KHOA CÔNG NGHỆ THÔNG TIN

BỘ MÔN CÔNG NGHỆ PHẦN MỀM

\*\*\*\*\*

Description: Icon

Description automatically generated

ĐỒ ÁN

**THIẾT KẾ KIẾN TRÚC**

**PHẦN MỀM**

Sinh viên thực hiện: **Nguyễn Gia Cường <0182366>**

**Nguyễn Mạnh Cường <0182666>**

Lớp học phần: **66PM**

Giảng viên hướng dẫn: Gv. **Phạm Hữu Tùng**

Mục lục

[Chương 1. Giới thiệu đề tài 3](#_Toc199157770)

[1.1. Lý do chọn đề tài 3](#_Toc199157771)

[1.2. Mục tiêu của đồ án 4](#_Toc199157772)

[1.3. Phạm vi và giới hạn 4](#_Toc199157773)

[1.4. Phương pháp thực hiện 4](#_Toc199157774)

[Chương 2. Phân tích hệ thống 5](#_Toc199157775)

[2.1. Mô tả bài toán 5](#_Toc199157776)

[2.2. Yêu cầu chức năng 5](#_Toc199157777)

[2.3. Yêu cầu phi chức năng 5](#_Toc199157778)

[2.4. Use Case Diagram tổng quát 6](#_Toc199157779)

[2.5. Mô tả các Use Case chính 7](#_Toc199157780)

[2.5.1. Use Case Đăng nhập 7](#_Toc199157781)

[2.5.2. Use Case Đăng ký 8](#_Toc199157782)

[2.5.3. Use Case Phát Nhạc 8](#_Toc199157783)

[2.5.4. Use Case Quản lý playlist riêng 10](#_Toc199157784)

[2.6. Biểu đồ hoạt động 13](#_Toc199157785)

[2.6.1. Use Case Đăng nhập 13](#_Toc199157786)

[2.6.2. Use Case Đăng ký 14](#_Toc199157787)

[2.6.3. Use Case Phát nhạc 15](#_Toc199157788)

[2.6.4. Use Case Tạo playlist 16](#_Toc199157789)

[2.6.5. Use Case Thêm bài hát 16](#_Toc199157790)

[Chương 3. Thiết kế hệ thống 17](#_Toc199157791)

[3.1. Lựa chọn mô hình kiến trúc 17](#_Toc199157792)

[3.2. Sơ đồ kiến trúc tổng thể 18](#_Toc199157793)

[3.3. Biểu đồ tuần tự 19](#_Toc199157794)

[3.3.1. Use Case Đăng ký 19](#_Toc199157795)

[3.3.2. Use Case Đăng nhập 20](#_Toc199157796)

[3.3.3. Use Case Phát nhạc 21](#_Toc199157797)

[3.3.4. Use Case Tạo Playlist 22](#_Toc199157798)

[3.3.5. Use Case Thêm bài hát 23](#_Toc199157799)

[3.4. Thiết kế lớp 24](#_Toc199157800)

[3.5. Thiết kế cơ sở dữ liệu 25](#_Toc199157801)

[3.6. Thiết kế giao diện người dùng 26](#_Toc199157802)

[3.6.1. Trang chủ 26](#_Toc199157803)

[3.6.2. Trang đăng ký 26](#_Toc199157804)

[3.6.3. Trang đăng nhập 27](#_Toc199157805)

[3.6.4. Trang quên mật khẩu 27](#_Toc199157806)

[3.6.5. Trang Favorite 28](#_Toc199157807)

[3.6.6. Trang Playlist 28](#_Toc199157808)

[3.6.7. Trang Profile 29](#_Toc199157809)

[Chương 4. Cài đặt và triển khai hệ thống 30](#_Toc199157810)

[4.1. Môi trường triển khai 30](#_Toc199157811)

[4.2. Cài đặt hệ thống 30](#_Toc199157812)

[4.3. Yêu cầu hệ thống 30](#_Toc199157813)

[4.4. Các bước cài đặt chương trình 30](#_Toc199157814)

[Chương 5. Kết quả và đánh giá 31](#_Toc199157815)

