

**BỘ GIÁO DỤC VÀ ĐÀO TẠO  
TRƯỜNG ĐẠI HỌC CẦN THƠ  
TRƯỜNG CÔNG NGHỆ THÔNG TIN & TRUYỀN THÔNG**



**NIÊN LUẬN  
NGÀNH CÔNG NGHỆ THÔNG TIN**

**Đề tài**

**XÂY DỰNG WEBSITE QUẢN LÝ  
VÀ CHƠI NHẠC  
BUILD A WEBSITE TO MANAGE  
AND PLAY MUSIC**

**Sinh viên: Nguyễn Minh Tân  
Mã số: B2014609  
Khóa: K46**

**Cần Thơ, 05/2024**

**BỘ GIÁO DỤC VÀ ĐÀO TẠO  
TRƯỜNG ĐẠI HỌC CẦN THƠ  
TRƯỜNG CÔNG NGHỆ THÔNG TIN & TRUYỀN THÔNG  
KHOA CÔNG NGHỆ THÔNG TIN**



**NIÊN LUẬN  
NGÀNH CÔNG NGHỆ THÔNG TIN**

**Đề tài**

**XÂY DỰNG WEBSITE QUẢN LÝ  
VÀ CHƠI NHẠC  
BUILD A WEBSITE TO MANAGE  
AND PLAY MUSIC**

**Người hướng dẫn  
ThS/TS. Bùi Võ Quốc Bảo**

**Sinh viên thực hiện  
Nguyễn Minh Tân  
Mã số: B2014609  
Khóa: K46**

***Cần Thơ, 05/2024***

## LỜI CẢM ƠN

Chân thành cảm ơn thầy Bùi Võ Quốc Bảo đã hướng dẫn và hỗ trợ em trong quá trình xây dựng website nghe nhạc. Nhờ sự chỉ dẫn tận tình của thầy, em đã học được rất nhiều kiến thức mới và cải thiện được kỹ năng lập trình web của mình. Thầy đã gợi ý chủ đề và hướng dẫn cho tôi một cách chi tiết, cùng đồng hành và hỗ trợ tôi trong suốt quá trình làm niên luận. Em tin rằng những kiến thức và kỹ năng mà tôi đã học được từ thầy, cô và anh sẽ rất hữu ích cho tôi trong công việc và sự nghiệp sau này. Một lần nữa, tôi xin chân thành cảm ơn!

# MỤC LỤC

LỜI CẢM ƠN .....	- 2 -
MỤC LỤC.....	- 3 -
DANH SÁCH BẢNG BIỂU VÀ HÌNH VẼ.....	- 5 -
Danh Sách Bảng Biểu.....	- 5 -
Danh Sách Hình Ảnh.....	- 5 -
TÓM TẮT .....	- 6 -
ABSTRACT .....	- 7 -
GIỚI THIỆU .....	- 8 -
1. Đặt vấn đề.....	- 8 -
2. Mục tiêu đề tài .....	- 9 -
3. Đối tượng và phạm vi nghiên cứu .....	- 9 -
3.1 Đối tượng nghiên cứu .....	- 9 -
3.2 Phạm vi nghiên cứu.....	- 9 -
4. Phương pháp nghiên cứu .....	- 9 -
5. Các công cụ hỗ trợ.....	- 9 -
5.1 Visual studio code .....	- 9 -
5.2 Postman .....	- 9 -
6. Bố cục của quyển niên luận.....	- 10 -
CHƯƠNG I: ĐẶC TẢ YÊU CẦU .....	- 11 -
1. Thư viện supertokens .....	- 11 -
2. Framework React JS.....	- 11 -
3. PostgreSQL 15.....	- 11 -
4. Framework Express JS .....	- 11 -
5. Restful Api.....	- 11 -
6. Mô hình MVC .....	- 12 -
CHƯƠNG II: THIẾT KẾ GIẢI PHÁP .....	- 13 -
1. Cây phân rã chức năng .....	- 13 -
2. Use case Diagrams của các tác nhân .....	- 14 -
2.1 Use case Diagram của tác nhân User.....	- 14 -
2.2 Use case Diagram của tác nhân Admin .....	- 15 -
3. Mô hình dữ liệu .....	- 16 -

3.1 Sơ đồ CDM .....	- 16 -
3.2 Tích hợp supertokens vào mô hình dữ liệu .....	- 16 -
4. Luồng thực thi đăng nhập supertokens.....	- 18 -
5. Quy trình xác minh và làm mới phiên.....	- 19 -
CHƯƠNG III: CÀI ĐẶT GIẢI PHÁP .....	- 20 -
1. Giao diện người dùng .....	- 20 -
1.1 Giao diện user .....	- 20 -
1.2 Giao diện Admin .....	- 23 -
KẾT LUẬN .....	- 28 -
1. Kết quả đạt được.....	- 28 -
2. Hướng phát triển.....	- 28 -
TÀI LIỆU THAM KHẢO.....	- 29 -
PHỤ LỤC.....	- 30 -

# DANH SÁCH BẢNG BIỂU VÀ HÌNH VẼ

## Danh Sách Bảng Biểu

Biểu đồ 1: Cây phân rã chức năng .....	- 13 -
Biểu đồ 2: Use case Diagram của tác nhân User .....	- 14 -
Biểu đồ 3: Use case Diagram của tác nhân Admin.....	- 15 -
Biểu đồ 4: Sơ đồ CDM.....	- 16 -
Biểu đồ 5: Cấu trúc supertokens .....	- 16 -
Biểu đồ 6: luồng thực thi đăng nhập .....	- 18 -

## Danh Sách Hình Ảnh

Hình 1: Trang chủ .....	- 20 -
Hình ảnh 2: Trang playlist.....	- 20 -
Hình ảnh 3: Trang danh sách playlist.....	- 21 -
Hình ảnh 4: Trang danh sách yêu thích.....	- 21 -
Hình ảnh 5: Trang tìm kiếm .....	- 22 -
Hình ảnh 6-7: Trang danh sách yêu thích .....	- 22 -
Hình ảnh 8: Trang danh sách yêu thích.....	- 23 -
Hình ảnh 9: Trang đăng nhập admin.....	- 23 -
Hình ảnh 10: Trang danh sách người dùng .....	- 24 -
Hình ảnh 11: Trang chi tiết người dùng.....	- 24 -
Hình ảnh 12: Trang quản lý và phân quyền .....	- 25 -
Hình ảnh 13: Trang danh sách nhạc .....	- 25 -
Hình ảnh 14: Trang thêm nhạc.....	- 26 -
Hình ảnh 15: Trang sửa nhạc .....	- 26 -
Hình ảnh 16: Trang danh sách thể loại .....	- 27 -
Hình ảnh 17-18: Trang thêm/sửa thể loại .....	- 27 -

## TÓM TẮT

Trang web quản lý và nghe nhạc trực tuyến mang lại sự tiện lợi cho người dùng khi có thể truy cập từ điện thoại di động hoặc máy tính để tìm kiếm, tạo danh sách phát cá nhân và thưởng thức âm nhạc mọi lúc, mọi nơi. Giải pháp này giúp tiết kiệm thời gian và tạo ra trải nghiệm nghe nhạc cá nhân hóa. Sử dụng công nghệ React, Restful API, Node.js và Supertokens giúp tăng cường tính năng và bảo mật cho trang web và dễ dàng nâng cấp.

## **ABSTRACT**

The online music management and streaming platform offers convenience to users by allowing access from mobile devices or computers to search, create personalized playlists, and enjoy music anytime, anywhere. This solution saves time and provides a personalized music listening experience. Leveraging technologies such as React, Restful API, Node.js, and Supertokens enhances features and security for the website, making upgrades easy.



# GIỚI THIỆU

## 1. Đặt vấn đề

Trong thời đại số 4.0, dịch vụ web trở thành phần không thể thiếu đối với các doanh nghiệp. Sự hội nhập quốc tế tạo ra nhu cầu sử dụng dịch vụ trực tuyến, bao gồm việc nghe nhạc. Người dùng càng ngày muốn được trải nghiệm dịch vụ nghe nhạc trên nền tảng web mượt mà và bắt mắt và nâng cấp liên tục. Bên cạnh đó việc bảo mật thông tin người dùng là yếu tố hàng đầu.

