

ĐẠI HỌC QUỐC GIA THÀNH PHỐ HỒ CHÍ MINH  
TRƯỜNG ĐẠI HỌC BÁCH KHOA  
KHOA KHOA HỌC VÀ KỸ THUẬT MÁY TÍNH



Đồ án tổng hợp - CNPM (CO3103)

---

Báo cáo

# *Ứng dụng nghe nhạc trực tuyến* *MusicHUB*

---

**GVHD:** ThS. Trần Trương Tuấn Phát

**Sinh viên:** Lữ Chấn Vũ - 2313955 (*Nhóm 10, Leader*)  
Nguyễn Phú Vinh - 2313922 (*Nhóm 10*)  
Trần Dương Khiết Nhi - 2312509 (*Nhóm 10*)  
Lê Minh Khoa - 2311593 (*Nhóm 10*)  
Lê Minh Trí - 2313593 (*Nhóm 10*)

TP. HỒ CHÍ MINH, 09/2025



## Mục lục

Danh sách kí hiệu	2
Danh sách từ viết tắt	2
Danh sách hình ảnh	4
Danh sách bảng biểu	4
Danh sách thành viên và nhiệm vụ	4
<b>1 Tổng quan về đề tài</b>	<b>5</b>
1.1 Mô tả đề tài	5
1.2 Mục tiêu đề tài	5
1.3 Phạm vi đề tài	5
<b>2 Functional Requirements - Non-functional Requirements</b>	<b>5</b>
2.1 Functional Requirements	5
2.1.1 Non-interactive Requirements	5
2.1.1.a Tính năng Streaming ở nhiều mức bitrate	5
2.1.1.b Tính năng gợi ý bài hát dựa vào lượt nghe gần đây	6
2.1.1.c Tính năng tự động tạo ảnh cover/thumbnail nếu Người dùng không upload	6
2.1.1.d Tính năng tự động phát bài hát tiếp theo trong playlist/queue	7
2.1.1.e Tính năng thông báo khi Nghệ sĩ được follow phát hành bài hát/album mới	8
2.1.1.f Tính năng thống kê thời gian nghe và lượt nghe	8
2.1.1.g Tính năng tạo và đồng bộ lời bài hát (Lyric Sync)	9
2.2 Non-functional Requirements	10
2.2.1 Hiệu năng (Performance)	10
2.2.2 Bảo mật (Security)	10
2.2.3 Độ tin cậy (Reliability)	10
2.2.4 Khả năng sử dụng (Usability)	11
2.2.5 Khả năng mở rộng (Scalability)	11
2.2.6 Tính sẵn sàng (Availability)	11
2.2.7 Tính tương thích (Compatibility)	11
2.2.8 Tuân thủ pháp lý (Compliance)	12
2.2.9 Khả năng kiểm toán (Audit)	12
2.2.10 Hiệu quả sử dụng tài nguyên (Efficiency)	12
2.2.11 Khả năng bảo trì (Maintainability)	12
<b>3 Use-case Diagram và Use-case Scenario</b>	<b>13</b>
3.1 Use-case Diagram	13
3.2 Use-case Scenario	13
<b>4 Thiết kế hành vi và giao diện người dùng</b>	<b>13</b>
4.1 Đăng nhập - đăng kí	13
4.1.1 Đăng nhập - Login	13
4.1.2 Đăng ký - Register	16



## Danh sách kí hiệu

N Tập hợp số tự nhiên

## Danh sách từ viết tắt

**CSP** Cutting Stock Problem

**FFD** First Fit Decreasing

**GA** Genetic Algorithm

**LP** Linear Programming



## Danh sách hình ảnh

1	Use-case Diagram . . . . .	13
2	Sequence Diagram - Login . . . . .	21
3	Activity Diagram - Login . . . . .	22
4	Sequence Diagram - Register . . . . .	23
5	Activity Diagram - Register . . . . .	24

## Danh sách bảng biểu

1	Danh sách thành viên và nhiệm vụ . . . . .	4
2	Mô tả usecase Upload bài hát/album . . . . .	14
3	Mô tả usecase Đăng nhập & Đăng ký . . . . .	15
4	Mô tả usecase Quản lý tài khoản . . . . .	16
5	Mô tả usecase Tương tác . . . . .	17
6	Mô tả usecase Nghe nhạc . . . . .	18
7	Mô tả usecase Khám phá và tìm kiếm . . . . .	19
8	Mô tả usecase Nhận diện bài hát . . . . .	20



## Danh sách thành viên và nhiệm vụ

STT	Họ và tên	MSSV	Nhiệm vụ	% hoàn thành
1	Lư Chấn Vũ	2313955	-	100%
2	Nguyễn Phú Vinh	2313922	-	100%
3	Trần Dương Khiết Nhi	2312509	-	100%
4	Lê Minh Khoa	2311593	-	100%
5	Lê Minh Trí	2313593	-	100%

Bảng 1: Danh sách thành viên và nhiệm vụ

# 1 Tổng quan về đề tài

## 1.1 Mô tả đề tài

## 1.2 Mục tiêu đề tài

## 1.3 Phạm vi đề tài

# 2 Functional Requirements - Non-functional Requirements

## 2.1 Functional Requirements

### 2.1.1 Non-interactive Requirements

#### 2.1.1.a Tính năng Streaming ở nhiều mức bitrate

Chức năng này được thiết kế nhằm tối ưu trải nghiệm nghe nhạc của Người dùng trên nhiều loại thiết bị và trong các điều kiện mạng khác nhau. Sau khi một bài hát được upload thành công lên hệ thống, máy chủ sẽ tự động tiến hành xử lý và chuyển đổi file gốc sang nhiều phiên bản với các mức bitrate khác nhau (ví dụ: 64kbps, 128kbps, 320kbps).

- **Cách hoạt động:**

- Khi Người dùng phát một bài hát, hệ thống sẽ cung cấp nhiều lựa chọn bitrate khác nhau (64kbps, 128kbps, 320kbps).
  - \* 64kbps: Phù hợp cho kết nối mạng yếu, tiết kiệm dữ liệu.
  - \* 128kbps: Chất lượng tiêu chuẩn, cân bằng giữa tốc độ tải và chất lượng âm thanh.
  - \* 320kbps: Chất lượng cao, dành cho Người dùng muốn trải nghiệm âm nhạc tốt nhất.
- Người dùng có thể chọn thủ công mức bitrate mong muốn, phù hợp với chất lượng mạng và nhu cầu nghe nhạc.
- Hệ thống hỗ trợ **Adaptive Streaming**: trong quá trình nghe, nếu mạng yếu hoặc không ổn định, bitrate sẽ tự động hạ xuống để tránh gián đoạn; khi mạng mạnh trở lại, bitrate được nâng lên để đảm bảo chất lượng âm thanh tốt nhất.
- Các phiên bản nhạc ở nhiều bitrate được tạo sẵn trong quá trình xử lý upload, do đó việc chuyển đổi diễn ra nhanh chóng và mượt mà.

- **Input:**

- File nhạc gốc (định dạng hợp lệ: MP3, WAV, FLAC,...).
- Metadata bài hát (tên, nghệ sĩ, thể loại, ảnh bìa).
- Thông tin cấu hình hệ thống (các mức bitrate cần tạo).

- **Output:**

- Các phiên bản bài hát ở nhiều mức bitrate (64kbps, 128kbps, 320kbps).
- Đường dẫn hoặc ID truy cập các file đã xử lý để phát trực tuyến.

- **Lợi ích:**

- Người dùng có trải nghiệm nghe nhạc mượt mà, ngay cả khi mạng yếu.
- Tối ưu dung lượng lưu trữ và băng thông cho hệ thống.
- Đáp ứng nhu cầu đa dạng: nghe nhạc tiết kiệm dữ liệu hoặc chất lượng cao.

#### 2.1.1.b Tính năng gợi ý bài hát dựa vào lượt nghe gần đây

Chức năng này giúp cá nhân hoá trải nghiệm nghe nhạc của Người dùng. Hệ thống sẽ phân tích lịch sử nghe nhạc gần đây (bài hát, nghệ sĩ, thể loại) để tự động đưa ra danh sách gợi ý phù hợp với sở thích hiện tại của Người dùng. Danh sách gợi ý có thể hiển thị dưới dạng playlist hoặc phần “Đề xuất cho bạn” trên giao diện chính.

- **Cách hoạt động:**

- Hệ thống lưu trữ và theo dõi lịch sử nghe nhạc của Người dùng.
- Dựa trên dữ liệu lượt nghe gần đây, hệ thống áp dụng thuật toán gợi ý (lọc cộng tác, phân tích nội dung hoặc kết hợp).
- Trả về danh sách bài hát, album hoặc Nghệ sĩ có mức độ tương đồng cao.

- **Input:**

- Lịch sử nghe nhạc gần đây (danh sách bài hát đã phát).
- Metadata của bài hát (thể loại, nghệ sĩ, album, nhãn).
- Dữ liệu hành vi Người dùng khác (để tăng độ chính xác).

- **Output:**

- Danh sách gợi ý gồm các bài hát, playlist hoặc Nghệ sĩ liên quan.
- Giao diện hiển thị mục “Gợi ý cho bạn” được cập nhật động.

- **Lợi ích:**

- Giúp Người dùng khám phá thêm nhạc mới phù hợp với sở thích.
- Tăng mức độ gắn bó và thời gian sử dụng ứng dụng.
- Nâng cao trải nghiệm nhờ cá nhân hoá thông minh.

#### 2.1.1.c Tính năng tự động tạo ảnh cover/thumbnail nếu Người dùng không upload

Khi Người dùng upload bài hát nhưng không cung cấp ảnh bìa (cover) hoặc thumbnail, hệ thống sẽ tự động sinh ra hình ảnh thay thế. Hình ảnh được tạo có thể dựa trên thông tin metadata của bài hát (tên, nghệ sĩ, thể loại) hoặc sử dụng mẫu có sẵn. Mục tiêu là đảm bảo tất cả các bài hát trong hệ thống đều có ảnh hiển thị nhất quán và trực quan.

- **Cách hoạt động:**

- Hệ thống kiểm tra file ảnh cover kèm theo khi upload bài hát.
- Nếu không có ảnh, hệ thống kích hoạt module sinh ảnh tự động.
- Ảnh được tạo bằng cách:
  - \* Sinh ngẫu nhiên từ template mặc định theo thể loại nhạc.
  - \* Kết hợp text (tên bài hát, nghệ sĩ) với nền gradient hoặc ảnh mẫu.
- Ảnh được gán cho bài hát và hiển thị trong thư viện, playlist, cũng như trình phát nhạc.