[5.1.Kết quả thử nghiệm hệ thống 31](#_Toc199157816)

[5.2.Đánh giá hiệu quả hệ thống 31](#_Toc199157817)

# Chương 1. Giới thiệu đề tài

## Lý do chọn đề tài

Đề tài ‘Web nghe nhạc’ không phải là đề tài đầu tiên mà bọn em nghĩ đến, tuy nhiên khi họp bàn, xem xét thời gian, khả năng code của nhóm thì bọn em chọn đề tài này vì nó phù hợp nhất. Với mục đích tạo ra một trang web nghe nhạc miễn phí 3 tiêu chí, không quảng cáo, không mất phí, không phức tạp, cũng như là tài liệu học tập cho các bạn trẻ đang hứng thú ngành Công Nghệ Thông Tin.

## Mục tiêu của đồ án

• Xây dựng một nền tảng nghe nhạc trực tuyến có giao diện dễ dùng, phù hợp với mọi người.

• Tích hợp đầy đủ các chức năng điều khiển: phát, dừng, tua nhạc, phát bài tiếp theo hoặc bài trước, xem lyric.

## Phạm vi và giới hạn

Phạm vi của đề tài:

* Ứng dụng web hỗ trợ người dùng nghe nhạc trực tuyến qua giao diện website.
* Cho phép hiển thị danh sách bài hát, nghệ sĩ, playlist.
* Hỗ trợ các chức năng phát nhạc: phát/tạm dừng, tua, chuyển bài, điều chỉnh âm lượng.
* Hiển thị lời bài hát khi người dùng yêu cầu.

Giới hạn của đề tài:

* Hệ thống không có chức năng tải nhạc về thiết bị.
* Không tích hợp tính năng gợi ý bài hát theo sở thích người dùng.
* Không có hệ thống phân quyền người dùng (quản trị, người dùng thường).
* Chỉ sử dụng HTML/CSS/JS thuần, không sử dụng framework frontend như React/Vue/Angular.

## Phương pháp thực hiện

Để xây dựng website nghe nhạc, nhóm tiến hành theo các bước sau:

• **Tìm hiểu và phân tích yêu cầu**

* Nghiên cứu các trang web nghe nhạc phổ biến như Zing MP3, Nhaccuatui để xác định các chức năng cơ bản.
* Phân tích nhu cầu người dùng và xác định các chức năng cần thiết như: phát nhạc, danh sách phát, lời bài hát,….

• **Thiết kế hệ thống**

* Thiết kế giao diện người dùng (UI) bằng HTML và CSS theo hướng đơn giản, dễ sử dụng.
* Thiết kế luồng hoạt động của các chức năng chính: phát nhạc, chuyển bài, tạo playlist…

• **Xây dựng chức năng**

* Sử dụng HTML/CSS/JavaScript thuần để xây dựng giao diện và các tương tác người dùng.
* Sử dụng API để lấy danh sách bài hát, thông tin nghệ sĩ, đường dẫn nhạc,...
* Tổ chức logic JavaScript để xử lý phát nhạc, điều khiển playlist và đồng bộ giao diện.

• **Kiểm thử và sửa lỗi**

* Thực hiện kiểm thử chức năng trên trình duyệt Chrome.
* Sửa lỗi và tối ưu mã nguồn.

• **Hoàn thiện sản phẩm**

* Chuẩn bị báo cáo và trình bày sản phẩm.

# Chương 2. Phân tích hệ thống

## 2.1. Mô tả bài toán

Xây dựng một website nghe nhạc dạng SPA bằng HTML/CSS/JS thuần, giao tiếp với API backend để lấy dữ liệu bài hát, playlist, nghệ sĩ và thông tin người dùng. Hệ thống cho phép người dùng nghe nhạc trực tuyến với giao diện mượt mà, thân thiện, không cần tải lại trang.