Để đáp ứng nhu cầu này, việc áp dụng công nghệ ReactJS là một lựa chọn thông minh. ReactJS cung cấp một cơ sở cho giao diện người dùng linh hoạt và hiệu quả, tạo ra trải nghiệm người dùng mượt mà và hấp dẫn. Đồng thời, việc sử dụng Restful API giúp tương tác linh hoạt giữa client và server, cùng với việc áp dụng Supertokens giúp bảo mật thông tin cá nhân và tài khoản người dùng. Điều này giúp tạo ra một trang web nghe nhạc trực tuyến an toàn, tiện lợi và đáp ứng được mong muốn của người dùng.

## 2. Mục tiêu đề tài

- Xây dựng một website nghe nhạc với các chức năng cơ bản và áp dụng kỹ thuật Single Page Application (SPA) để cải thiện trải nghiệm người dùng. Có thể quản lý, phát nhạc và lưu trữ bài hát playlist...
- Ứng dụng và tìm hiểu thư viện supertokens để quản lý và xác thực người dùng

## 3. Đối tượng và phạm vi nghiên cứu

### 3.1 Đối tượng nghiên cứu

- PostgreSQL, React JS, Express JS, Supertokens, Restful API
- Mô hình MVC
- Cách hoạt động của một web nghe nhạc

### 3.2 Phạm vi nghiên cứu

- Tối ưu hóa trải nghiệm người dùng
- Tăng cường bảo mật và quản lý dữ liệu

## 4. Phương pháp nghiên cứu

- Tìm hiểu về mô hình MVC, Restful API thông qua các dự án trên Github
- Cách ứng dụng supertokens vào dự án thông qua trang chủ supertokens.com
- Tìm hiểu và sử dụng postgresSQL, React JS qua các tài liệu sẵn có trên mạng

## 5. Các công cụ hỗ trợ

### 5.1 Visual studio code

Visual Studio Code là một trình soạn thảo mã nguồn miễn phí và là phần mềm mã nguồn mở được phát triển bởi Microsoft. Nhờ vào kho extension to lớn được đóng góp bởi cộng đồng, nó hỗ trợ debug, syntax highlighting, autocomplete, snippets cho đa số ngôn ngữ cùng với tùy chỉnh theme, phím tắt, và các tùy chọn khác.

### 5.2 Postman

Postman là một ứng dụng desktop dùng để thử nghiệm và phát triển các API. Nó cho phép người dùng tạo và gửi các yêu cầu HTTP RESTful, xem và kiểm tra phản hồi từ server, quản lý biến môi trường, và tự động hóa các bước kiểm thử. Postman hỗ trợ một loạt các phương thức HTTP như GET, POST, PUT, DELETE, PATCH và các chức năng khác như Authentication, Headers, Cookies, và Environment Variables.

## **6. Bố cục của quyển niên luận**

Bố cục quyển niên luận gồm ba chương:

- Chương I: Đặc tả yêu cầu
- Chương II: Thiết kế giải pháp
- Chương III: Cài đặt giải pháp

# CHƯƠNG I: ĐẶC TẢ YÊU CẦU

## 1. Thư viện supertokens

Supertokens là một thư viện mã nguồn mở cung cấp các công cụ và giải pháp để quản lý xác thực và bảo mật trong ứng dụng web. Được thiết kế để giảm thiểu các vấn đề bảo mật phổ biến như tấn công CSRF, XSS và lợi dụng lỗ hổng xác thực, Supertokens cung cấp các tính năng như quản lý phiên đăng nhập, xác thực hai yếu tố, quản lý quyền truy cập, và mã hóa mật khẩu. Đặc biệt, Supertokens tích hợp dễ dàng với các framework phổ biến như ReactJS, Node.js, và Express, giúp cho việc triển khai và quản lý bảo mật trong ứng dụng web trở nên đơn giản và hiệu quả.

## 2. Framework React JS

ReactJS là một framework JavaScript phổ biến được sử dụng để xây dựng giao diện người dùng cho các ứng dụng web. Nó hỗ trợ single-page rendering (SPR), một phương pháp cho phép tải trang web một lần duy nhất và sau đó cập nhật nội dung trang mà không cần tải lại toàn bộ trang. Điều này phù hợp để tạo ra một trang web nghe nhạc tăng trải nghiệm người dùng khi không cần phải reload lại nhạc mỗi khi tương tác.

## 3. PostgreSQL 15

PostgreSQL là một hệ quản trị cơ sở dữ liệu mạnh mẽ và mã nguồn mở, thường được sử dụng để lưu trữ và quản lý dữ liệu trong ứng dụng web và các hệ thống thông tin. Nó cung cấp tính năng bảo mật cao, hỗ trợ ACID (Atomicity, Consistency, Isolation, Durability), và có khả năng mở rộng linh hoạt. PostgreSQL thường được ưa chuộng trong các dự án doanh nghiệp và cộng đồng mã nguồn mở.

## 4. Framework Express JS

Express.js là một framework web cho Node.js, nhằm đơn giản hóa quá trình xây dựng ứng dụng web. Nó cung cấp các tính năng cơ bản như xử lý tuyến đường, middleware, và giao diện chương trình ứng dụng (API).

## 5. Restful Api

RESTful API là một kiểu kiến trúc cho các dịch vụ web, dựa trên các nguyên tắc của REST (Representational State Transfer). Nó cho phép các ứng dụng giao tiếp với nhau thông qua các yêu cầu HTTP, như GET, POST, PUT, DELETE, để thực hiện các thao tác trên tài nguyên. RESTful API được thiết kế để đơn giản, dễ hiểu và linh hoạt, giúp tăng cường khả năng tương tác và tái sử dụng mã nguồn.

## 6. Mô hình MVC

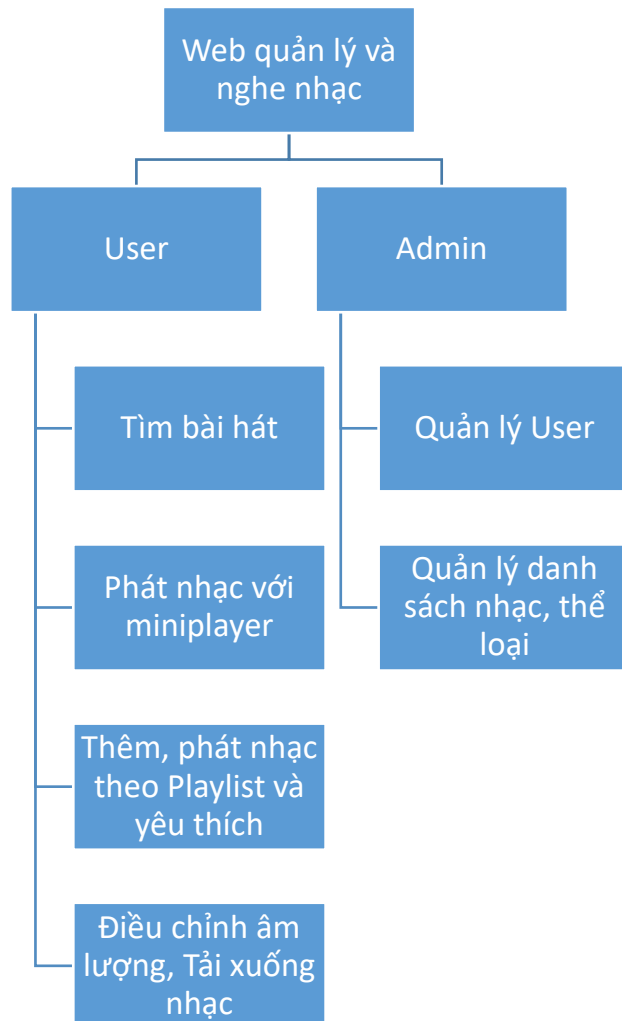
Mô hình MVC là một kiến trúc thiết kế sử dụng trong kỹ thuật phần mềm.

MVC chia một ứng dụng thành ba phần:

- Model (M): là trung tâm của mô hình, quản lý trực tiếp với dữ liệu và cung cấp cho các phần khác.
- View (V): cung cấp giao diện người dùng từ các dữ liệu được cung cấp.
- Controller (C): tiếp nhận yêu cầu từ người dùng, tương tác với dữ liệu thông qua Model và trả về View cho người dùng.