- **Input:**

- File nhạc gốc upload lên hệ thống.
- Metadata bài hát (tên, nghệ sĩ, thể loại).
- Bộ template mặc định của hệ thống.

- **Output:**

- Ảnh cover/thumbnail tự động sinh ra cho bài hát.
- Đường dẫn hoặc ID của ảnh lưu trữ trên server.

- **Lợi ích:**

- Đảm bảo giao diện ứng dụng đồng bộ, không có bài hát bị thiếu ảnh hiển thị.
- Giúp Người dùng tiết kiệm thời gian chuẩn bị file ảnh trước khi upload.
- Nâng cao trải nghiệm nghe nhạc với hình ảnh trực quan và thẩm mỹ.

#### 2.1.1.d Tính năng tự động phát bài hát tiếp theo trong playlist/queue

Khi một bài hát trong playlist hoặc queue kết thúc, hệ thống sẽ tự động phát bài hát kế tiếp mà không cần thao tác thủ công từ Người dùng. Điều này giúp trải nghiệm nghe nhạc liền mạch và thuận tiện, đặc biệt khi Người dùng đang nghe nhạc trong lúc làm việc, tập luyện hoặc thư giãn.

- **Cách hoạt động:**

- Khi bài hát hiện tại kết thúc, hệ thống kiểm tra danh sách playlist/queue đang hoạt động.
- Nếu còn bài hát trong danh sách, hệ thống sẽ tự động phát bài kế tiếp theo thứ tự.
- Nếu đến cuối danh sách: Tùy chế độ, có thể dừng phát nhạc, lặp lại playlist, hoặc bật chế độ phát ngẫu nhiên.
- Quá trình chuyển bài diễn ra mượt mà, không tạo khoảng trống âm thanh lớn.

- **Input:**

- Danh sách playlist hoặc queue do Người dùng tạo hoặc hệ thống gợi ý.
- Thiết lập chế độ phát nhạc (bình thường, lặp lại, ngẫu nhiên).

- **Output:**

- Bài hát kế tiếp được phát tự động ngay sau khi bài hiện tại kết thúc.
- Trạng thái phát nhạc được cập nhật trong trình phát và UI của ứng dụng.

- **Lợi ích:**

- Mang lại trải nghiệm nghe nhạc liền mạch, không bị gián đoạn.
- Người dùng không cần thao tác thủ công, thuận tiện trong nhiều ngữ cảnh.
- Hỗ trợ các chế độ phát linh hoạt, phù hợp với nhu cầu nghe nhạc đa dạng.



#### 2.1.1.e Tính năng thông báo khi Nghệ sĩ được follow phát hành bài hát/album mới

Khi Người dùng follow một Nghệ sĩ, hệ thống sẽ theo dõi các hoạt động phát hành của Nghệ sĩ đó. Khi có bài hát hoặc album mới được phát hành, hệ thống sẽ gửi thông báo đến Người dùng để họ có thể thưởng thức ngay lập tức. Điều này giúp tăng mức độ gắn kết giữa Người dùng và Nghệ sĩ, đồng thời nâng cao trải nghiệm khám phá nhạc mới.

- **Cách hoạt động:**

- Hệ thống lưu danh sách Nghệ sĩ mà Người dùng đã follow.
- Khi Nghệ sĩ phát hành bài hát/album mới, hệ thống kiểm tra và xác định những Người dùng đã follow.
- Gửi thông báo qua ứng dụng (push notification).
- Thông báo chứa thông tin cơ bản như tên bài hát/album, ngày phát hành, và liên kết để nghe trực tiếp.

- **Input:**

- Danh sách Nghệ sĩ được Người dùng follow.
- Dữ liệu phát hành bài hát/album mới của Nghệ sĩ.

- **Output:**

- Thông báo gửi đến Người dùng khi có bài hát/album mới.
- Liên kết dẫn trực tiếp đến nội dung nhạc trong ứng dụng.

- **Lợi ích:**

- Người dùng luôn cập nhật kịp thời nhạc mới từ Nghệ sĩ yêu thích.
- Tăng mức độ tương tác và gắn bó giữa Người dùng và nền tảng.
- Giúp Nghệ sĩ tiếp cận nhanh chóng đến fan hâm mộ của mình.

#### 2.1.1.f Tính năng thống kê thời gian nghe và lượt nghe

Hệ thống cung cấp cho Người dùng thống kê chi tiết về hoạt động nghe nhạc của họ, bao gồm tổng thời gian nghe, số lượt nghe theo ngày/tuần/tháng, các bài hát được nghe nhiều nhất, và Nghệ sĩ được yêu thích nhất. Mục tiêu là giúp Người dùng theo dõi thói quen nghe nhạc, đồng thời tăng mức độ gắn bó với ứng dụng thông qua các báo cáo cá nhân hóa.

- **Cách hoạt động:**

- Mỗi lần Người dùng phát một bài hát, hệ thống ghi nhận thời lượng nghe và số lượt nghe.
- Dữ liệu được lưu trữ và cập nhật liên tục trong cơ sở dữ liệu.
- Người dùng có thể xem thống kê dưới dạng biểu đồ, danh sách hoặc báo cáo theo mốc thời gian (ngày/tuần/tháng/năm).
- Hệ thống có thể tạo báo cáo đặc biệt (ví dụ: tổng kết cuối năm) để tăng trải nghiệm Người dùng.

- **Input:**

- Hoạt động nghe nhạc của Người dùng (thời gian phát, bài hát, Nghệ sĩ, playlist).
- Thông tin Người dùng để liên kết dữ liệu thống kê.

- **Output:**

- Báo cáo thống kê: tổng thời gian nghe, số lượt nghe, top bài hát/Nghệ sĩ/album.
- Biểu đồ hoặc bảng dữ liệu trực quan hiển thị trong ứng dụng.

- **Lợi ích:**

- Người dùng có thể theo dõi thói quen nghe nhạc của bản thân.
- Tăng sự gắn bó với ứng dụng thông qua báo cáo cá nhân hóa.
- Tạo cơ sở dữ liệu cho hệ thống gợi ý nhạc chính xác hơn.
- Có thể sử dụng để tổ chức sự kiện đặc biệt (ví dụ: “Top bài hát năm của bạn”).

### 2.1.1.g Tính năng tạo và đồng bộ lời bài hát (Lyric Sync)

Hệ thống cho phép Người dùng trải nghiệm nghe nhạc với phần hiển thị lời bài hát được đồng bộ theo thời gian phát nhạc (karaoke-style). Khi upload bài hát, Nghệ sĩ hoặc quản trị viên có thể thêm file lyric kèm theo, hoặc hệ thống hỗ trợ nhập thủ công và đồng bộ từng câu hát với thời gian. Mục tiêu là mang đến trải nghiệm nghe nhạc sinh động, giúp Người dùng dễ dàng theo dõi và hát theo bài hát.

- **Cách hoạt động:**

- Khi upload, Người dùng hoặc Nghệ sĩ cung cấp file lời bài hát (.lrc hoặc định dạng hỗ trợ).
- Nếu không có file sẵn, hệ thống cho phép nhập lời và sử dụng công cụ đồng bộ thời gian (timestamp editor).
- Khi phát nhạc, trình phát hiển thị lời bài hát, cuộn và highlight theo đúng nhịp bài hát.

- **Input:**

- File nhạc upload lên hệ thống.
- File lyric (.lrc) hoặc text lyric do Người dùng/Nghệ sĩ nhập vào.
- Timestamp đồng bộ (có thể nhập thủ công hoặc tự động gợi ý).

- **Output:**

- Lời bài hát hiển thị trực quan và đồng bộ theo thời gian phát nhạc.
- Highlight từng dòng lyric theo nhạc.

- **Lợi ích:**

- Nâng cao trải nghiệm nghe nhạc, đặc biệt với Người dùng muốn hát theo.
- Tăng tính chuyên nghiệp, tạo cảm giác gần gũi như ứng dụng karaoke.

## 2.2 Non-functional Requirements

### 2.2.1 Hiệu năng (Performance)

Hệ thống phải có khả năng phản hồi nhanh chóng trong mọi tình huống sử dụng phổ biến.

- Thời gian phản hồi cho các thao tác chính (như đăng nhập, tìm kiếm bài hát, phát nhạc, chuyển bài, tạo playlist,...) **không được vượt quá 2 giây** (đối với tối thiểu 95% yêu cầu), ngay cả khi có **1000 người dùng đồng thời** truy cập.
- Thời gian bắt đầu phát nhạc (từ khi ấn nút Play đến khi nghe)  $\leq 800$  ms trong điều kiện mạng ổn định.
- Tác vụ nhận diện bài hát từ đoạn ghi âm 6-10s phải trả kết quả trong vòng  $\leq 2$  giây để đảm bảo trải nghiệm mượt mà.

### 2.2.2 Bảo mật (Security)

Hệ thống cần đảm bảo mức độ bảo mật cao.

- Người dùng phải đăng nhập qua tài khoản riêng (email/social login OAuth 2.0) để đảm bảo xác thực an toàn.
- Dữ liệu cá nhân (tên, email, playlist, lịch sử nghe) và dữ liệu nhạc phải được mã hóa khi lưu trữ (AES-256) và truyền tải (TLS 1.2+).
- Hệ thống áp dụng **role-based access control** (phân quyền): người dùng thường không thể truy cập chức năng quản trị.
- Phiên đăng nhập tự động hết hạn sau **30 phút không hoạt động**.
- Hệ thống có cơ chế giới hạn tốc độ (rate limit) để ngăn chặn brute force và tấn công DDoS:
  - Tối đa 5 lần đăng nhập thất bại trong vòng 15 phút sẽ khóa tài khoản trong 30 phút.
  - Giới hạn tối đa 100 yêu cầu API mỗi phút cho mỗi địa chỉ IP.
  - Mỗi tài khoản người dùng chỉ được phép phát tối đa 3 luồng nhạc đồng thời.
  - Giới hạn tối đa 10 yêu cầu nhận diện bài hát mỗi phút cho mỗi tài khoản người dùng.

### 2.2.3 Độ tin cậy (Reliability)

Hệ thống phải có khả năng hoạt động liên tục, ổn định và không bị gián đoạn.