## 2.2. Yêu cầu chức năng

* Đăng ký / Đăng nhập
* Người dùng có thể đăng ký tài khoản với thông tin: email, mật khẩu, tên hiển thị.
* Cho phép đăng nhập bằng email và mật khẩu.
* Hỗ trợ tính năng quên mật khẩu và đổi mật khẩu.
* Quản lý hồ sơ cá nhân
* Người dùng có thể xem và cập nhật hồ sơ cá nhân.
* Xem danh sách các playlist riêng của mình.
* Xem lịch sử nghe nhạc.
* Tìm kiếm bài hát hoặc nghệ sĩ yêu thích theo từ khóa.
* Hiển thị kết quả tìm kiếm một cách nhanh chóng và chính xác.
* Phát nhạc và hiển thị lời bài hát
* Cho phép phát nhạc khi người dùng nhấn vào bất kỳ bài hát nào trong trang.
* Xem lyric của bài hát đang phát để hát theo.
* Trình phát hỗ trợ các thao tác: phát, tạm dừng, tua, chuyển bài, điều chỉnh âm lượng.
* Quản lý playlist và bài hát yêu thích
* Tạo playlist cá nhân và thêm bài hát tùy thích.
* Nghe nhạc từ danh sách bài hát yêu thích đã lưu trước đó.
* Thêm hoặc gỡ bài hát khỏi danh sách yêu thích.

## 2.3. Yêu cầu phi chức năng

* Hệ thống cần đảm bảo:
* Giao diện thân thiện, dễ sử dụng.
* Tương thích trên nhiều trình duyệt.
* Phản hồi nhanh, hiệu năng cao.
* Thư viện bài hát phải lớn.

## 2.4. Use Case Diagram tổng quát

A diagram of a person with text

AI-generated content may be incorrect.

**Hình 1. Use Case tổng quát**

## 2.5. Mô tả các Use Case chính

### 2.5.1. Use Case Đăng nhập

A person standing next to a blue square

AI-generated content may be incorrect.

**Hình 2. Use Case Đăng nhập**

|  |  |
| --- | --- |
| **ID** | UC1 |
| **Name** | Đăng nhập |
| **Brief description** | Chức năng cho phép người dùng chưa đăng nhập truy cập vào tài khoản cá nhân để sử dụng các tính năng của hệ thống. |
| **Actors** | Người dùng |
| **Pre-conditions** | Người dùng đã có tài khoản trong hệ thống. |
| **Basic flow** | * Người dùng truy cập giao diện đăng nhập. * Người dùng nhập thông tin đăng nhập (email/tên người dùng và mật khẩu). * Người dùng nhấn nút "Đăng nhập". * Hệ thống kiểm tra thông tin: * Nếu thông tin hợp lệ: chuyển người dùng tới giao diện chính của hệ thống. * Nếu thông tin không hợp lệ: hiển thị thông báo lỗi. |
| **Alternative flow** | Nếu người dùng quên mật khẩu:   * Người dùng nhấn vào nút “Quên mật khẩu”. * Hệ thống yêu cầu người dùng nhập email để đặt mới mật khẩu. * Hệ thống gửi hướng dẫn đặt mới mật khẩu qua email. |
| **Result** | Người dùng đăng nhập thành công vào hệ thống. |
| **Exceptions** | Tên đăng nhập hoặc mật khẩu không chính xác. |

### 2.5.2. Use Case Đăng ký

|  |  |
| --- | --- |
| **ID** | UC2 |
| **Name** | Đăng ký |
| **Brief description** | Chức năng cho phép người dùng tạo tài khoản để sử dụng các tính năng của hệ thống. |
| **Actors** | Người dùng |
| **Pre-conditions** | Người dùng chưa có tài khoản trong hệ thống. |
| **Basic flow** | * Người dùng truy cập giao diện đăng ký. * Người dùng nhập thông tin đăng nhập (email/tên người dùng, mật khẩu và xác nhận mật khẩu). * Người dùng nhấn nút "Đăng ký". * Hệ thống kiểm tra thông tin: * Nếu thông tin hợp lệ: thông báo thành công và chuyển người dùng tới giao diện đăng nhập của hệ thống. * Nếu thông tin không hợp lệ: hiển thị thông báo lỗi. |
| **Alternative flow** | Nếu thông tin nhập không hợp lệ thì hệ thống hiển thị thông báo lỗi:   * “Email không hợp lệ, vui lòng nhập lại” * "Mật khẩu phải có tối thiểu 8 ký tự, bao gồm chữ hoa, chữ thường và ký tự đặc biệt." |
| **Result** | Người dùng đăng ký tài khoản thành công vào hệ thống. |
| **Exceptions** | Hệ thống gặp lỗi khi tạo tài khoản mới |