## CHƯƠNG II: THIẾT KẾ GIẢI PHÁP

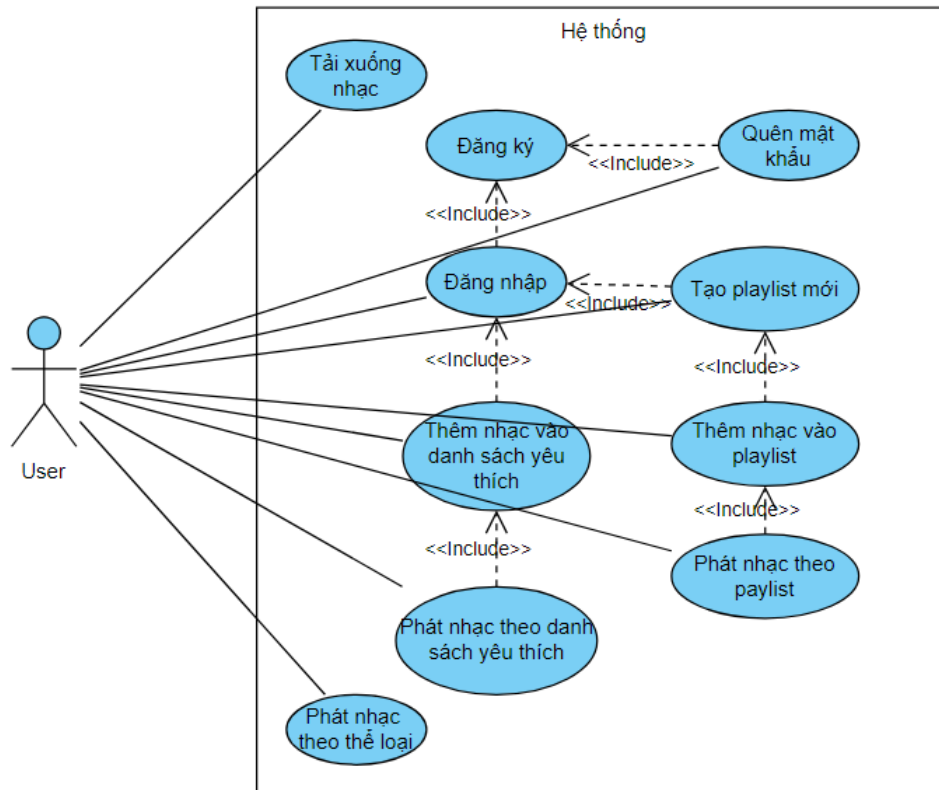
### 1. Cây phân rã chức năng



*Biểu đồ 1: cây phân rã chức năng*

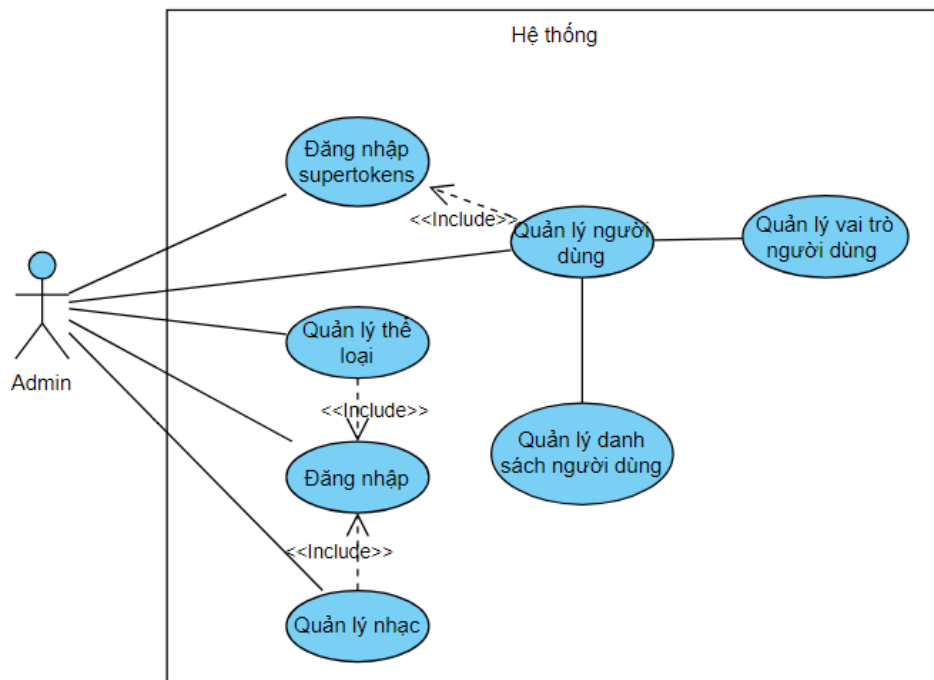
## 2. Use case Diagrams của các tác nhân