- Hệ thống cần đảm bảo **độ sẵn sàng (uptime)** tối thiểu là **99.9%** cho dịch vụ phát nhạc, và tối thiểu **99.5%** cho dịch vụ nhận diện bài hát.
- Cơ chế sao lưu dữ liệu được thực hiện tự động hàng ngày để tránh mất mát thông tin trong trường hợp xảy ra sự cố hệ thống.
- Có thông báo lỗi rõ ràng nếu hệ thống không khả dụng, kèm theo phương án xử lý.
- Khi gặp sự cố, phải có cơ chế **khôi phục dữ liệu trong vòng 24 giờ**.

#### 2.2.4 Khả năng sử dụng (Usability)

- Giao diện người dùng (UI) trực quan, thân thiện, quen thuộc với các biểu tượng phổ biến (Play/Pause, Next, Like, Playlist,...).
- Hỗ trợ **song ngữ (tiếng Việt và tiếng Anh)** để phù hợp với nhiều đối tượng người dùng.
- Người dùng mới có thể làm quen và thành thạo với các chức năng cơ bản (tìm kiếm, phát nhạc, playlist, nhận diện bài hát,...) sau  **$\leq 15$  phút** hướng dẫn.
- Hỗ trợ **đa nền tảng**: trình duyệt web phổ biến (Chrome, Firefox, Edge, Safari – 2 phiên bản gần nhất) và ứng dụng di động (Android 10+, iOS 14+).
- Hỗ trợ **accessibility** (screen reader, contrast mode, điều khiển bàn phím,...) để đảm bảo các đối tượng khuyết tật có thể sử dụng.

#### 2.2.5 Khả năng mở rộng (Scalability)

- Hệ thống có khả năng phục vụ **thêm ít nhất 3000 người dùng/năm** mà không ảnh hưởng hiệu năng.
- Có thể mở rộng, tích hợp thêm các module mới như: gợi ý nhạc bằng AI, podcast,... mà không làm thay đổi cấu trúc hiện tại.
- Các microservice (nghe nhạc, tìm kiếm, nhận diện, gợi ý) có thể scale độc lập trên cloud.

#### 2.2.6 Tính sẵn sàng (Availability)

- Hệ thống cần hoạt động liên tục và vận hành ổn định **24/7** để hỗ trợ người dùng mọi lúc.
- Thời gian bảo trì định kỳ **không quá 4 lần/năm**, mỗi lần  $\leq 4$  giờ.
- Nếu bảo trì, cần có thông báo trên hệ thống trước ít nhất **24 giờ**.
- Trong thời gian bảo trì, hệ thống phải hiển thị thông báo rõ ràng cho người dùng.

#### 2.2.7 Tính tương thích (Compatibility)

- Ứng dụng tương thích với nhiều thiết bị (điện thoại, máy tính, máy tính bảng) và nhiều chuẩn phát nhạc (HLS/DASH, AAC/Opus).
- Hệ thống hoạt động tốt trên nhiều nền tảng thiết bị khác nhau:
  - Mobile app tương thích với hệ điều hành **Android 10 trở lên** và **iOS 14 trở lên**.
  - Web app tương thích với các trình duyệt hiện đại như **Google Chrome, Mozilla Firefox, Microsoft Edge, Apple Safari** (đối với 2 phiên bản gần nhất).
- Hệ thống tích hợp API với dịch vụ bên ngoài: social login (Google, Facebook), thanh toán (nếu có Premium), API âm nhạc (metadata, lyric).

### 2.2.8 Tuân thủ pháp lý (Compliance)

Hệ thống cần tuân thủ các quy định về bản quyền âm nhạc, bảo mật dữ liệu và quyền riêng tư.

- Hệ thống phải tuân thủ **Luật An ninh mạng Việt Nam** liên quan đến thu thập, xử lý, lưu trữ và bảo vệ thông tin cá nhân trong lãnh thổ Việt Nam.
- Nếu mở rộng quốc tế, phải đáp ứng chuẩn **GDPR (General Data Protection Regulation)** về bảo mật dữ liệu và quyền riêng tư.
- Đảm bảo bản quyền âm nhạc: chỉ lưu trữ, phát nhạc khi có giấy phép; có cơ chế gỡ bỏ nội dung vi phạm bản quyền theo yêu cầu trong vòng **48 giờ**.

### 2.2.9 Khả năng kiểm toán (Audit)

- Hệ thống phải ghi lại toàn bộ hoạt động (logs) của người dùng và quản trị viên.
- Dữ liệu log cần được sao lưu định kỳ (ví dụ: ngày 28 hàng tháng).
- Log chỉ được đọc (read-only), không thể chỉnh sửa hay xóa qua giao diện người dùng.

### 2.2.10 Hiệu quả sử dụng tài nguyên (Efficiency)

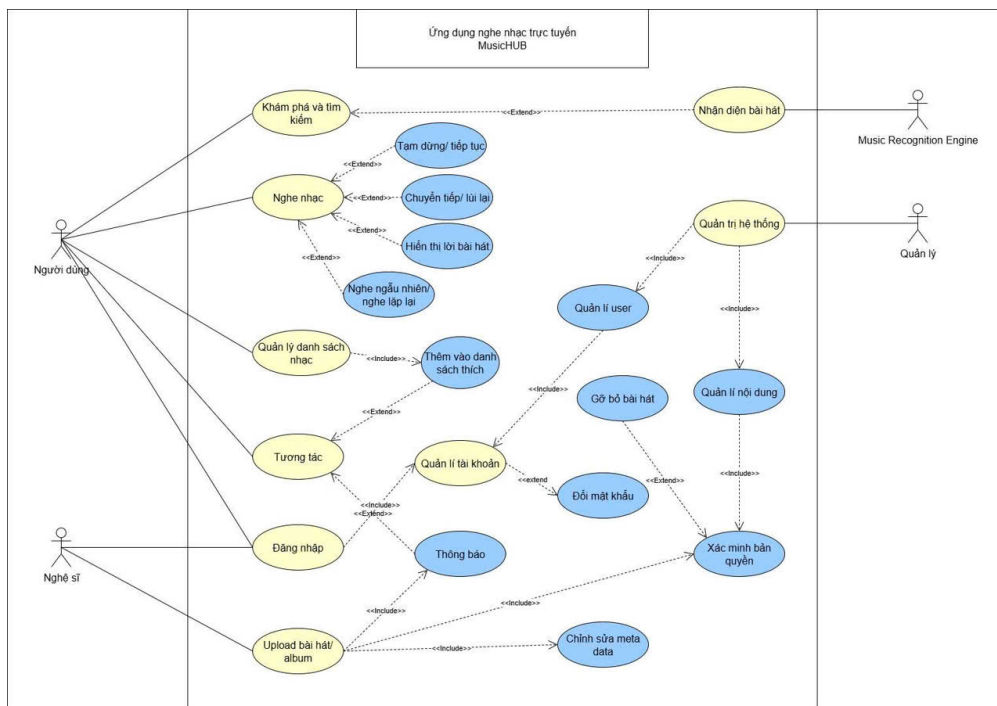
- Hệ thống phải phục vụ ổn định với lưu lượng tối đa khoảng **5000 người dùng hoạt động đồng thời**.
- Tối ưu hóa tài nguyên hệ thống (RAM, CPU) để tránh lãng phí; sử dụng **adaptive bitrate (ABR)** để giảm băng thông khi mạng yếu.
- Ứng dụng mobile tối ưu pin: giảm bitrate khi màn hình tắt (audio-only mode).

### 2.2.11 Khả năng bảo trì (Maintainability)

- Có **tài liệu kỹ thuật** đầy đủ, rõ ràng (hướng dẫn cài đặt, vận hành, xử lý sự cố).
- Các lỗi phát sinh được ghi log và gửi thông báo cho kỹ thuật viên **trong vòng 5 phút** (qua email/sms).
- Hệ thống cần được kiểm tra và bảo trì định kỳ **tối đa 3 tháng/lần**.
- Kiến trúc phần mềm module hóa, hỗ trợ CI/CD, dễ dàng thêm mới tính năng như gợi ý nhạc AI, podcast,...

### 3 Use-case Diagram và Use-case Scenario

#### 3.1 Use-case Diagram



Hình 1: Use-case Diagram

## 3.2 Use-case Scenario

### 3.2.1 Use-case Upload bài hát/album

<b>Use case name</b>	Upload bài hát/album
<b>Actors</b>	Nghệ sĩ
<b>Description</b>	Chức năng cho phép Người dùng tải lên các bài hát của mình lên hệ thống để lưu trữ, quản lý và chia sẻ với cộng đồng nghe nhạc.
<b>Trigger</b>	Người dùng chọn chức năng “Upload nhạc” trên giao diện hệ thống sau khi đăng nhập thành công.
<b>Pre-Condition(s)</b>	1. Người dùng đã đăng nhập vào hệ thống. 2. Thiết bị của Người dùng được kết nối internet. 3. File nhạc đáp ứng đúng định dạng và dung lượng cho phép (ví dụ: MP3, WAV, dung lượng tối đa 20MB).
<b>Post-Condition(s)</b>	Bài hát được lưu trữ trong hệ thống, hiển thị trong thư viện cá nhân và có thể phát trực tuyến cho Người dùng khác (nếu được chia sẻ công khai).
<b>Normal Flow</b>	1. Người dùng chọn vào phần “Upload nhạc”. 2. Hệ thống hiển thị form tải nhạc (chọn file nhạc, nhập tiêu đề, Nghệ sĩ, thể loại, ảnh bìa,...). 3. Người dùng nhập thông tin và chọn file nhạc từ thiết bị. 4. Người dùng nhấn nút “Upload”. 5. Hệ thống tiến hành tải file nhạc lên server. 6. Sau khi upload thành công, hệ thống thông báo và hiển thị bài hát trong thư viện cá nhân.
<b>Exception Flow</b>	3a. Nếu file nhạc sai định dạng hoặc vượt quá dung lượng cho phép, hệ thống thông báo lỗi. 5a. Nếu kết nối internet bị gián đoạn trong khi tải lên, hệ thống hiển thị thông báo thất bại và yêu cầu thử lại. 6a. Nếu hệ thống lỗi trong quá trình xử lý file, hiển thị thông báo “Upload thất bại”.
<b>Alternative Flow</b>	3b. Người dùng có thể hủy thao tác upload bất kỳ lúc nào để quay lại thư viện. 4b. Người dùng có thể chọn chế độ chia sẻ: công khai, riêng tư. 5b. Hệ thống hỗ trợ upload nhiều bài hát cùng lúc để tiết kiệm thời gian.