### 2.5.3. Use Case Phát Nhạc

|  |  |
| --- | --- |
| **ID** | UC3 |
| **Name** | Phát Nhạc |
| **Brief description** | Chức năng cho phép người dùng phát nhạc từ hàng đợi danh sách bài hát, danh sách tìm kiếm, playlist. |
| **Actors** | Người dùng |
| **Pre-conditions** | Người dùng đã đăng nhập vào hệ thống. |
| **Basic flow** | * Người dùng truy cập giao diện trang chủ, playlist,... * Người dùng có thể click vào bài hát ở chỗ mà họ thấy * Hệ thống tải và phát nhạc thông qua trình phát. * Hệ thống hiển thị các điều khiển trình phát |
| **Alternative flow** | Nếu người dùng chuyển bài khác khi bài hiện tại đang phát:   * Người dùng nhấn nút next hoặc chọn bài khác khi bài hiện tại đang phát. * Hệ thống hủy phát bài hiện tại và bắt đầu phát bài mới (xử lý bình thường, không lỗi). |
| **Result** | Nhạc phát thành công, người dùng có thể sử dụng các nút chức năng. |
| **Exceptions** | File nhạc bị lỗi thì hiển thị thông báo lỗi |

### 2.5.4. Use Case Quản lý playlist riêng

A diagram of a person with text

AI-generated content may be incorrect.

**Hình 3. Use Case Quản lý Playlist riêng**

1. Đặc tả Use Case Tạo Playlist

|  |  |
| --- | --- |
| **ID** | UC4 |
| **Name** | Tạo playlist |
| **Brief description** | Chức năng cho phép người dùng tạo playlist riêng. |
| **Actors** | Người dùng |
| **Pre-conditions** | Người dùng đã đăng nhập vào hệ thống. |
| **Basic flow** | * Người dùng truy cập giao diện playlist. * Người dùng chọn tạo playlist * Người dùng điền thông tin tên playlist và link ảnh minh họa. * Người dùng ấn tạo playlist. * Hệ thống báo thành công và hiển thị playlist. * Người dùng có thể thêm hoặc xóa bài hát trong playlist. |
| **Alternative flow** | Nếu tạo thất bại thì hệ thống báo lỗi. |
| **Result** | Người dùng tạo playlist thành công. |
| **Exceptions** |  |

1. Đặc tả Use Case Thêm bài hát

|  |  |
| --- | --- |
| **ID** | UC5 |
| **Name** | Thêm bài hát |
| **Brief description** | Chức năng cho phép người dùng thêm bài hát vào playlist riêng của họ. |
| **Actors** | Người dùng |
| **Pre-conditions** | Người dùng đã đăng nhập vào hệ thống.  Người dùng đã tạo playlist. |
| **Basic flow** | * Người dùng truy cập giao diện chi tiết playlist. * Người dùng điền từ khóa vào thanh tìm kiếm. * Hệ thống lấy dữ liệu và hiển thị ra các bài hát phù hợp. * Người dùng ấn vào nút Thêm ở cuối các bài hát. * Hệ thống báo thêm thành công và hiển thị bài hát. |
| **Alternative flow** | Nếu thêm thất bại thì hệ thống báo lỗi. |
| **Result** | Người dùng thêm bài hát thành công. |
| **Exceptions** | Hệ thống gặp lỗi khi lấy dữ liệu để gợi ý bài hát. |