### 2.1 Use case Diagram của tác nhân User



Biểu đồ 2: Use case Diagram của tác nhân User

## 2.2 Use case Diagram của tác nhân Admin

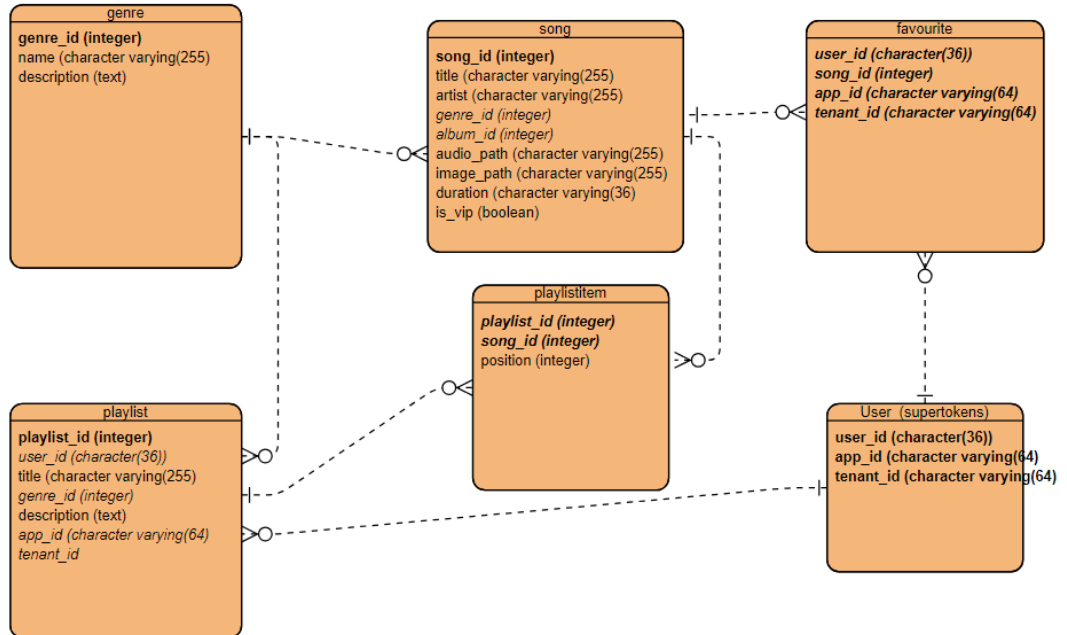


*Biểu đồ 3: Use case Diagram của tác nhân Admin*



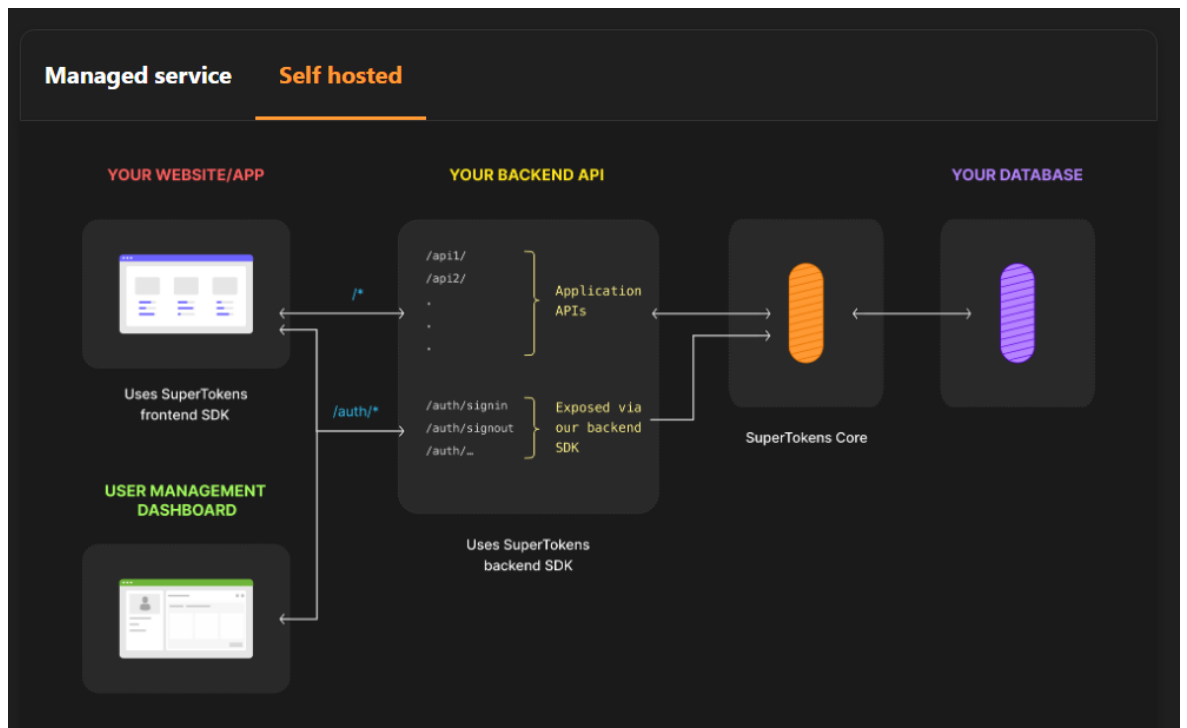
### 3. Mô hình dữ liệu

#### 3.1 Sơ đồ CDM



Biểu đồ 4: Sơ đồ CDM

#### 3.2 Tích hợp supertokens vào mô hình dữ liệu



Biểu đồ 5: Cấu trúc supertokens

Sử dụng cấu trúc tự lưu trữ của supertokens bao gồm 4 phần

### **3.2.1 Fontend layer**

- Tích hợp SDK để giao tiếp với các API xác thực ở backend layer sử dụng FDI gọi là "Giao diện trình điều khiển Frontend"
- Chịu trách nhiệm hiển thị các tiện ích giao diện người dùng đăng nhập và tự động quản lý mã thông báo phiên.

### **3.2.2 Backend layer**

- Backend tích hợp SDK giao tiếp trực tiếp với lõi supertokens thông qua CDI gọi là "Giao diện trình điều khiển lõi"
- Cung cấp các API để đăng ký, đăng nhập, đăng xuất, làm mới phiên...

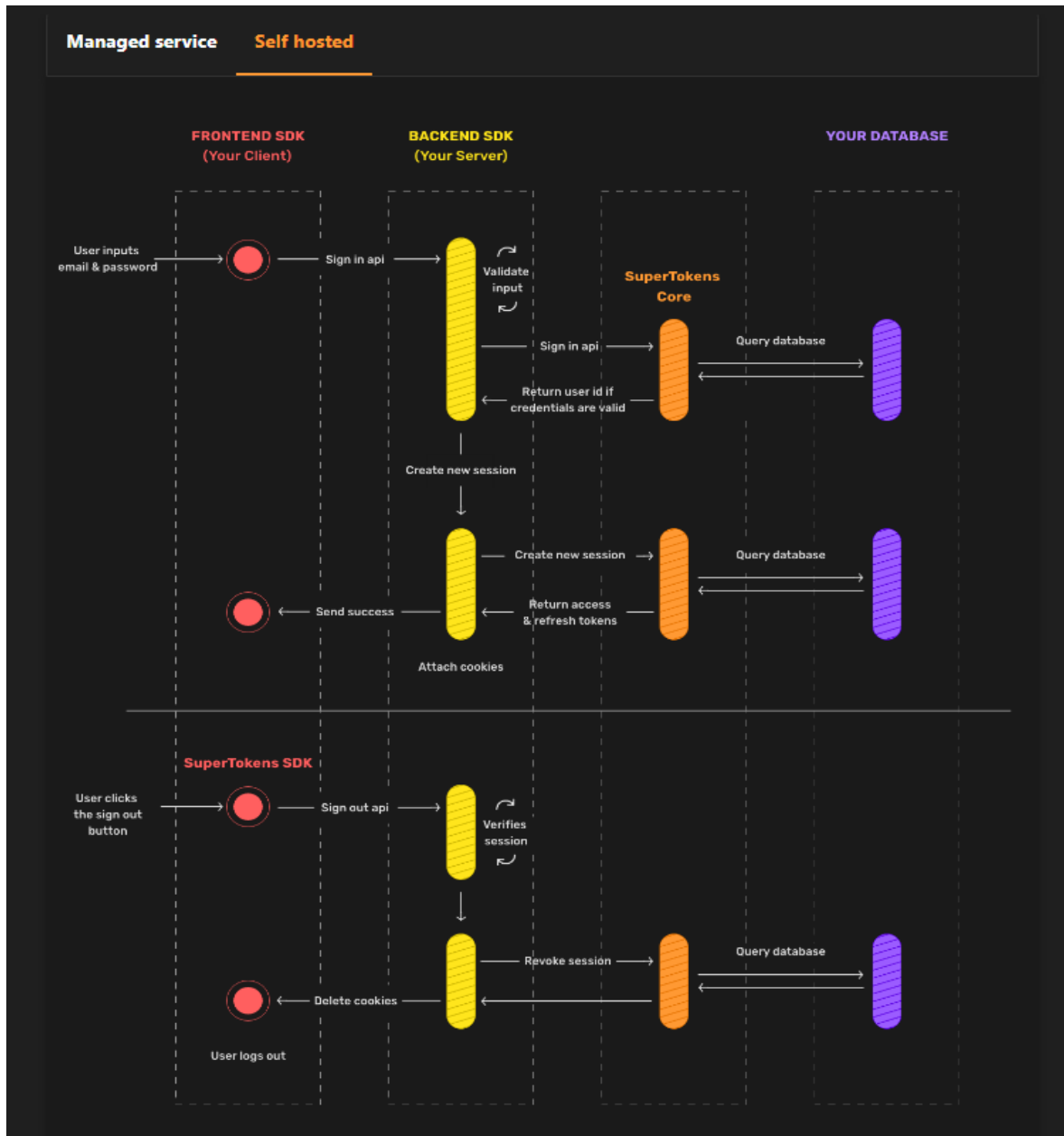
### **3.2.3 Supertokens core**

- Lõi supertokens được nhúng và chạy trên máy chủ cục bộ và kết nối với cơ sở dữ liệu
- Đây là dịch vụ HTTP chứa logic cốt lõi cho auth. Nó chịu trách nhiệm giao tiếp với cơ sở dữ liệu và được SDK phụ trợ truy vấn về các hoạt động yêu cầu db

### **3.2.4 Database**

- Sử dụng postgre SQL supertokens core kết nối đến.
- Tạo ra các bảng dữ liệu do supertokens cung cấp

## 4. Luồng thực thi đăng nhập supertokens



Biểu đồ 6: Luồng thực thi đăng nhập

Luồng thực thi đăng nhập trải qua năm bước:

- 1) Quy trình bắt đầu người dùng gửi yêu cầu đăng nhập từ giao diện do supertokens cung cấp truy vấn tới backend API SDKs
- 2) Backend SDK sẽ xác thực cú pháp đầu vào và gọi lỗi supertokens bằng thông tin xác thực người dùng nhập
- 3) Lỗi supertokens sẽ trả lời lại bằng chuỗi trạng thái OK (cùng với userId) hoặc `WRONG_CREDENTIALS_ERROR`.

- Nếu trạng thái là `WRONG_CREDENTIALS_ERROR` thì backend SDK sẽ gửi đến frontend để có thể hiển thị lỗi ra giao diện
  - Nếu trạng thái là `OK` thì backend SDK sẽ truy vấn vào lỗi supertokens để tạo ra session tokens mới (Access + Refresh Token)
- 4) Sau khi lỗi trả về tokens thì backend SDK sẽ đính kèm và phản hồi dưới dạng cookie hoặc header gửi để cho frontend (mặc định sẽ dưới dạng cookie cho ứng dụng web)
  - 5) Người dùng đăng nhập thành công.