Bảng 2: Mô tả use-case Upload bài hát/album

### 3.2.2 Use-case Đăng nhập & Đăng ký

<b>Use case name</b>	Đăng nhập & Đăng ký
<b>Actors</b>	Người dùng, Nghệ sĩ
<b>Description</b>	Người Dùng có thể tạo tài khoản mới (Đăng ký) hoặc truy cập tài khoản đã có (Đăng nhập) để sử dụng ứng dụng.
<b>Trigger</b>	Người dùng mở web và chọn tính năng “Đăng nhập / Đăng ký”.
<b>Pre-Condition(s)</b>	1. Thiết bị của Người dùng phải được kết nối internet. 2. Người Dùng chưa có tài khoản (nếu đăng ký) hoặc đã có tài khoản hợp lệ (nếu đăng nhập).
<b>Post-Condition(s)</b>	Người Dùng đăng nhập thành công hoặc tạo mới tài khoản thành công và được chuyển vào giao diện chính, tài khoản đăng ký thành công sẽ được thêm vào hệ thống.
<b>Normal Flow</b>	<b>Đăng nhập:</b> 1. Người Dùng chọn “Đăng nhập”. 2. Nhập email/số điện thoại/tên đăng nhập và mật khẩu. 3. Hệ thống kiểm tra thông tin xác thực. 4. Nếu đúng → Đăng nhập thành công và hiển thị trang chủ. <b>Đăng ký:</b> 1. Người Dùng chọn “Đăng ký”. 2. Nhập thông tin cá nhân: email/số điện thoại, mật khẩu, tên hiển thị. 3. Xác thực tài khoản qua email. 4. Hệ thống tạo tài khoản mới và hiển thị thông báo thành công.
<b>Exception Flow</b>	<b>Đăng nhập:</b> 2a. Bỏ trống trường thông tin → báo lỗi “Vui lòng nhập đầy đủ thông tin”. 3a. Sai mật khẩu → báo lỗi “Sai mật khẩu”. 3b. Tài khoản bị khóa → báo lỗi “Tài khoản của bạn đã bị khóa”. <b>Đăng ký:</b> 2a. Email/số điện thoại đã tồn tại → báo lỗi “Tài khoản đã tồn tại”. 2b. Mật khẩu yếu/không hợp lệ → báo lỗi “Mật khẩu không hợp lệ”. 3a. Không xác thực được email → báo lỗi “Xác thực thất bại”.
<b>Alternative Flow</b>	<b>Đăng nhập:</b> 2b. Người Dùng chọn “Đăng nhập bằng Google/Facebook/Apple ID”. 2c. Người Dùng chọn “Quên mật khẩu” → chuyển sang luồng khôi phục mật khẩu. <b>Đăng ký:</b> 2c. Người Dùng có thể tải ảnh đại diện hoặc chọn sở thích âm nhạc ngay khi tạo tài khoản.

Bảng 3: Mô tả use-case Đăng nhập & Đăng ký



### 3.2.3 Use-case Quản lý tài khoản

<b>Use case name</b>	Quản lý tài khoản
<b>Actors</b>	Người dùng, Nghệ sĩ
<b>Description</b>	Người Dùng có thể quản lý tài khoản cá nhân bao gồm: đăng ký, đổi mật khẩu, cập nhật thông tin cá nhân (avatar, tên hiển thị, email), và xóa tài khoản.
<b>Trigger</b>	Người dùng chọn chức năng “Quản lý tài khoản” trên giao diện hệ thống sau khi đăng nhập thành công.
<b>Pre-Condition(s)</b>	1. Thiết bị của Người dùng phải được kết nối internet. 2. Người dùng đã đăng nhập thành công.
<b>Post-Condition(s)</b>	Hệ thống cập nhật thông tin tài khoản theo thao tác của Người Dùng.
<b>Normal Flow</b>	1. Người dùng chọn vào phần “Quản lý tài khoản” trong giao diện hệ thống. 2. Hệ thống hiển thị thông tin tài khoản hiện tại. 3. Người dùng chọn hành động: chỉnh sửa thông tin cá nhân / đổi mật khẩu / thiết lập bảo mật. 4. Người dùng nhập thông tin mới hoặc thay đổi cần thiết. 5. Người dùng xác nhận và lưu thay đổi. 6. Hệ thống cập nhật dữ liệu và thông báo thành công.
<b>Exception Flow</b>	3a. Nếu hệ thống không tải được thông tin tài khoản thì hiển thị lỗi “Không thể tải dữ liệu”. 4a. Nếu thông tin nhập sai định dạng (ví dụ email không hợp lệ, mật khẩu quá ngắn), hệ thống thông báo lỗi và yêu cầu nhập lại. 5a. Nếu kết nối internet bị gián đoạn trong khi lưu, hệ thống hiển thị thông báo thất bại và yêu cầu thử lại.
<b>Alternative Flow</b>	3b. Người dùng có thể hủy thao tác và quay lại trang chính mà không thay đổi gì. 4b. Người dùng có thể bật/tắt các tính năng nâng cao như xác thực 2 lớp.

Bảng 4: Mô tả use-case Quản lý tài khoản

### 3.2.4 Use-case Tương tác

<b>Use case name</b>	Tương tác
<b>Actors</b>	Người dùng
<b>Description</b>	Người dùng có thể thực hiện các hành động tương tác trong hệ thống như: thích (like) bài hát, bình luận bài hát, theo dõi nghệ sĩ, chia sẻ bài hát, nghệ sĩ, playlist lên mạng xã hội.
<b>Trigger</b>	Người dùng chọn một bài hát, nghệ sĩ hoặc playlist bất kỳ trong hệ thống và mở giao diện chi tiết.
<b>Pre-Condition(s)</b>	1. Thiết bị của Người dùng phải được kết nối internet. 2. Người dùng đã đăng nhập thành công. 3. Bài hát, nghệ sĩ đã có sẵn trên hệ thống.
<b>Post-Condition(s)</b>	Hệ thống lưu lại thông tin tương tác (like, bình luận, theo dõi, chia sẻ) và cập nhật dữ liệu liên quan.
<b>Normal Flow</b>	1. Người dùng chọn một bài hát, nghệ sĩ, playlist. 2. Hệ thống hiển thị chi tiết bài hát, nghệ sĩ, playlist. 3. Người dùng chọn hành động muốn thực hiện: 3a. Nhấn nút “Thích” (Like) để thêm vào danh sách yêu thích. 3b. Viết và gửi bình luận cho bài hát. 3c. Nhấn “Theo dõi” để theo dõi nghệ sĩ. 3d. Chọn “Chia sẻ” và lựa chọn kênh chia sẻ (Các trang mạng xã hội). 4. Hệ thống xác nhận và cập nhật thông tin tương tác.
<b>Exception Flow</b>	2a. Nếu bài hát, nghệ sĩ, playlist không tồn tại hoặc bị xóa → hiển thị thông báo lỗi. 3b1. Nếu bình luận chứa nội dung vi phạm chính sách → hệ thống từ chối đăng và hiển thị thông báo. 3d1. Nếu chia sẻ thất bại do mất kết nối internet → hiển thị thông báo “Chia sẻ không thành công, vui lòng thử lại”.
<b>Alternative Flow</b>	3a1. Người dùng có thể “Bỏ thích” (Unlike) nếu trước đó đã thích. 3b2. Người dùng có thể chỉnh sửa hoặc xóa bình luận đã đăng. 3c1. Người dùng có thể hủy theo dõi nghệ sĩ. 3d2. Người dùng có thể sao chép đường dẫn (link) thay vì chia sẻ trực tiếp.

Bảng 5: Mô tả use-case Tương tác

### 3.2.5 Use-case Nghe nhạc

<b>Use case name</b>	Nghe nhạc
<b>Actors</b>	Người dùng
<b>Description</b>	Người dùng có thể phát và điều khiển nhạc với nhiều tính năng nâng cao: phát/tạm dừng, tiếp tục, chuyển tiếp/lùi lại, bật chế độ nghe ngẫu nhiên, nghe lặp lại (một bài hoặc toàn bộ danh sách), thêm bài vào hàng chờ, cài giờ tắt nhạc (sleep timer), và xem lời bài hát.
<b>Trigger</b>	Người dùng chọn một bài hát, playlist, album và nhấn nút “Phát nhạc”.
<b>Pre-Condition(s)</b>	1. Thiết bị của Người dùng phải được kết nối internet. 2. Ứng dụng được cấp quyền truy cập âm thanh. 3. Bài hát đã có sẵn trên hệ thống và có thể phát.
<b>Post-Condition(s)</b>	Hệ thống phát nhạc theo thao tác của Người dùng, cập nhật trạng thái trình phát (player state) và lưu lại lịch sử nghe nhạc.
<b>Normal Flow</b>	1. Người dùng chọn một bài hát, playlist, album. 2. Hệ thống tải dữ liệu nhạc và bắt đầu phát. 3. Người dùng có thể thực hiện các thao tác điều khiển: 3a. Nhấn “Tạm dừng” (Pause) hoặc “Tiếp tục” (Play). 3b. Nhấn “Chuyển tiếp” (Next) hoặc “Lùi lại” (Previous). 3c. Bật chế độ “Nghe ngẫu nhiên” (Shuffle). 3d. Chọn chế độ “Nghe lặp lại” (Repeat one / Repeat all). 3e. Thêm bài hát vào hàng chờ phát (Queue). 3f. Cài đặt “Giờ tắt nhạc” (Sleep timer). 3g. Xem lời bài hát (Lyrics) nếu có sẵn. 4. Hệ thống phản hồi ngay lập tức và cập nhật trình phát nhạc. 5. Khi bài hát kết thúc, hệ thống tự động phát bài tiếp theo trong playlist/queue hoặc dừng lại nếu không còn bài nào
<b>Exception Flow</b>	2a. Nếu bài hát không thể phát (do bản quyền hoặc lỗi file) → hiển thị thông báo. 2b. Nếu người dùng bị mất kết nối trong lúc đang phát nhạc → hiển thị thông báo “Mất kết nối, vui lòng kiểm tra lại” và tạm dừng phát nhạc. 3g1. Nếu bài hát không có lời (lyrics) trong cơ sở dữ liệu → hiển thị “Chưa có lời bài hát”.
<b>Alternative Flow</b>	3a1. Người dùng có thể sử dụng tai nghe hoặc thiết bị ngoài để điều khiển (nút Play/Pause). 3e1. Người dùng có thể sắp xếp lại thứ tự bài hát trong hàng chờ. 3f2. Người dùng có thể hủy hoặc thay đổi thời gian sleep timer.

