hát để vào màn hình phát nhạc
- Bấm vào các tuyển tập bài hát để chuyển sang màn hình danh sách các bài hát
- Bấm vào icon kính lúp để chuyển sang màn hình tìm kiếm
- Bấm vào icon trái tim để chuyển sang màn hình bài hát yêu thích
- Bấm vào icon người để chuyển sang màn hình người dùng

7.3.6. Màn hình danh sách bài hát



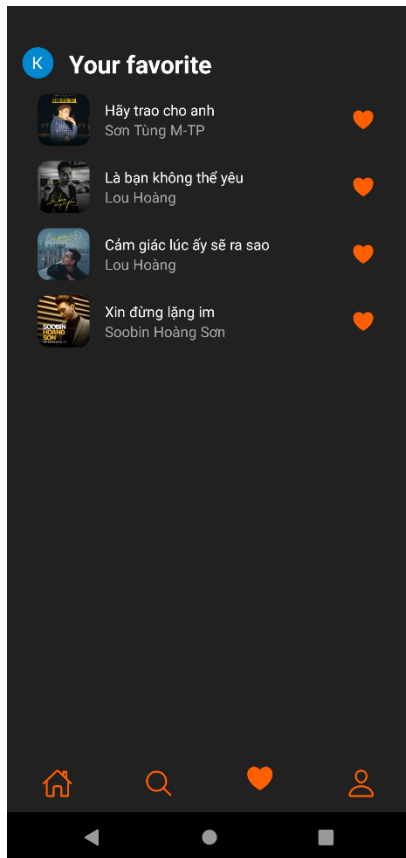
- Bấm nút phát nhạc để phát ngẫu nhiên 1 bài hát trong danh sách bài hát
- Bấm bài hát để chuyển sang màn hình phát nhạc
- Hiện thị các bài hát cùng chủ đề, thể loại hoặc ca sĩ

7.3.7. Màn hình tìm kiếm



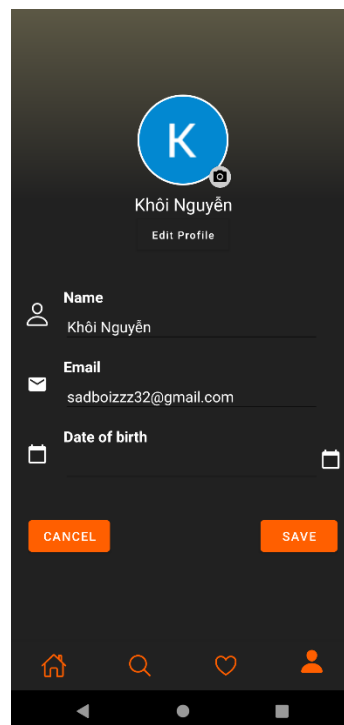
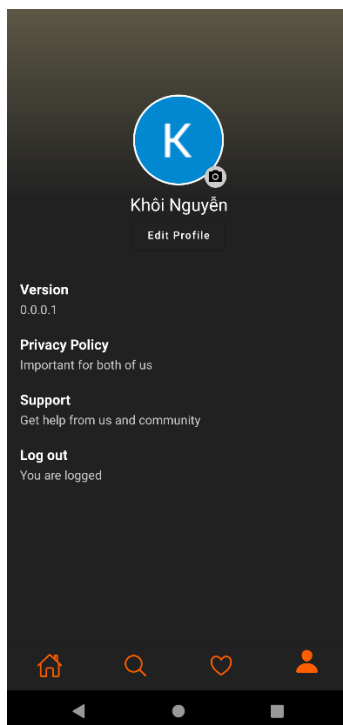
- Nhập tên bài hát hoặc ca sĩ cần tìm kiếm vào SearchView
- Bấm bài hát để chuyển sang màn hình phát nhạc
- Gợi ý một số bài hát và hiển thị bài hát tìm kiếm

7.3.8. Màn hình bài hát yêu thích



- Hiển thị các bài hát mà người dùng đã thích
- Bấm bài hát để chuyển sang màn hình phát nhạc

7.3.9. Màn hình người dùng



- Hiển thị thông tin người dùng và ứng dụng
- Bấm vào nút máy ảnh để cập nhật ảnh đại diện
- Bấm nút Edit Profile để hiện Fragment chỉnh sửa thông tin người dùng
- Bấm nút Cancel để thoát chỉnh sửa
- Bấm nút Save để lưu thông tin

- Bấm nút Privacy Policy để truy cập vào trang web chính sách riêng tư
- Bấm nút Support để gửi mail cho nhà phát triển ứng dụng
- Bấm nút Log out để đăng xuất tài khoản

Chương 8: NHẬN XÉT VÀ KẾT LUẬN

8.1. Ưu điểm đồ án

- Giao diện ưa nhìn
- Dễ dàng sử dụng
- Đáp ứng được nhu cầu nghe nhạc cơ bản
- Dễ dàng đăng nhập cũng như đăng ký.
- Có liên kết với google và facebook
- Vận dụng tương đối đầy đủ các kiến thức đã học.

8.2. Hạn chế của đồ án

- Còn tồn tại một số bug
- Tổ chức source code chưa tốt
- Các chức năng còn cơ bản và chưa hướng thực tế
- Database còn đơn giản chưa chặt chẽ
- Còn ít chức năng
- Chưa có chức năng tải nhạc.
- Chưa có lời bài hát chạy song song với nhạc đang phát

8.3. Hướng phát triển của đồ án

- Thêm chức năng mở lại Activity phát nhạc với trạng thái nhạc cũ.
- Thêm phần thu nhỏ phát nhạc.
- Nâng cấp Server để tải nhạc nhanh hơn
- Thêm chức năng mở nhạc từ máy
- Thêm lời bài hát
- Phân chia các danh sách phát rõ ràng hơn

Tài liệu tham khảo

Tài liệu hướng dẫn sử dụng các component trong android

<https://developer.android.com/>

Tài liệu về SQLite

<https://viblo.asia/p/sqlite-la-gi-E375zVVR5GW>

Các video hướng dẫn làm database ứng dụng nghe nhạc

<https://www.youtube.com/playlist?list=PLzrVYRai0riTMwAVJxP2hk9VTMK4yKEkC>

Các video hướng dẫn làm phát nhạc

https://www.youtube.com/watch?v=bsXaM4G4D_w&list=PLwQLA73lSe1RfjMzbRLolkcIJBu25FnVJ

Tài liệu về MVC:

<https://developer.mozilla.org/en-US/docs/Glossary/MVC>