## 5. Quy trình xác minh và làm mới phiên

Quy trình xác minh và làm mới phiên trải qua sáu bước:

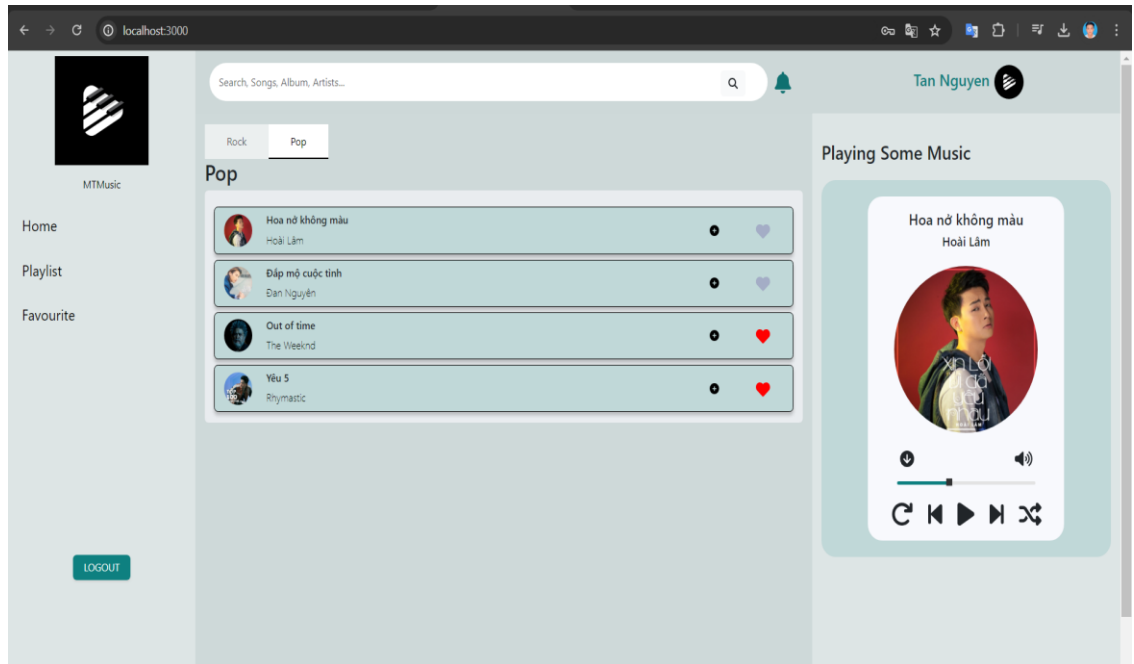
- 1) Sau khi đăng nhập, frontend sẽ có access token tồn tại trong thời gian ngắn và refresh token sẽ có thời gian tồn tại dài hơn (httpOnly Cookies)
- 2) Khi thực hiện gọi một API thì mà access token sẽ được đính kèm theo nhờ frontend SDK
- 3) Phần backend sẽ sử dụng hàm `Verify Session` để kiểm tra tính hợp lệ của phiên
  - Nếu hợp lệ sẽ nhận được một đối tượng session để có thể trích xuất thông tin người dùng và phiên
  - Nếu không hợp lệ, hàm `Verify Session` sẽ gửi mã trạng thái 401 về frontend.
- 4) Khi nhận được mã 401, frontend sẽ gọi `POST API /auth/session/refresh` để làm mới phiên.
- 5) API sẽ gọi lỗi supertokens để tạo access token và refresh token mới sau đó gửi đến frontend
- 6) Frontend sẽ thử lại yêu cầu ban đầu và lần này
- 7) Khi mã access token được làm mới thì xác thực phiên sẽ thành công