1. Đặc tả Use Case Xóa bài hát

|  |  |
| --- | --- |
| **ID** | UC6 |
| **Name** | Xóa bài hát |
| **Brief description** | Chức năng cho phép người dùng xóa bài hát vào playlist riêng của họ. |
| **Actors** | Người dùng |
| **Pre-conditions** | Người dùng đã đăng nhập vào hệ thống.  Người dùng đã tạo playlist.  Trong playlist đã có bài hát được thêm vào. |
| **Basic flow** | * Người dùng truy cập giao diện chi tiết playlist. * Người dùng ấn vào icon ba chấm ở cuối bài hát. * Hệ thống hiển thị ra nút Xóa. * Người dùng ấn vào nút Xóa. * Hệ thống báo xóa thành công và cập nhật lại danh sách. |
| **Alternative flow** | Nếu xóa thất bại thì hệ thống báo lỗi. |
| **Result** | Người dùng xóa bài hát thành công. |
| **Exceptions** |  |

1. Đặc tả Use Case Xóa Playlist

|  |  |
| --- | --- |
| **ID** | UC7 |
| **Name** | Xóa playlist |
| **Brief description** | Chức năng cho phép người dùng xóa playlist riêng của họ. |
| **Actors** | Người dùng |
| **Pre-conditions** | Người dùng đã đăng nhập vào hệ thống.  Người dùng đã tạo playlist. |
| **Basic flow** | * Người dùng truy cập giao diện chi tiết playlist. * Người dùng ấn vào icon ba chấm ở phần giới thiệu playlist. * Hệ thống hiển thị ra nút Xóa. * Người dùng ấn vào nút Xóa. * Hệ thống báo xóa thành công và quay lại giao diện trước. |
| **Alternative flow** | Nếu xóa thất bại thì hệ thống báo lỗi. |
| **Result** | Người dùng xóa playlist thành công. |
| **Exceptions** |  |

## 2.6. Biểu đồ hoạt động

### 2.6.1. Use Case Đăng nhập

A diagram of a diagram

AI-generated content may be incorrect.

**Hình 4. Activity Diagram của UC Đăng nhập**

### 2.6.2. Use Case Đăng ký

A diagram of a flowchart

AI-generated content may be incorrect.

**Hình 5. Activity Dỉagram của UC Đăng ký**

### 2.6.3. Use Case Phát nhạc

A diagram of a diagram

AI-generated content may be incorrect.

**Hình 6. Activity Diagram của UC Phát nhạc**

### 2.6.4. Use Case Tạo playlist

A diagram of a music playlist

AI-generated content may be incorrect.

**Hình 7. Activity Diagram của UC Tạo playlist**

### 2.6.5. Use Case Thêm bài hát

A diagram of a playlist

AI-generated content may be incorrect.

**Hình 8. Activity Diagram của UC Thêm bài hát**

# Chương 3. Thiết kế hệ thống

## 3.1. Lựa chọn mô hình kiến trúc

Về backend chúng em chọn mô hình MVC.

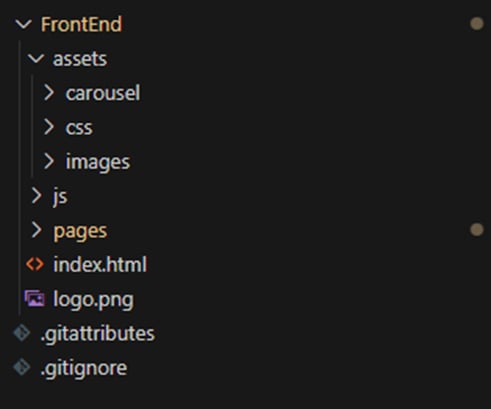
Về frontend thì chỉ là HTML, CSS, JS thuần xây dựng theo dạng trang web SPA.

## 3.2. Sơ đồ kiến trúc tổng thể

A screenshot of a computer

AI-generated content may be incorrect.

Hình 9. Sơ đồ kiến trúc backend



Hình 10. Sơ đồ kiến trúc frontend

## 3.3. Biểu đồ tuần tự

### 3.3.1. Use Case Đăng ký

A diagram of a diagram

AI-generated content may be incorrect.

**Hình 11. Sequence của UC Đăng ký**

### 3.3.2. Use Case Đăng nhập

A diagram of a diagram

AI-generated content may be incorrect.