Bảng 6: Mô tả use-case Nghe nhạc

### 3.2.6 Use-case Khám phá và tìm kiếm

<b>Use case name</b>	Khám phá và tìm kiếm
<b>Actors</b>	Người dùng
<b>Description</b>	Người dùng có thể tìm kiếm bài hát, nghệ sĩ, album, playlist và khám phá nhạc mới thông qua bảng xếp hạng, xu hướng (trending) và gợi ý cá nhân hóa.
<b>Trigger</b>	Người dùng mở thanh tìm kiếm hoặc tab “Khám phá” trong ứng dụng.
<b>Pre-Condition(s)</b>	1. Thiết bị của Người dùng phải được kết nối internet. 2. Dữ liệu về bài hát, nghệ sĩ, album phải có trong cơ sở dữ liệu.
<b>Post-Condition(s)</b>	Hệ thống trả về kết quả tìm kiếm hoặc danh sách nhạc khám phá phù hợp, cho phép Người dùng chọn và nghe nhạc ngay.
<b>Normal Flow</b>	1. Người dùng mở tính năng Tìm kiếm/Khám phá. 2. Người dùng có thể thực hiện: 2a. Nhập từ khóa để tìm bài hát, nghệ sĩ, album, playlist. 2b. Xem danh sách Trending hoặc Top Chart. 2c. Nhận gợi ý nhạc cá nhân hóa dựa trên lịch sử nghe, lượt thích và nghệ sĩ theo dõi. 3. Hệ thống hiển thị danh sách kết quả. 4. Người dùng chọn nội dung mong muốn (phát nhạc, xem chi tiết nghệ sĩ, album, playlist).
<b>Exception Flow</b>	2a1. Nếu không tìm thấy kết quả → hiển thị thông báo “Không tìm thấy nội dung phù hợp”. 2a2. Nếu không nhập gì → hiển thị “Vui lòng nhập từ khóa”. 2a3. Nếu kết nối internet bị gián đoạn trong khi tìm kiếm → hiển thị thông báo “Mất kết nối, vui lòng kiểm tra lại”. 2c1. Nếu Người dùng mới/ẩn danh (không đăng nhập) chưa có lịch sử nghe → gợi ý nhạc phổ biến mặc định.
<b>Alternative Flow</b>	2a2. Người dùng có thể dùng bộ lọc nâng cao (theo thể loại, năm phát hành, nghệ sĩ, album... ). 2b2. Người dùng có thể chọn xem bảng xếp hạng theo từng khu vực/quốc gia. 2c2. Người dùng có thể cập nhật gợi ý bằng cách thay đổi sở thích cá nhân (genres, mood... ).

Bảng 7: Mô tả use-case Khám phá và tìm kiếm

### 3.2.7 Use-case Nhận diện bài hát

<b>Use case name</b>	Nhận diện bài hát
<b>Actors</b>	Người dùng
<b>Description</b>	Người dùng bật tính năng nhận diện nhạc để hệ thống nghe một đoạn nhạc từ môi trường xung quanh, sau đó so sánh với cơ sở dữ liệu fingerprint để xác định bài hát.
<b>Trigger</b>	Người dùng nhấn vào biểu tượng "Nhận diện bài hát" trên thanh tìm kiếm.
<b>Pre-Condition(s)</b>	1. Thiết bị của Người dùng phải được kết nối internet. 2. Ứng dụng được cấp quyền truy cập micro. 3. Hệ thống có cơ sở dữ liệu nhạc đủ lớn để so khớp.
<b>Post-Condition(s)</b>	Hệ thống hiển thị thông tin bài hát (tên, nghệ sĩ, album) và cung cấp tùy chọn phát nhạc hoặc lưu vào playlist.
<b>Normal Flow</b>	1. Người dùng chọn tính năng Nhận diện bài hát. 2. Hệ thống yêu cầu quyền truy cập micro và người dùng cấp phép. 3. Hệ thống ghi lại một đoạn âm thanh từ 15-20s. 4. Hệ thống trích xuất "Fingerprint" (dấu vân tay âm thanh) từ đoạn ghi âm để so khớp với dữ liệu. 5. Hệ thống trả về kết quả gồm tên bài hát, nghệ sĩ, album (nếu có). 6. Người dùng có thể chọn nghe bài hát, thêm vào playlist hoặc chia sẻ.
<b>Exception Flow</b>	2a. Người dùng từ chối quyền micro → hệ thống báo không thể sử dụng tính năng. 3a. Không thu được âm thanh do quá ồn, micro hỏng → hệ thống yêu cầu thử lại. 5a. Nếu không tìm thấy kết quả khớp → hệ thống thông báo “Không nhận diện được bài hát”. 5b. Nếu có lỗi kết nối trong quá trình so khớp → hệ thống báo “Mất kết nối” và cho phép thử lại sau.
<b>Alternative Flow</b>	5c. Người dùng có thể tìm kiếm bài hát thủ công.

Bảng 8: Mô tả use-case Nhận diện bài hát

### 3.2.8 Use-case Quản lý danh sách nhạc

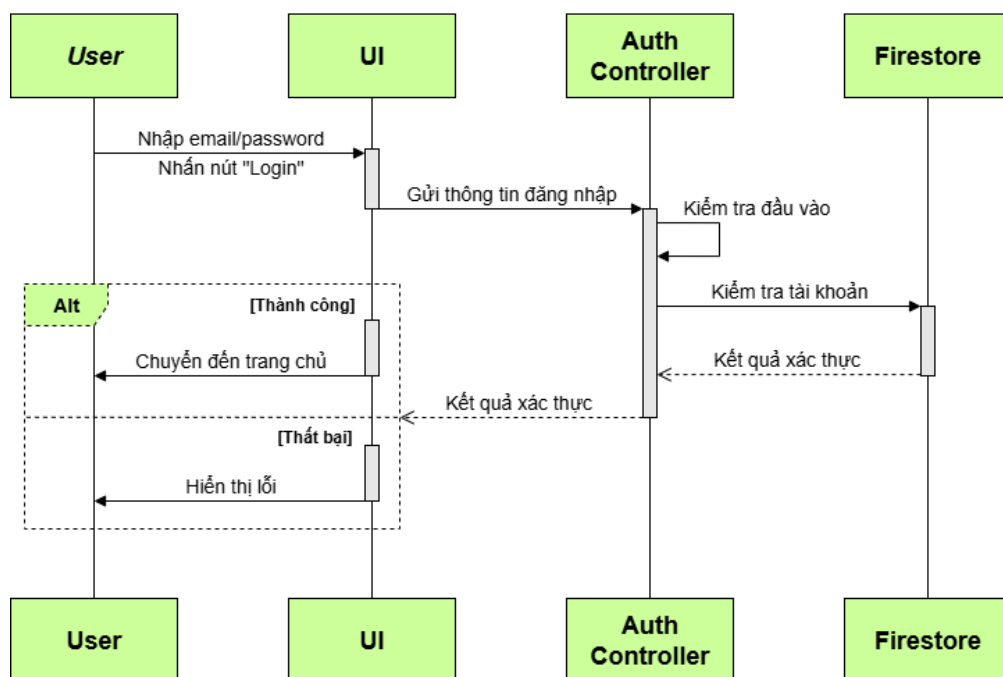
<b>Use case name</b>	Quản lý danh sách nhạc
<b>Actors</b>	Người dùng
<b>Description</b>	Cho phép người dùng xem, thêm, sửa và xóa bài hát trong danh sách nhạc của họ.
<b>Trigger</b>	Người dùng truy cập danh sách nhạc và chọn một chức năng quản lý danh sách nhạc (tạo danh sách, thêm bài hát, xóa bài hát, xóa danh sách, .v.v).
<b>Pre-Condition(s)</b>	<ol style="list-style-type: none"> <li>Thiết bị của Người dùng phải được kết nối internet.</li> <li>Người dùng đã đăng nhập thành công.</li> <li>Hệ thống có ít nhất một bài hát.</li> </ol>
<b>Post-Condition(s)</b>	<ol style="list-style-type: none"> <li>Danh sách nhạc của User được cập nhật theo hành động đã thực hiện (thêm, sửa, xóa).</li> <li>Người dùng nhận được thông báo về kết quả của hành động.</li> </ol>
<b>Normal Flow</b>	<ol style="list-style-type: none"> <li>Người dùng truy cập vào phần “Danh sách nhạc của tôi” trong giao diện hệ thống.</li> <li>Hệ thống hiển thị danh sách nhạc của người dùng hiện tại.</li> <li>Người dùng thực hiện thao tác điều khiển: <ol style="list-style-type: none"> <li>Người dùng chọn “Tạo playlist mới”, hệ thống hiển thị yêu cầu điền tên playlist, người dùng nhập và nhấn xác nhận.</li> <li>Người dùng chọn “Thêm vào playlist” với bài hát muốn thêm, hệ thống yêu cầu chọn playlist muốn thêm, người dùng chọn và nhấn xác nhận.</li> <li>Người dùng chọn “Xóa khỏi playlist” với bài hát muốn xóa, hệ thống hiển thị thông báo xác nhận, người dùng xác nhận.</li> <li>Người dùng chọn “Xóa playlist”, hệ thống hiển thị thông báo xác nhận, người dùng xác nhận.</li> </ol> </li> <li>Hệ thống ghi nhận và cập nhật danh sách playlist của người dùng.</li> </ol>
<b>Exception Flow</b>	<ol style="list-style-type: none"> <li>Nếu lỗi xảy ra (mất kết nối máy chủ, hệ thống bận, .v.v), mọi yêu cầu từ người dùng đều bị từ chối và hệ thống sẽ hiển thị thông báo “Đang trong quá trình bảo trì”.</li> <li>Nếu người dùng nhập tên playlist đã tồn tại, hệ thống thông báo lỗi và yêu cầu người dùng nhập lại.</li> <li>Nếu người dùng thêm 1 bài hát đã tồn tại trong playlist, hệ thống hiển thị thông báo đã tồn tại và không cập nhật bài hát.</li> </ol>
<b>Alternative Flow</b>	<ol style="list-style-type: none"> <li>Người dùng không đặt tên mà trực tiếp xác nhận, hệ thống vẫn sẽ ghi nhận và đặt tên theo format mặc định “Unname Playlist”.</li> <li>3c/3d. Nếu người dùng không xác nhận, hệ thống hoàn tác toàn bộ thao tác và cập nhật lại trạng thái ban đầu của danh sách playlist.</li> </ol>