# CHƯƠNG III: CÀI ĐẶT GIẢI PHÁP

## 1. Giao diện người dùng

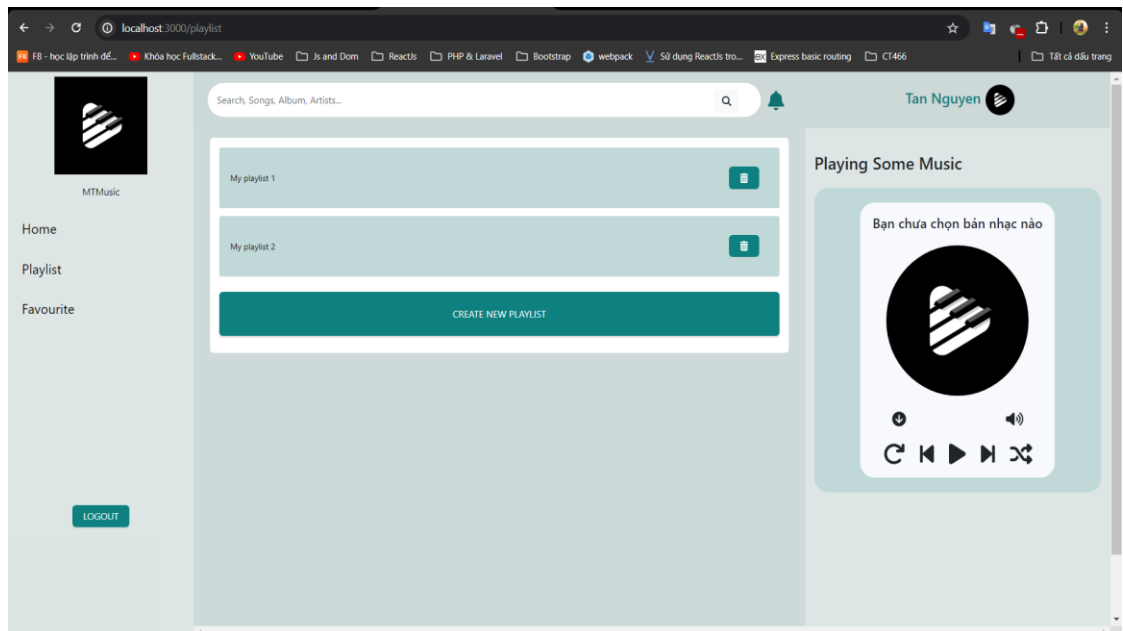
### 1.1 Giao diện user

#### 1.1.1 Trang chủ



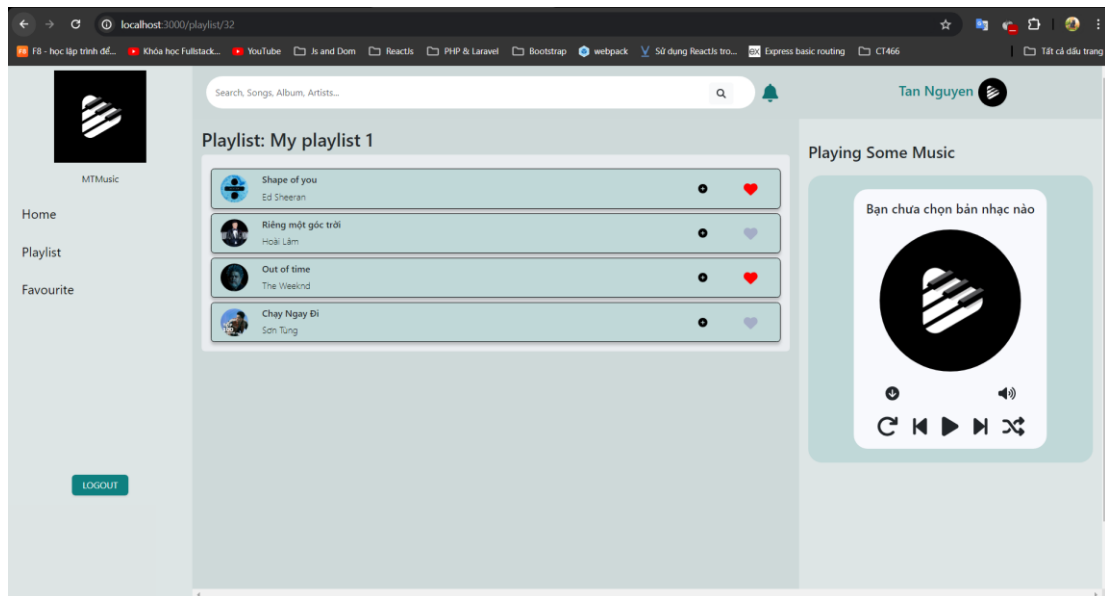
Hình 1: Trang chủ

#### 1.1.2 Trang playlist



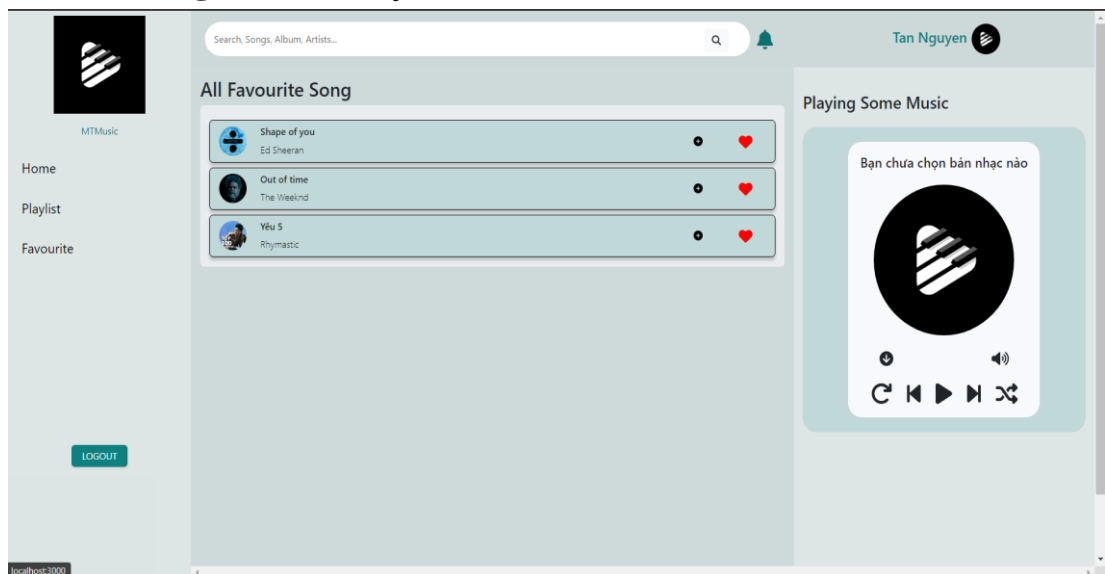
Hình ảnh 2: Trang playlist

### 1.1.3 Trang danh sách playlist



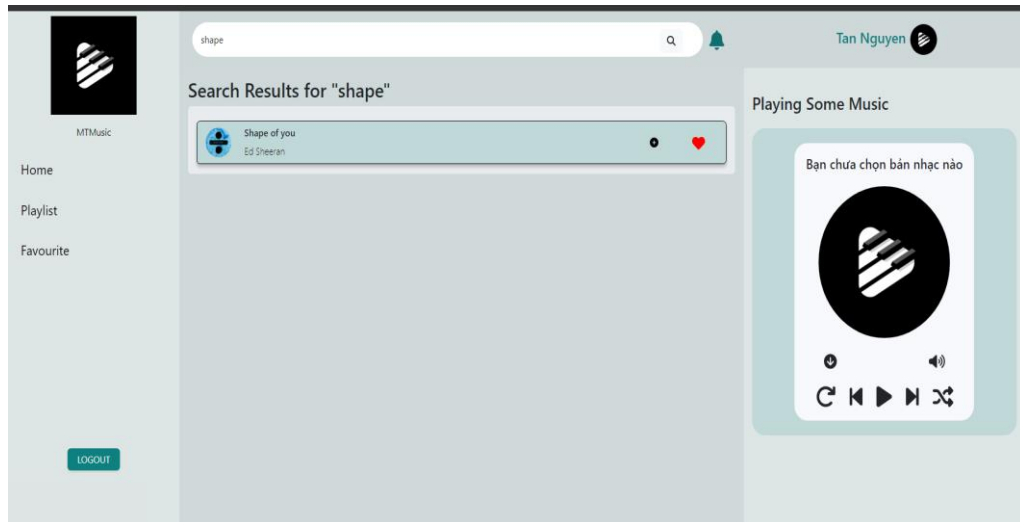
Hình ảnh 3: Trang danh sách playlist

### 1.1.4 Trang danh sách yêu thích



Hình ảnh 4: Trang danh sách yêu thích

### 1.1.5 Trang tìm kiếm




Hình ảnh 5: Trang tìm kiếm

### 1.1.6 Trang đăng ký, đăng nhập

#### Sign Up

Already have an account? [Sign In](#)

 Continue with Google

or

**Email**


**Password**

SIGN UP

Powered by SuperTokens

#### Sign In

Not registered yet? [Sign Up](#)

 Continue with Google

or

**Email**

**Password**

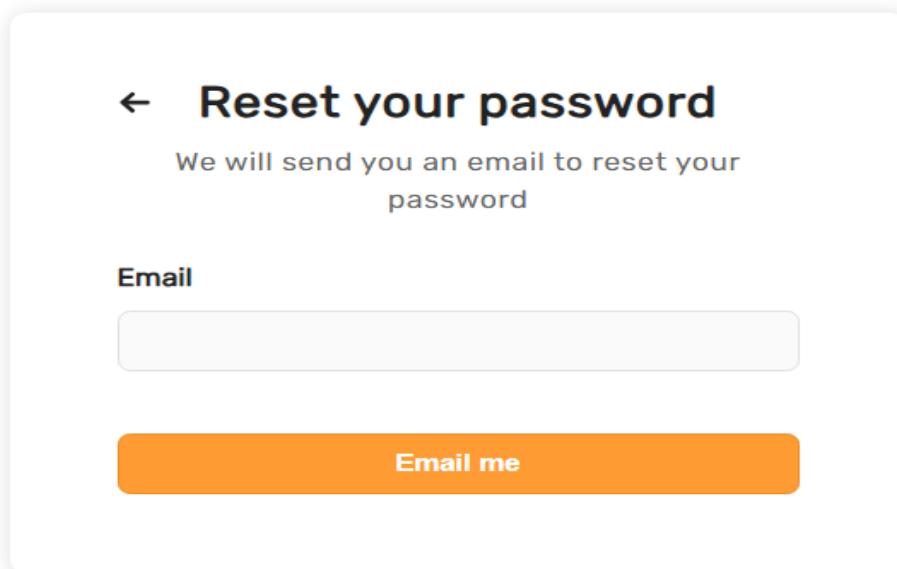
SIGN IN

[Forgot password?](#)

Powered by SuperTokens

Hình ảnh 6-7: Trang danh sách yêu thích

### 1.1.7 Trang quên mật khẩu



← **Reset your password**

We will send you an email to reset your password

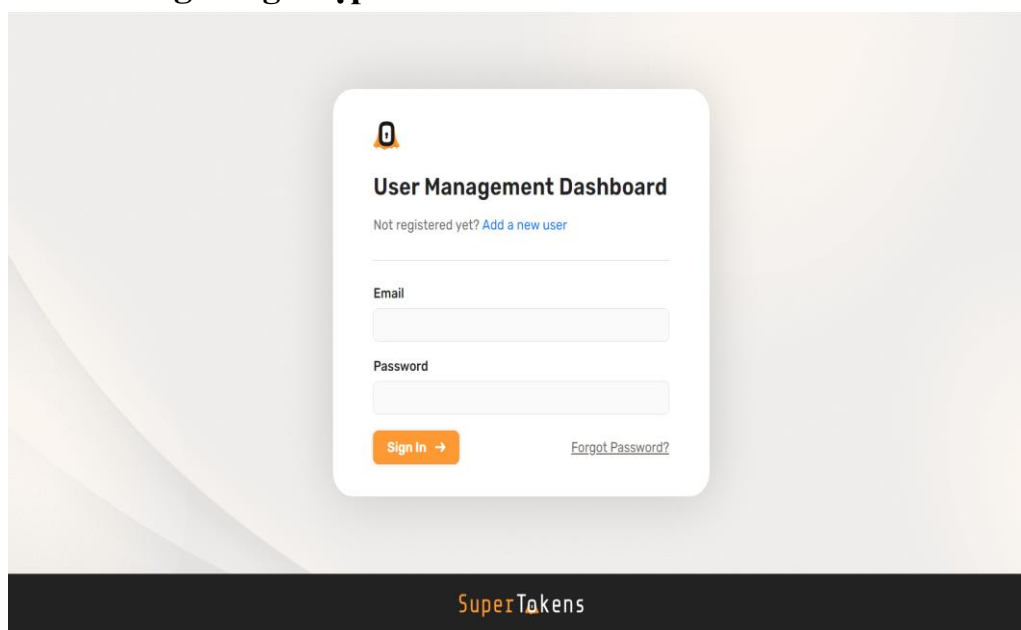
Email


**Email me**

Hình ảnh 8: Trang danh sách yêu thích

## 1.2 Giao diện Admin

### 1.2.1 Trang đăng nhập



 **User Management Dashboard**

Not registered yet? [Add a new user](#)