**Hình 12. Sequence của UC Đăng nhập**

### 3.3.3. Use Case Phát nhạc

A diagram of a diagram of a song control system

AI-generated content may be incorrect.

Hình 13. Sequence của UC Phát nhạc

### 3.3.4. Use Case Tạo Playlist

A diagram of a playlist

AI-generated content may be incorrect.

Hình 14. Sequence của UC Tạo Playlist

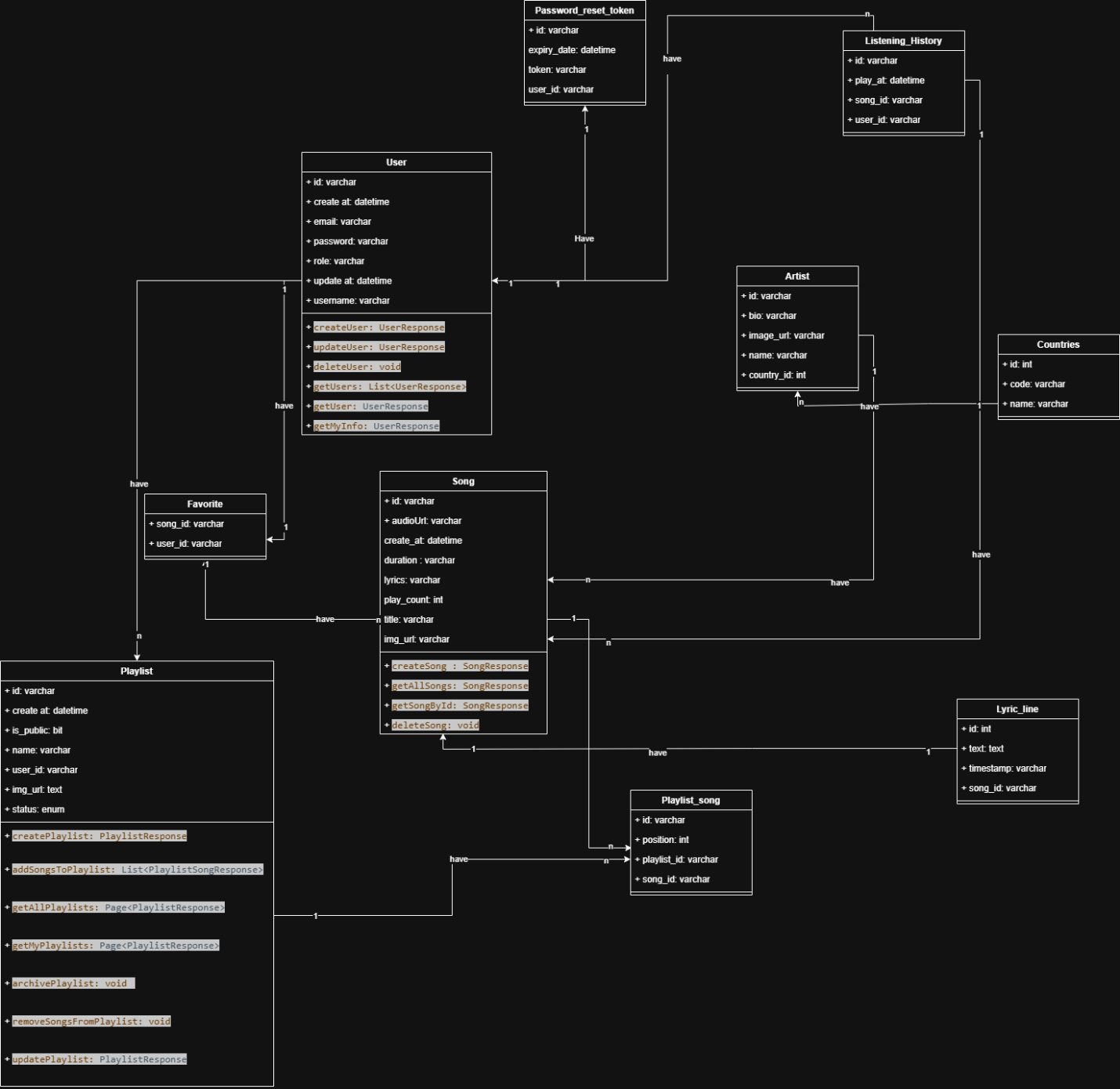
### 3.3.5. Use Case Thêm bài hát

A diagram of a project

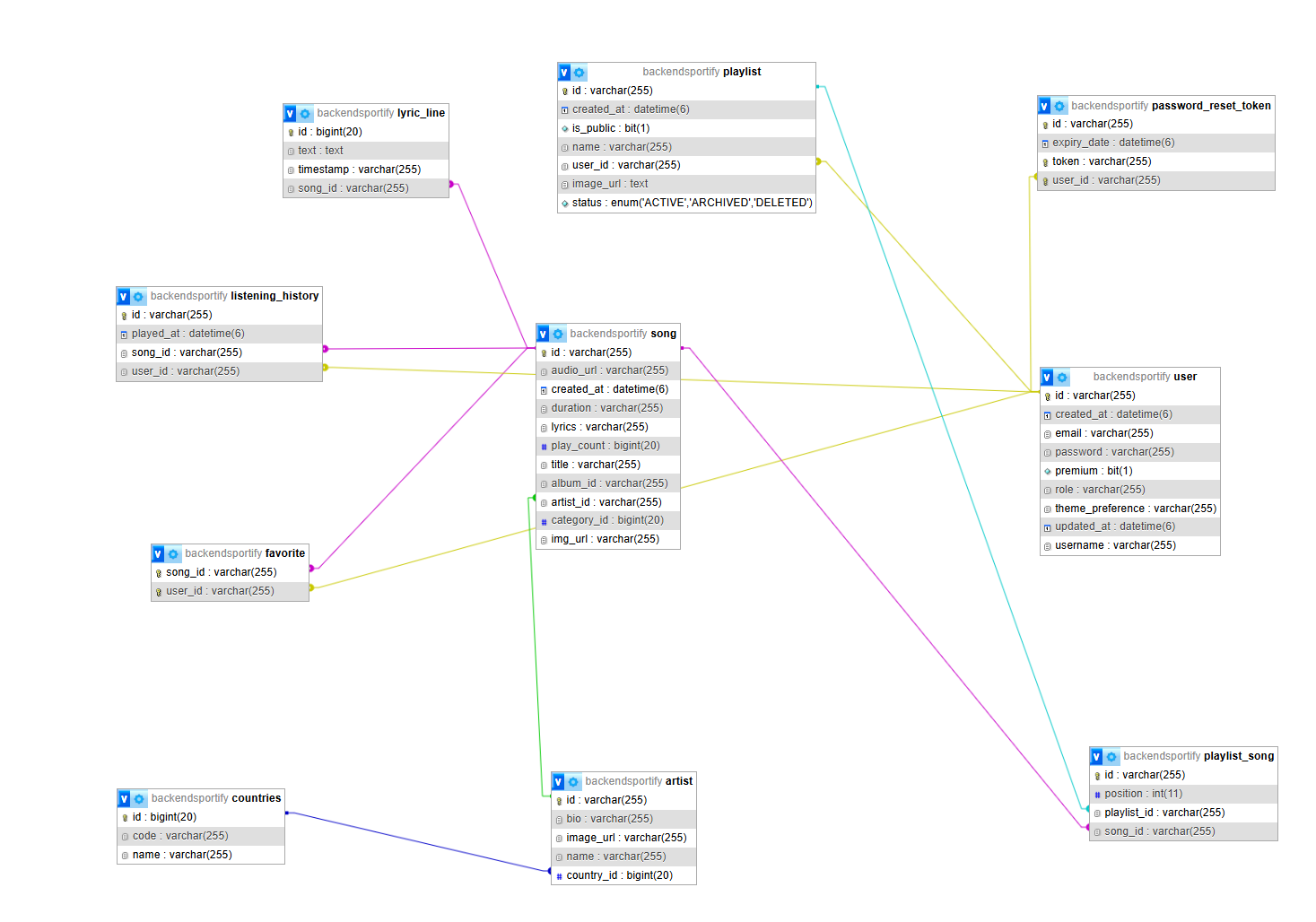
AI-generated content may be incorrect.