Bảng 9: Mô tả use-case Quản lý danh sách nhạc

### 3.2.9 Use-case Quản trị hệ thống

<b>Use case name</b>	Quản trị hệ thống
<b>Actors</b>	Quản lý
<b>Description</b>	Cho phép quản lý thực hiện các tác vụ quản trị để duy trì và điều hành hệ thống.
<b>Trigger</b>	Quản lý truy cập vào trang quản lý hệ thống trong giao diện quản lý.
<b>Pre-Condition(s)</b>	1. Thiết bị của quản lý phải được kết nối internet. 2. Quản lý đã đăng nhập thành công <b>dưới quyền quản trị</b> .
<b>Post-Condition(s)</b>	1. Các thay đổi về người dùng, nội dung, hoặc bản quyền đã được cập nhật thành công trong hệ thống. 2. Hệ thống duy trì trạng thái ổn định và các quy định được tuân thủ.
<b>Normal Flow</b>	1. Quản lý truy cập vào trang quản lý hệ thống với tài khoản có quyền quản trị. 2. Hệ thống điều hướng đến trang quản trị hệ thống. 3. Quản lý có thể lựa chọn hành động: 3a. Xem, sửa (đối với quyền truy cập), xóa người dùng. 3b. Phê duyệt, gỡ bỏ nội dung, bài hát, album công khai thủ công. 3c. Thay đổi các thông tin của nội dung phù hợp với chính sách. 3d. Kiểm tra tính minh bạch và bản quyền của nội dung đăng tải. 4. Hệ thống thông báo thao tác thành công. 5. Hệ thống tự động ghi log đảm bảo tính minh bạch về sau.
<b>Exception Flow</b>	3.1. Nếu quản lý cố gắng truy cập một chức năng không có quyền, hệ thống sẽ thông báo lỗi "Bạn không có quyền truy cập chức năng này". 3.2. Nếu có lỗi trong quá trình cập nhật dữ liệu (thông tin không hợp lệ, .v.v), hệ thống sẽ hiển thị thông báo lỗi và không thực hiện thay đổi. 3.3. Nếu lỗi xảy ra (mất kết nối máy chủ, hệ thống bận, .v.v), mọi thông tin thay đổi sẽ được lưu tạm thời và hệ thống sẽ hiển thị thông báo "Đang trong quá trình bảo trì" cho tới khi có kết nối lại và tự động thực hiện lại thao tác.
<b>Alternative Flow</b>	1a. Tài khoản không có quyền thì hệ thống tự động điều hướng đến giao diện người dùng bình thường. 3a/3b/3c/3d. Nếu thao tác không được xác nhận, hệ thống hoàn tác toàn bộ thao tác và cập nhật lại trạng thái ban đầu của nội dung.

Bảng 10: Mô tả use-case Quản trị hệ thống



Hình 2: Sequence Diagram - Login

## 4 Thiết kế hành vi và giao diện người dùng

### 4.1 Đăng nhập - đăng kí

#### 4.1.1 Đăng nhập - Login

«««< HEAD

- **Sequence Diagram:**

- **Mục đích:** Thể hiện quy trình đăng nhập của người dùng, bao gồm các bước từ khi người dùng nhập thông tin đến khi hệ thống xác thực và phản hồi kết quả.

- **Thành phần liên quan:**

- Người dùng (User): người thực hiện thao tác đăng nhập.
- Giao diện đăng nhập (Login UI): giao diện để người dùng nhập thông tin và hiển thị kết quả.
- Hệ thống xác thực (Auth Controller): nhận thông tin đăng nhập và xử lý xác thực.
- Cơ sở dữ liệu người dùng (Firestore): lưu trữ thông tin tài khoản.

- **Luồng tương tác:**

1. Người dùng mở trang đăng nhập.
2. Người dùng nhập email + mật khẩu và nhấn nút “Đăng nhập”.



3. Giao diện đăng nhập gửi yêu cầu đăng nhập đến Hệ thống xác thực.
4. Hệ thống xác thực kiểm tra dữ liệu đầu vào (có trống không, định dạng hợp lệ không).
5. Nếu hợp lệ, Hệ thống xác thực truy vấn Cơ sở dữ liệu người dùng để tìm tài khoản.
6. Cơ sở dữ liệu trả về thông tin tài khoản (nếu tồn tại).
7. Hệ thống xác thực phản hồi lại cho Giao diện đăng nhập:
  - Nếu đúng: thông báo đăng nhập thành công và chuyển đến giao diện trang chủ.
  - Nếu sai: thông báo lỗi (sai mật khẩu, tài khoản không tồn tại).

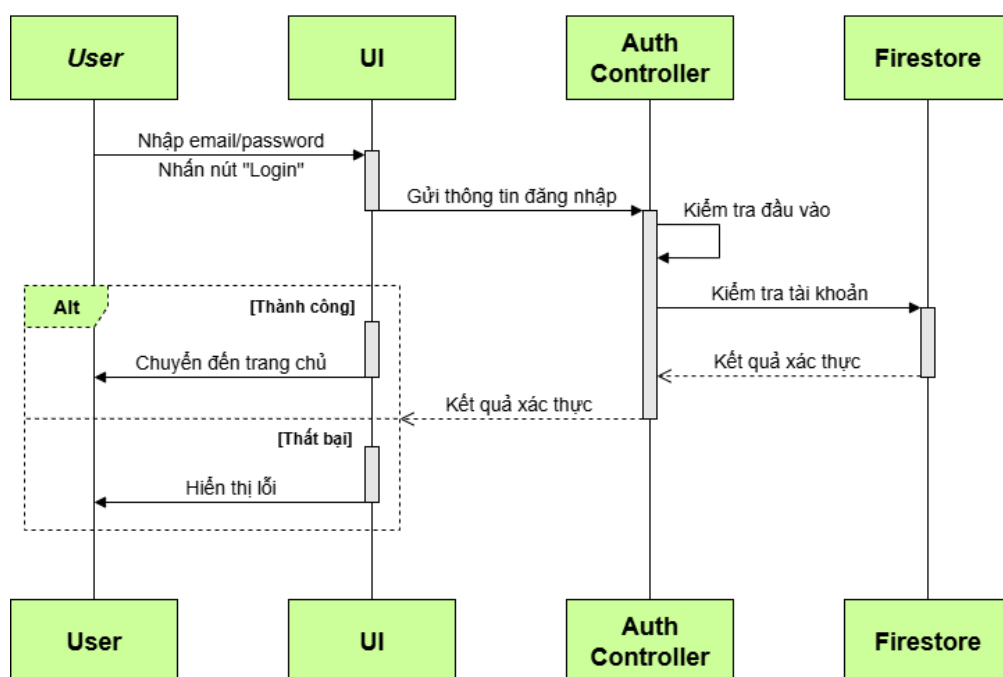
- **Activity Diagram:**

- **UI:**

#### 4.1.2 Đăng ký - Register

=====

Sequence Diagram:

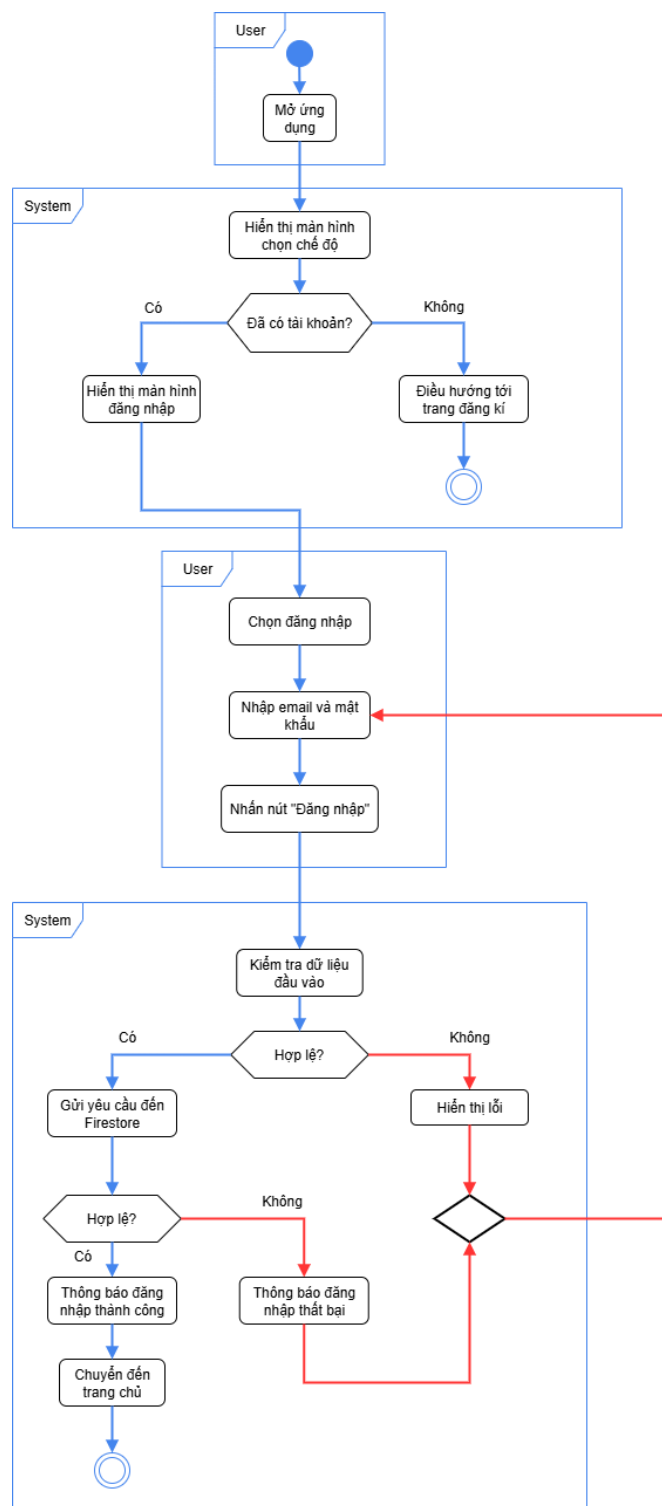


Hình 4: Sequence Diagram - Login

**Mục đích:** Thể hiện quy trình đăng nhập của người dùng, bao gồm các bước từ khi người dùng nhập thông tin đến khi hệ thống xác thực và phản hồi kết quả.