Email

Password

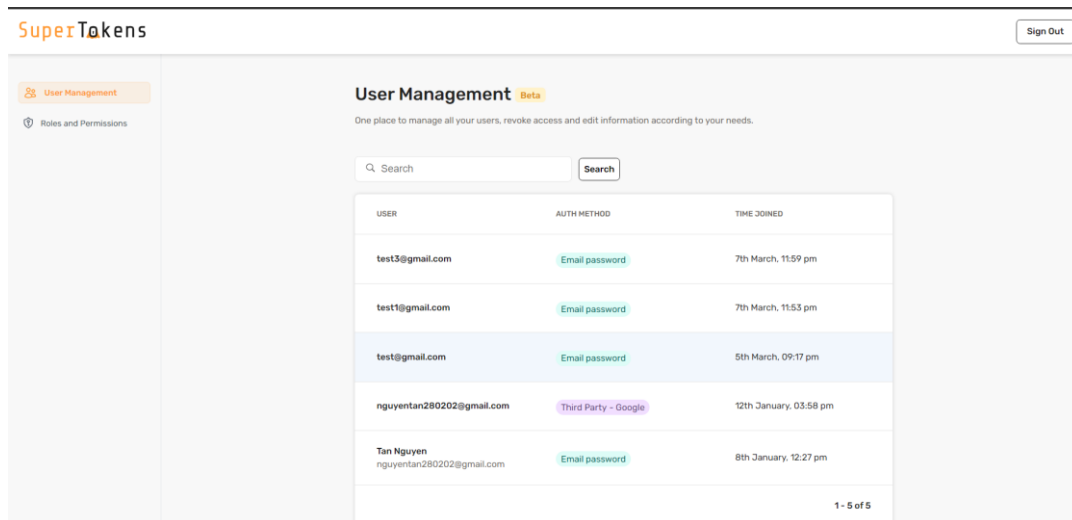
**Sign In →** [Forgot Password?](#)

**SuperTakens**

Hình ảnh 9: Trang đăng nhập admin

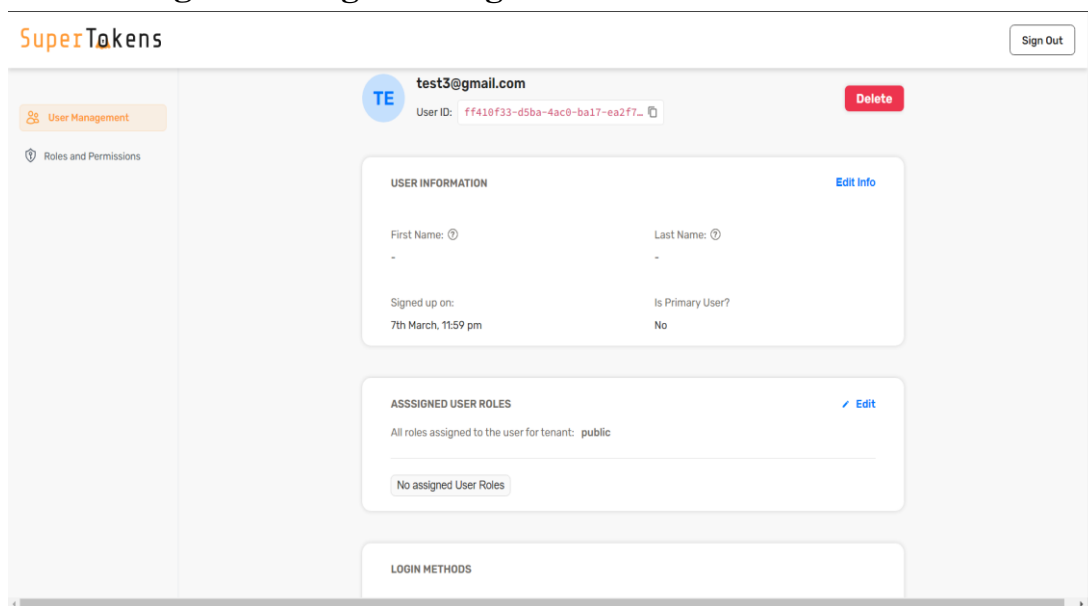


## 1.2.2 Trang danh sách người dùng



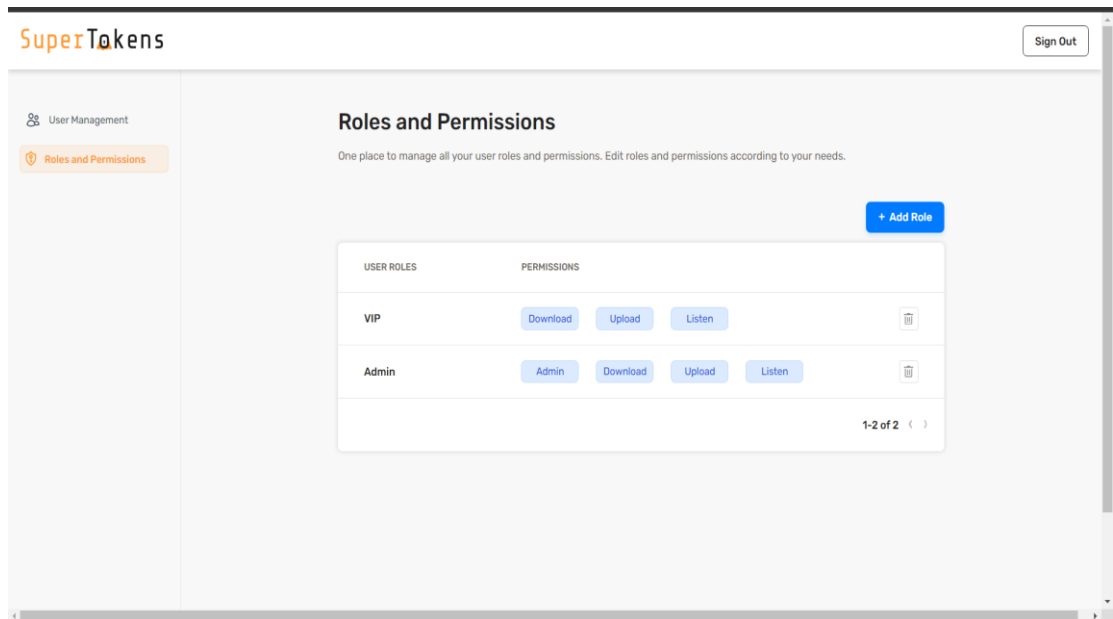
Hình ảnh 10: Trang danh sách người dùng

## 1.2.3 Trang chi tiết người dùng



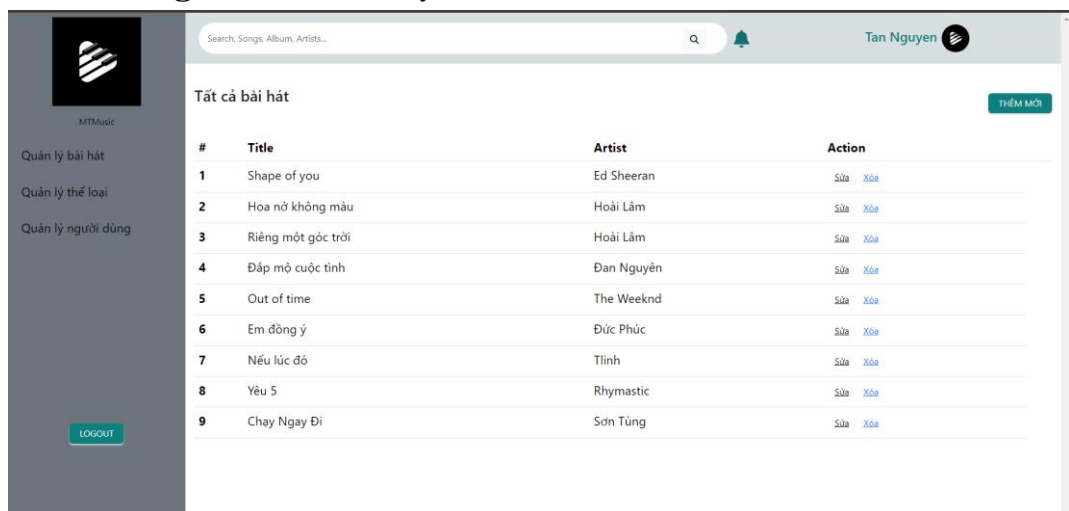
Hình ảnh 11: Trang chi tiết người dùng

## 1.2.4 Trang quản lý và phân quyền người dùng



Hình ảnh 12: Trang quản lý và phân quyền

## 1.2.5 Trang danh sách nhạc



Hình ảnh 13: Trang danh sách nhạc

### 1.2.6 Trang thêm nhạc

MTMusic

Quản lý bài hát

Quản lý thể loại

Quản lý người dùng

LOGOUT

Search, Songs, Album, Artists...