Hình 15. Sequence của UC Thêm bài hát

## 3.4. Thiết kế lớp



## 3.5. Thiết kế cơ sở dữ liệu



## 3.6. Thiết kế giao diện người dùng

### 3.6.1. Trang chủ

A screenshot of a computer

AI-generated content may be incorrect.

### 3.6.2. Trang đăng ký

A screenshot of a computer

AI-generated content may be incorrect.

### 3.6.3. Trang đăng nhập

A screenshot of a computer

AI-generated content may be incorrect.

### 3.6.4. Trang quên mật khẩu

A screenshot of a computer

AI-generated content may be incorrect.

### 3.6.5. Trang Favorite

A screenshot of a computer

AI-generated content may be incorrect.

### 3.6.6. Trang Playlist

A screenshot of a computer

AI-generated content may be incorrect.

A screenshot of a computer

AI-generated content may be incorrect.

### 3.6.7. Trang Profile

A screenshot of a computer

AI-generated content may be incorrect.

# Chương 4. Cài đặt và triển khai hệ thống

## Môi trường triển khai

* FrontEnd:
* Sử dụng: HTML, CSS, JavaScript thuần.
* Trình duyệt hỗ trợ: Chrome, Firefox.
* BackEnd:
  + - Ngôn ngữ: Java
    - Database: MySql
    - IDE phát triển: IntelliJ IDEA
    - Server triển khai: Tomcat (nội bộ trong IntelliJ)

## Cài đặt hệ thống

Trên máy cá nhân, cài đặt:

* IntelliJ IDEA (đã tích hợp Maven/Gradle).
* JDK 17+
* MySQL
* Trình duyệt hiện đại: Chrome, Firefox…

Dự án frontend có thể mở file indẽ.html bằng cách chạy từ một local server như:

* Live Server trong VS Code.

## Yêu cầu hệ thống

Phần cứng:

* RAM: ≥ 4 GB
* CPU: Intel i3 hoặc tương đương trở lên
* Ổ cứng trống: ≥ 1 GB

Phần mềm:

* JDK: 17 hoặc cao hơn
* IntelliJ IDEA (phiên bản Community hoặc Ultimate)
* MySQL hoặc hệ quản trị CSDL tương đương
* Trình duyệt hiện đại hỗ trợ JavaScript

## Các bước cài đặt chương trình

Backend (Java):

* Mở IntelliJ, import project.
* Chạy file khởi động.
* Cấu hình CSDL trong application.properties.
* Khởi chạy ứng dụng, backend sẽ chạy ở dạng API.

Frontend (HTML/JS):

* Mở thư mục frontend bằng Live Server.
* Các file JS sẽ gọi các API từ backend thông qua fetch.
* Kiểm tra kết nối với backend.

# Chương 5. Kết quả và đánh giá

## 5.1.Kết quả thử nghiệm hệ thống

Hệ thống đã được triển khai và thử nghiệm trên môi trường cục bộ. Các chức năng chính đều hoạt động ổn định:

* Đăng ký, đăng nhập, và đăng xuất hoạt động đúng yêu cầu.
* Người dùng có thể phát nhạc, xem lời bài hát, tạo playlist và thêm bài hát vào playlist.
* Tìm kiếm bài hát và nghệ sĩ hoạt động chính xác.
* Giao diện hiển thị tốt trên nhiều trình duyệt và thiết bị.

## 5.2.Đánh giá hiệu quả hệ thống

* Giao diện đơn giản, dễ sử dụng và thân thiện với người dùng.
* Hệ thống phản hồi nhanh, phát nhạc mượt, không bị gián đoạn.
* Tổ chức mã nguồn rõ ràng, dễ bảo trì và mở rộng.
* Đảm bảo tính bảo mật cơ bản cho người dùng (quản lý đăng nhập, session/token).