**Thành phần liên quan:**

- Người dùng (User): người thực hiện thao tác đăng nhập.
- Giao diện đăng nhập (Login UI): giao diện để người dùng nhập thông tin và hiển thị kết quả.



Hình 3: Activity Diagram - Login

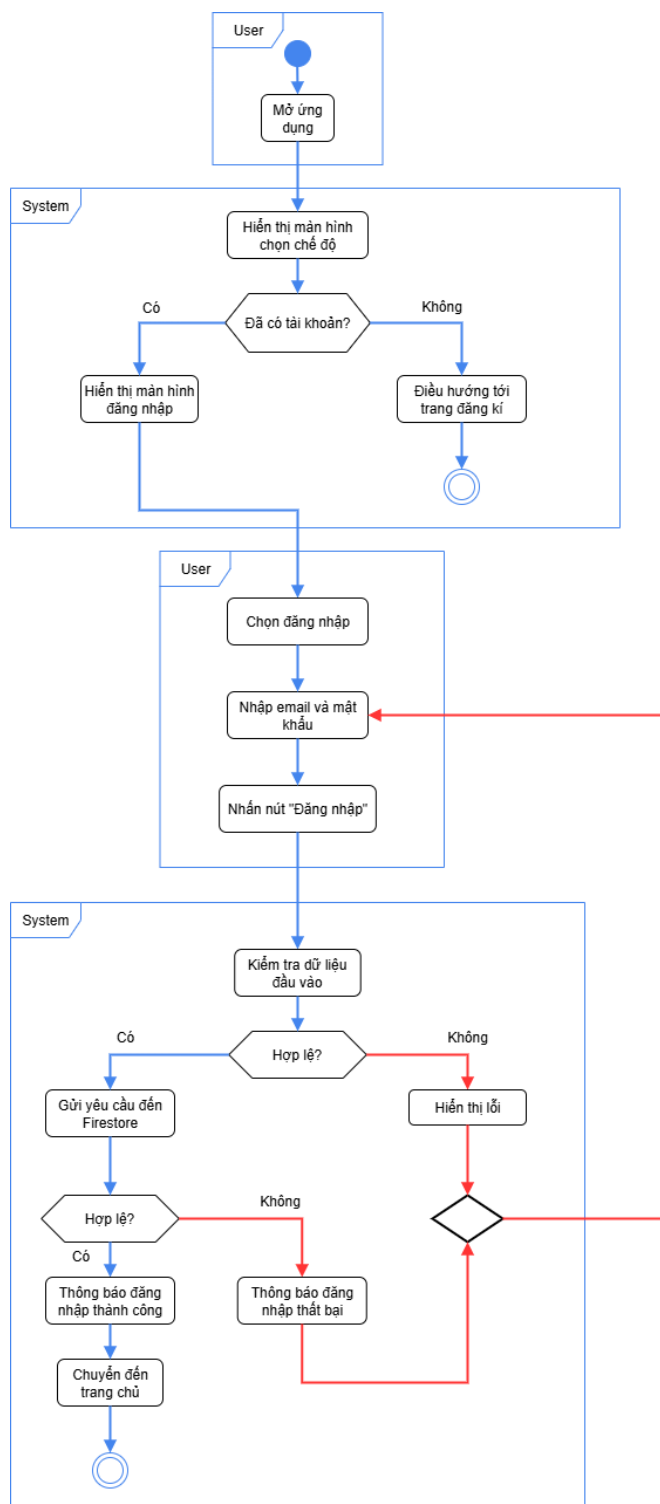


- Hệ thống xác thực (Auth Controller): nhận thông tin đăng nhập và xử lý xác thực.
- Cơ sở dữ liệu người dùng (Firestore): lưu trữ thông tin tài khoản.

**Luồng tương tác:**

1. Người dùng mở trang đăng nhập.
2. Người dùng nhập email + mật khẩu và nhấn nút “Đăng nhập”.
3. Giao diện đăng nhập gửi yêu cầu đăng nhập đến Hệ thống xác thực.
4. Hệ thống xác thực kiểm tra dữ liệu đầu vào (có trống không, định dạng hợp lệ không).
5. Nếu hợp lệ, Hệ thống xác thực truy vấn Cơ sở dữ liệu người dùng để tìm tài khoản.
6. Cơ sở dữ liệu trả về thông tin tài khoản (nếu tồn tại).
7. Hệ thống xác thực phản hồi lại cho Giao diện đăng nhập:
  - Nếu đúng: thông báo đăng nhập thành công và chuyển đến giao diện trang chủ.
  - Nếu sai: thông báo lỗi (sai mật khẩu, tài khoản không tồn tại).

**Activity Diagram:**



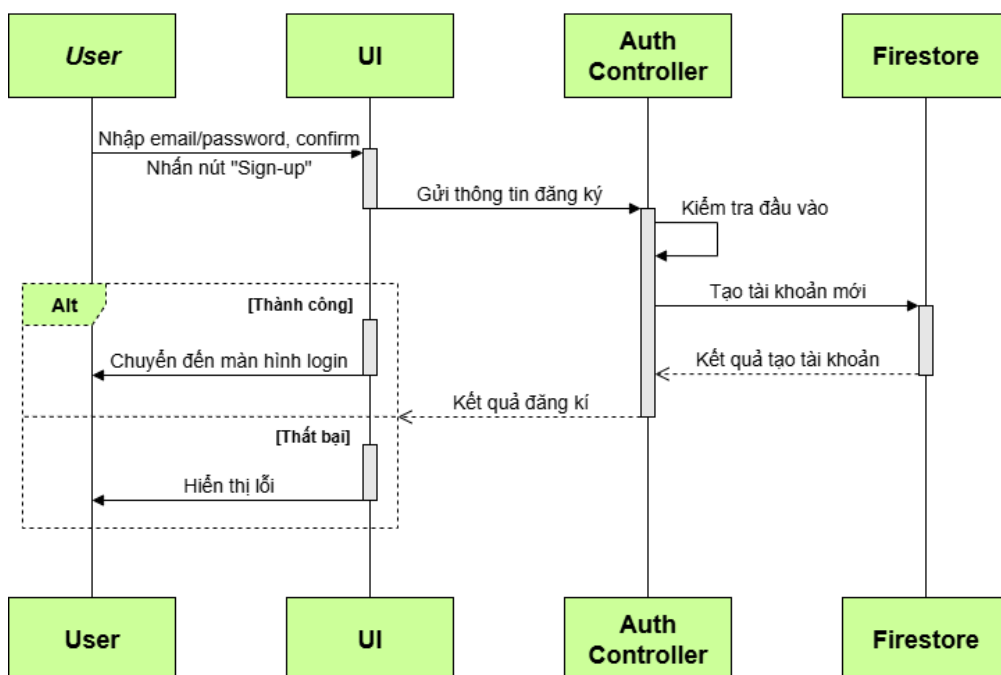
Hình 5: Activity Diagram - Login

UI:

### 4.1.3 Đăng ký - Register

»»»» > main

- Sequence Diagram:



Hình 6: Sequence Diagram - Register

- **Mục đích:** Thể hiện quy trình đăng ký tài khoản mới của người dùng, bao gồm các bước từ khi người dùng nhập thông tin đến khi hệ thống tạo tài khoản và phản hồi kết quả.

- **Thành phần liên quan:**

- Người dùng (User): người thực hiện thao tác đăng ký.
- Giao diện đăng ký (Register UI): giao diện để người dùng nhập thông tin và hiển thị kết quả.
- Hệ thống xác thực (Auth Controller): nhận thông tin đăng ký và xử lý xác thực.
- Cơ sở dữ liệu người dùng (Firestore): lưu trữ thông tin tài khoản.