Tan Nguyen

Đăng bài hát

[Danh sách bài hát](#)

Tên bài hát

Đường dẫn hình ảnh bài hát

Chọn tệp | Không có tệp nào được chọn

Đường dẫn bài hát

Chọn tệp | Không có tệp nào được chọn

Tên ca sĩ

Thể loại nhạc

☐ Rock

☒ Pop

Thêm bài hát

Hình ảnh 14: Trang thêm nhạc

### 1.2.7 Trang sửa nhạc

MTMusic

Quản lý bài hát

Quản lý thể loại

Quản lý người dùng

LOGOUT

Sửa thông tin bài hát

Tên bài hát

Shape of you

Đường dẫn hình ảnh bài hát

Chọn tệp | Không có tệp nào được chọn

Đường dẫn bài hát

Chọn tệp | Không có tệp nào được chọn

0:00 / 3:53

Tên ca sĩ

Ed Sheeran

Thể loại nhạc

☒ Rock

☐ Pop

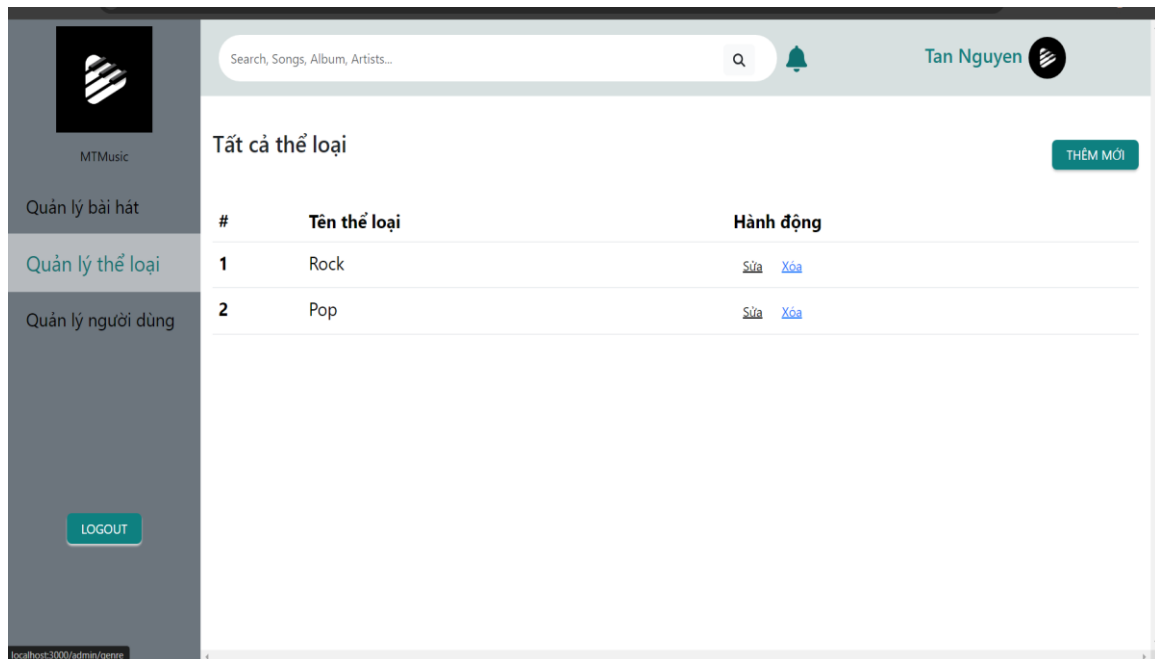
Bài hát VIP

☐

Cập nhật bài hát

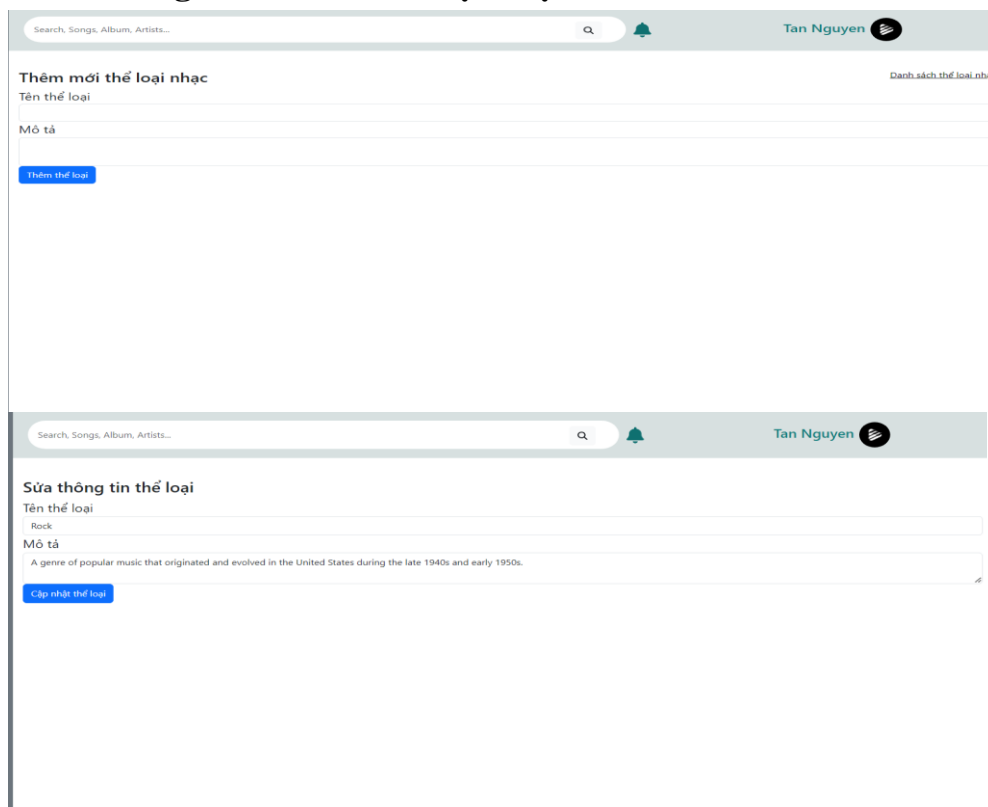
Hình ảnh 15: Trang sửa nhạc

## 1.2.8 Trang danh sách thể loại nhạc



Hình ảnh 16: Trang danh sách thể loại

## 1.2.9 Trang thêm/ sửa thể loại nhạc



Hình ảnh 17-18: Trang thêm/sửa thể loại

# KẾT LUẬN

## 1. Kết quả đạt được

- Đã ứng dụng được các chức năng cơ bản chứng thực người dùng của supertokens vào trang web
- Xây dựng được giao diện User, Admin
- Xây dựng được chức năng cơ bản của web nghe nhạc trực tuyến sử dụng mô hình SPA (Single Page Application)
- Xây dựng được các chức năng quản lý hệ thống cơ bản của Admin

## 2. Hướng phát triển

- Xây dựng thêm các chức năng mới
- Cải thiện thêm giao diện người dùng
- Thêm các điều kiện chặt chẽ

## TÀI LIỆU THAM KHẢO

- Trang chủ supertokens: <https://supertokens.com/>
- Website nghe nhạc Zing mp3: <https://zingmp3.vn/>
- Link github phát triển dự án: <https://github.com/tantahi28/CT466-MTMusic.git>
- Dạy học lập trình: <https://fullstack.edu.vn/>

## **PHỤ LỤC**