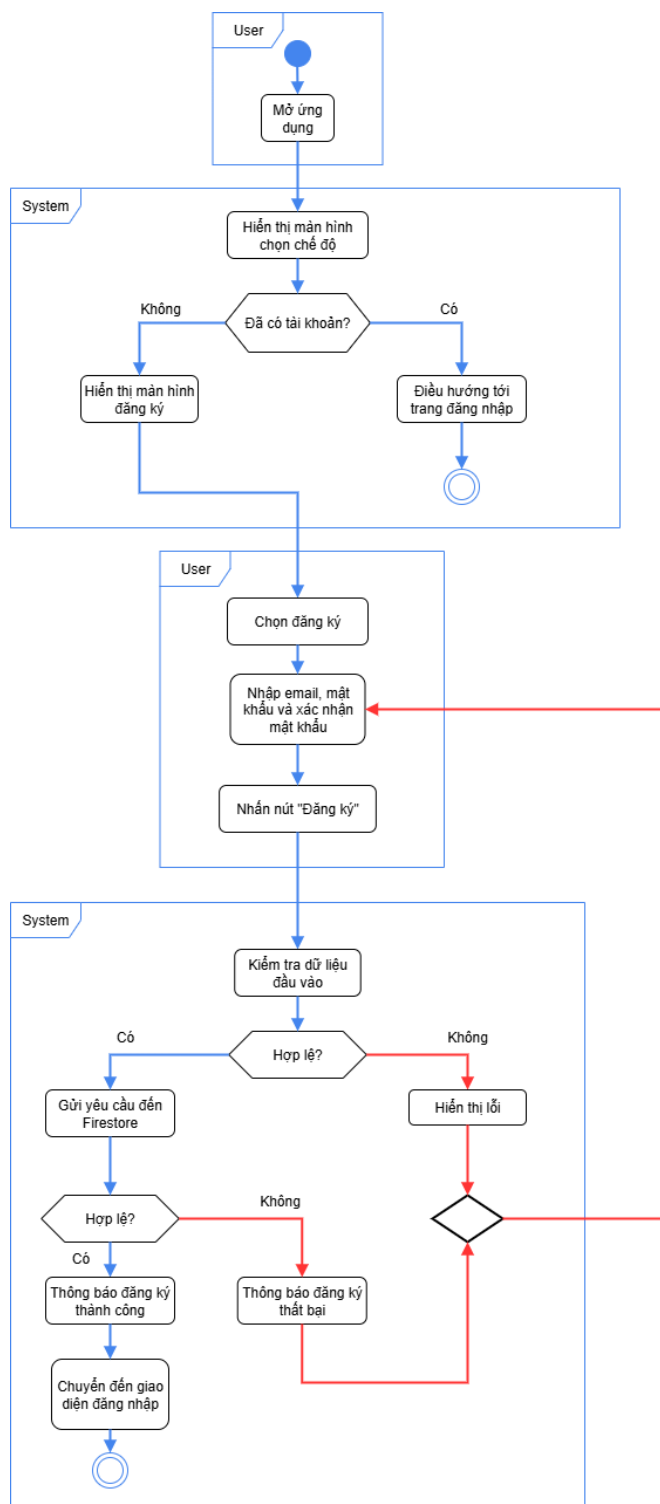
- **Luồng tương tác:**

1. Người dùng mở trang đăng ký.
2. Người dùng nhập email, mật khẩu và xác nhận mật khẩu, sau đó nhấn nút “Đăng ký”.



3. Giao diện đăng ký gửi yêu cầu tạo tài khoản đến Hệ thống xác thực.
4. Hệ thống xác thực kiểm tra dữ liệu đầu vào (có trống không, mật khẩu có khớp không, định dạng email hợp lệ không).
5. Nếu hợp lệ, Hệ thống xác thực truy vấn Cơ sở dữ liệu người dùng để kiểm tra xem tài khoản đã tồn tại hay chưa.
6. Cơ sở dữ liệu phản hồi kết quả kiểm tra.
7. Hệ thống xác thực phản hồi lại cho Giao diện đăng ký:
  - Nếu tài khoản chưa tồn tại: tạo tài khoản mới, thông báo đăng ký thành công và điều hướng người dùng đến giao diện đăng nhập.
  - Nếu tài khoản đã tồn tại hoặc dữ liệu không hợp lệ: thông báo lỗi (email đã tồn tại, mật khẩu không khớp, định dạng sai, v.v.).

• **Activity Diagram:**



Hình 7: Activity Diagram - Register

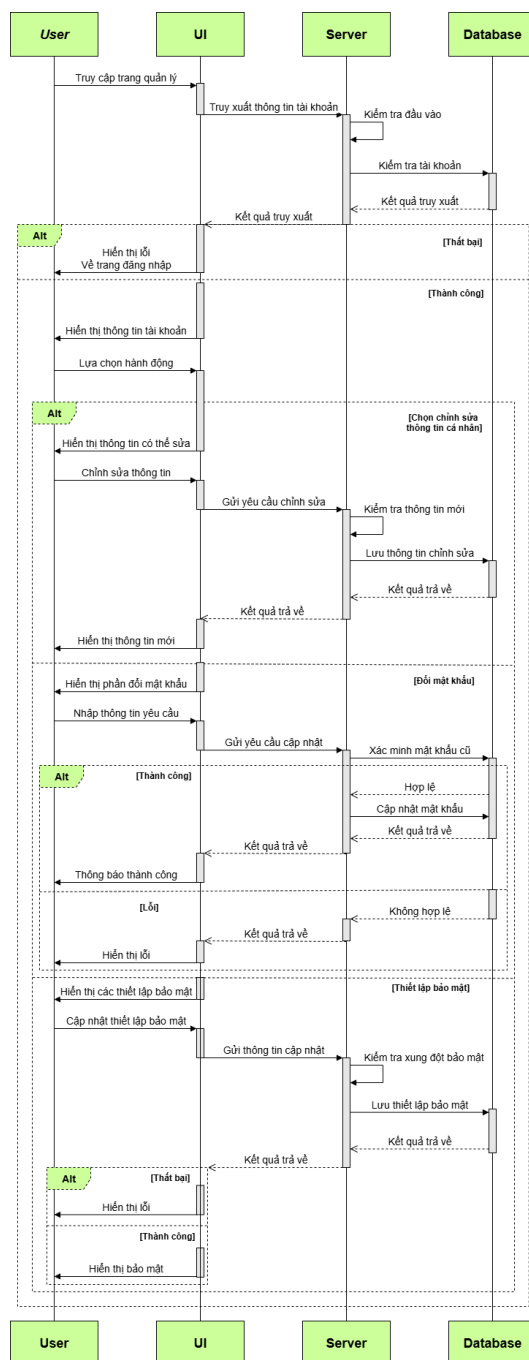


- UI:

#### 4.1.4 Quản lý tài khoản - Account Management

- Sequence Diagram:





Hình 8: Sequence Diagram - Account Management

- **Mục đích:** Thể hiện các tùy chọn quản lý tài khoản cho người dùng.
- **Thành phần liên quan:**

- Người dùng (User): người thực hiện thao tác đăng ký.
- Giao diện quản lý (Management UI): giao diện để người dùng nhập thông tin và hiển thị kết quả.
- Hệ thống xử lý (Server): nhận thông tin xác thực dữ liệu.
- Cơ sở dữ liệu người dùng (Database): lưu trữ thông tin tài khoản.

• **Luồng tương tác:**

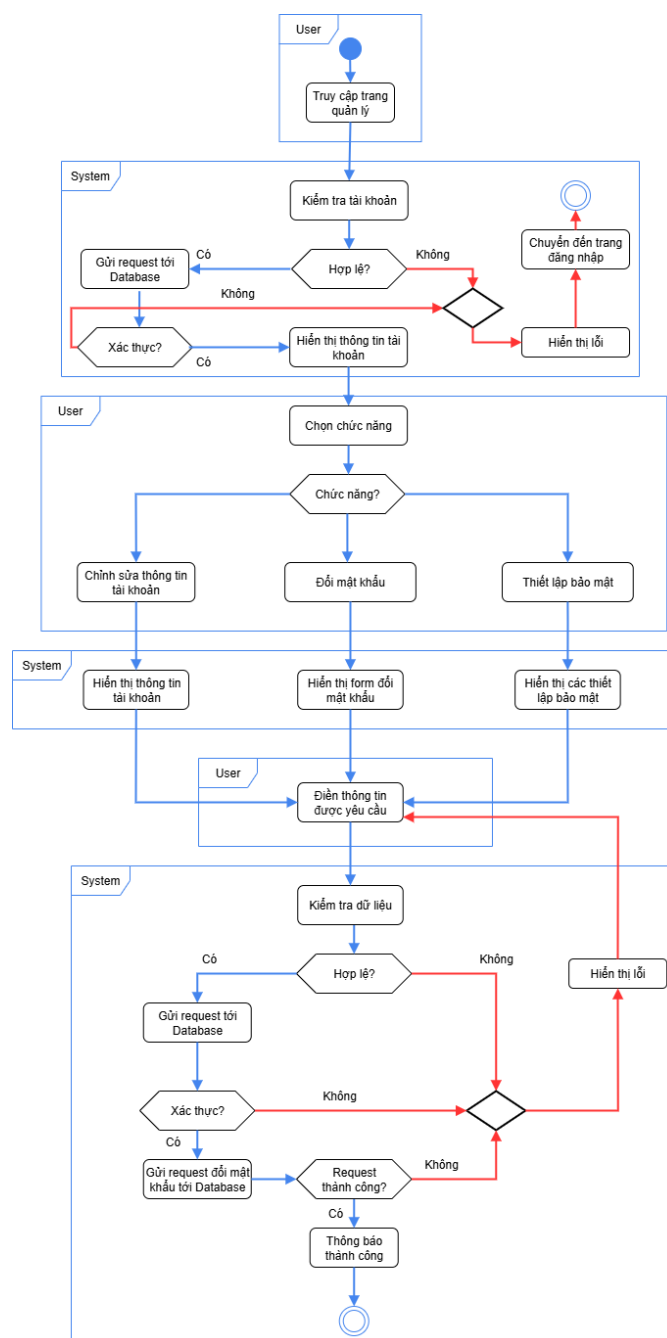
1. Người dùng mở trang quản lý tài khoản.
2. Hệ thống xử lý gửi yêu cầu đến cơ sở dữ liệu.
3. Cơ sở dữ liệu phản hồi kết quả kiểm tra.
4. Hệ thống kiểm tra trạng thái tài khoản, nếu chưa hợp lệ, thông báo lỗi và điều hướng về trang đăng nhập.
5. Nếu hợp lệ, hệ thống trả về thông tin tài khoản và hiển thị lên giao diện.
6. Người dùng lựa chọn chức năng:
  - Nếu người dùng chọn chỉnh sửa thông tin tài khoản:
    - \* Giao diện hiển thị thông tin tài khoản có thể chỉnh sửa.
    - \* Người dùng tiến hành chỉnh sửa và xác nhận.
    - \* Hệ thống tiếp nhận yêu cầu sửa chữa và kiểm tra tính đúng đắn của dữ liệu.
    - \* Nếu thông tin hợp lệ, hệ thống gửi yêu cầu cập nhật dữ liệu đến cơ sở dữ liệu.
    - \* Cơ sở dữ liệu cập nhật dữ liệu mới và phản hồi nội dung cập nhật.
    - \* Hệ thống gửi trả nội dung mới.
    - \* Giao diện cập nhật theo nội dung mới của người dùng.
  - Nếu người dùng chọn đổi mật khẩu:
    - \* Giao diện hiển thị biểu mẫu nhập mật khẩu cũ và mật khẩu mới.
    - \* Người dùng nhập dữ liệu theo yêu cầu và xác nhận.
    - \* Hệ thống tiếp nhận mật khẩu cũ và gửi đến cơ sở dữ liệu kiểm tra hợp lệ.
    - \* Cơ sở dữ liệu phản hồi kết quả.
    - \* Nếu dữ liệu hợp lệ, hệ thống gửi thông tin mật khẩu mới đến cơ sở dữ liệu.
    - \* Cơ sở dữ liệu cập nhật mật khẩu mới và phản hồi nội dung cập nhật.
    - \* Hệ thống gửi trả phản hồi thành công.
    - \* Giao diện thông báo cập nhật thành công.
  - Nếu người dùng chọn thiết lập bảo mật:
    - \* Giao diện hiển thị các thiết lập bảo mật.
    - \* Người dùng xác nhận các thiết lập bảo mật.
    - \* Hệ thống tiếp nhận và gửi yêu cầu đến cơ sở dữ liệu.
    - \* Cơ sở dữ liệu cập nhật và phản hồi nội dung cập nhật.
    - \* Hệ thống gửi trả phản hồi thành công.



\* Giao diện cập nhật thiết lập mới nhất.

7. Nếu phản hồi là thất bại, giao diện hiển thị lỗi cho người dùng và khôi phục thiết lập ban đầu.

- **Activity Diagram:**



Hình 9: Activity Diagram - Account Management

- UI: