

TRƯỜNG ĐẠI HỌC SÀI GÒN
KHOA CÔNG NGHỆ THÔNG TIN



HỌC PHẦN: PHẦN MỀM
MÃ NGUỒN MỞ

Phần mềm Spotify

Nhóm sinh viên thực hiện:

Họ và tên	MSSV
Văn Tuấn Kiệt	3122410202
Mai Phúc Lâm	3122410207
Nguyễn Đức Duy Lâm	3122410208
Nguyễn Hữu Lộc	3122410213
Hồ Hưng Lộc	3122410219
Nguyễn Đình Thông	3122410400

Giáo viên hướng dẫn: ThS. Từ Lãng Phiêu

TP.HCM, 2025

Mục lục

Lời cảm ơn	3
1 GIỚI THIỆU	4
1.1 Giới thiệu sơ lược về đề tài	4
1.2 Lý do chọn đề tài	5
1.3 Mục tiêu	5
1.4 Phân công công việc cho các thành viên	6
2 ĐẶC TẢ YÊU CẦU PHẦN MỀM	7
2.1 Yêu cầu chức năng	7
2.1.1 Người dùng	7
2.1.2 Quản lý	8
2.2 Yêu cầu phi chức năng	8
3 KIẾN TRÚC PHẦN MỀM	10
3.1 Mô hình ứng dụng	10
3.2 Công cụ	13
3.3 Kiến trúc phần mềm	18
4 SƠ ĐỒ ERD	20
4.1 Mô hình Dữ liệu	20
4.1.1 Mô hình Người Dùng (NguoiDung)	20
4.1.2 Mô hình Nghệ Sĩ (NgheSi)	21
4.1.3 Mô hình Album	22
4.1.4 Mô hình Bài Hát (BaiHat)	22
4.1.5 Mô hình Danh Sách Phát (DanhSachPhat)	23
4.1.6 Mô hình Thanh Toán (ThanhToan)	23
5 BÁO CÁO KẾT QUẢ	25
5.1 Chức năng phát nhạc	25

5.2	Chức năng phát video âm nhạc	25
5.3	Chức năng tải video âm nhạc	26
5.4	Chức năng tạo album và đánh dấu bài hát yêu thích	27
5.4.1	Chức năng tạo album	27
5.4.2	Chức năng đánh dấu bài hát yêu thích	30
5.5	Quản trị	30
5.5.1	Quản lý người dùng	30
5.5.2	Quản lý bài hát	33
5.5.3	Quản lý album	38
5.5.4	Quản lý nghệ sĩ	42
5.5.5	Quản lý loại bài hát	47
5.5.6	Thống kê	49
5.6	Tính năng chat tích hợp	50
5.7	Chức năng tìm kiếm	51
5.7.1	Tìm kiếm theo từ khóa	51
5.7.2	Tìm kiếm bài hát dựa trên tệp ghi âm	52
5.8	Tính năng chat tích hợp	53
5.9	Premium	54
5.9.1	Mua Premium thanh toán bằng PayPal	54
5.9.2	Mua Premium thanh toán bằng Zalo pay	56
5.10	Bảng xếp hạng và album hot	56
6	HƯỚNG PHÁT TRIỂN	58
7	MÔI TRƯỜNG CHẠY ỨNG DỤNG VÀ CÁCH CÀI ĐẶT	59
7.1	Môi trường chạy ứng dụng	59
7.2	Cách cài đặt	59

Lời cảm ơn

Trong suốt quá trình học tập môn Phần mềm mã nguồn mở và thực hiện phần mềm “Spotify”, chúng em xin gửi lời cảm ơn chân thành nhất đến tất cả những người đã hỗ trợ và đồng hành cùng chúng em trong suốt chặng đường này.

Trước tiên, chúng em xin bày tỏ lòng biết ơn sâu sắc đến giảng viên Từ Lãng Phiêu, người đã tận tình hướng dẫn chúng em từ những bước đầu tiên cho đến khi phần mềm được hoàn thành. Thầy không chỉ cung cấp những kiến thức quý báu mà còn tạo điều kiện thuận lợi để chúng em nghiên cứu, học hỏi và áp dụng vào thực tế. Sự hướng dẫn tận tâm, những góp ý sắc sảo cùng kinh nghiệm thực tế mà thầy chia sẻ đã giúp chúng em củng cố nền tảng chuyên môn vững chắc, đồng thời vượt qua những khó khăn trong quá trình phát triển phần mềm.

Bên cạnh đó, chúng em cũng xin gửi lời cảm ơn chân thành đến các thành viên trong nhóm, những người đã luôn hỗ trợ, đồng hành và chia sẻ kinh nghiệm trong suốt quá trình thực hiện dự án. Sự hợp tác chặt chẽ, tinh thần làm việc nhóm hiệu quả cùng những trải nghiệm thực tế từ các bạn đã giúp chúng em làm quen với quy trình làm việc chuyên nghiệp, tạo nền tảng quan trọng cho con đường phát triển sau này.

Mặc dù đã nỗ lực hết mình để hoàn thành phần mềm, nhưng do thời gian và kiến thức còn hạn chế, chắc chắn vẫn còn những thiếu sót. Chúng em rất mong nhận được sự góp ý từ quý thầy cô và các bạn để có thể tiếp tục cải thiện và hoàn thiện sản phẩm hơn nữa. Những ý kiến đóng góp của thầy cô sẽ là nguồn động lực to lớn, giúp chúng em rút ra bài học kinh nghiệm quý báu và không ngừng phát triển trong tương lai.

Chúng em xin chân thành cảm ơn!

Chương 1

GIỚI THIỆU

1.1 Giới thiệu sơ lược về đề tài

Spotify là một nền tảng phát nhạc trực tuyến hàng đầu thế giới, cung cấp cho người dùng quyền truy cập vào hàng triệu bài hát, podcast và video từ các nghệ sĩ trên toàn cầu. Ra mắt vào năm 2008, Spotify đã thay đổi cách mọi người tiếp cận và thưởng thức âm nhạc, chuyển từ việc sở hữu bản ghi vật lý sang mô hình phát trực tuyến tiện lợi và hợp pháp.

Spotify hoạt động trên nhiều thiết bị, bao gồm máy tính, điện thoại di động, máy tính bảng, loa thông minh, TV và ô tô, cho phép người dùng nghe nhạc mọi lúc, mọi nơi. Nền tảng này cung cấp cả phiên bản miễn phí với quảng cáo và phiên bản trả phí (Spotify Premium) với nhiều tính năng nâng cao như nghe nhạc offline, chất lượng âm thanh cao hơn và không có quảng cáo. Spotify cũng nổi tiếng với khả năng cá nhân hóa trải nghiệm người dùng thông qua các playlist được tạo tự động dựa trên thói quen nghe nhạc, như "Discover Weekly" và "Daily Mix".



1.2 Lý do chọn đề tài

Trước khi Spotify xuất hiện, ngành công nghiệp âm nhạc đối mặt với nhiều thách thức, đặc biệt là vấn đề vi phạm bản quyền do việc chia sẻ nhạc trái phép trên các nền tảng như Napster. Daniel Ek và Martin Lorentzon, những người sáng lập Spotify, nhận thấy cần thiết phải tạo ra một dịch vụ âm nhạc trực tuyến hợp pháp, cung cấp trải nghiệm nghe nhạc chất lượng cao và thuận tiện, đồng thời đảm bảo quyền lợi cho các nghệ sĩ và nhà sản xuất. Mục tiêu của họ là cung cấp một giải pháp thay thế hấp dẫn hơn so với việc tải nhạc bất hợp pháp, bằng cách mang đến cho người dùng quyền truy cập tức thì vào một thư viện âm nhạc khổng lồ với chất lượng cao.

1.3 Mục tiêu

Mục tiêu chính của Spotify là "democratize audio"—dân chủ hóa việc tiếp cận âm thanh. Điều này có nghĩa là cung cấp cho người dùng trên toàn thế giới quyền truy cập dễ dàng và hợp pháp vào kho nội dung âm thanh phong phú, đồng thời tạo cơ hội cho các nghệ sĩ, dù lớn hay nhỏ, tiếp cận với khán giả toàn cầu mà không cần thông qua các kênh phân phối truyền thống. Spotify cũng đặt mục tiêu không ngừng cải thiện trải nghiệm người dùng thông qua việc ứng dụng công nghệ tiên tiến như trí tuệ nhân tạo để cá nhân hóa nội dung và đề xuất âm nhạc phù hợp với sở thích của từng cá nhân.

Ngoài ra, Spotify còn hướng đến việc mở rộng hệ sinh thái âm thanh của

mình bằng cách tích hợp podcast và audiobook, biến nền tảng này thành điểm đến toàn diện cho mọi nhu cầu nghe của người dùng. Điều này không chỉ tăng cường giá trị cho người dùng mà còn tạo thêm nguồn thu nhập cho các nhà sáng tạo nội dung.

Tóm lại, Spotify được phát triển với mục tiêu cung cấp một giải pháp nghe nhạc trực tuyến hợp pháp, chất lượng cao và thuận tiện, đồng thời hỗ trợ các nghệ sĩ tiếp cận khán giả rộng rãi hơn. Nền tảng này không ngừng đổi mới và mở rộng để đáp ứng nhu cầu ngày càng cao của người dùng và thị trường âm nhạc toàn cầu.

1.4 Phân công công việc cho các thành viên

Thành viên	Công việc đã làm
Văn Tuấn Kiệt	Thiết kế yêu cầu chức năng, API bảng xếp hạng, tích hợp thanh toán ZaloPay/Visa, quản lý bài hát/thẻ loại, chức năng Premium, sửa chức năng phát nhạc, tổng hợp đóng góp, mô tả ERD.
Nguyễn Đức Duy Lâm	Tạo source Frontend, API danh sách phát và bài hát, tích hợp danh sách phát, chức năng phát nhạc, phát video, fix lỗi, viết README.
Hồ Hưng Lộc	Viết báo cáo (bìa, mục lục, lời cảm ơn, giới thiệu đề tài, yêu cầu chức năng/phi chức năng, công nghệ), vẽ ERD, API bài hát và tìm kiếm, giao diện admin, tạo album/danh sách phát theo cảm xúc.
Nguyễn Hữu Lộc	Thiết kế database, API nghệ sĩ, API thanh toán, tích hợp Frontend nghệ sĩ/album/thanh toán, chức năng thống kê, cải tiến quản lý bài hát.
Mai Phúc Lâm	Tạo source Backend, code model, API đăng nhập/đăng ký/đăng xuất/quên mật khẩu, tích hợp đăng nhập/tìm kiếm vào Frontend, chat real-time bằng WebSocket.
Nguyễn Đình Thông	Quản lý thể loại bài hát (BE), tích hợp FE và BE Danh sách phát, quản lý album (BE), quản lý người dùng (Xem danh sách người dùng, khóa/mở khóa tài khoản, thêm quyền admin), giao diện đăng ký premium, tìm bài hát từ tệp ghi âm (BE - FE).

Chương 2

ĐẶC TẢ YÊU CẦU PHÂN MỀM

2.1 Yêu cầu chức năng

2.1.1 Người dùng

- **Đăng ký và đăng nhập:** Người dùng có thể tạo tài khoản và đăng nhập vào hệ thống. Người dùng có thể đặt lại mật khẩu nếu quên mật khẩu.
- **Tìm kiếm bài hát:** Người dùng có thể tìm kiếm bài hát bằng tên bài hát, tên nghệ sĩ và có thể tìm kiếm album bằng tên bài hát, tên nghệ sĩ.
- **Phát nhạc:** Người dùng có thể phát bài hát, album, và danh sách phát yêu thích.
- **Danh sách phát:** Người dùng có thể thêm bài hát vào danh sách phát cá nhân của mình hay tải xuống.
- **Quản lý thư viện cá nhân:** Người dùng có thể xem và chỉnh sửa danh sách bài hát, bài hát yêu thích.
- **Quản lý thông tin cá nhân:** Người dùng có thể xem thông tin cá nhân của mình.
- **Premium:** Nếu người dùng chưa đăng ký tài khoản premium thì sẽ phải nghe các quảng cáo sau mỗi 2 bài hát, đồng thời không được nghe các bài hát vip.

- **Thanh toán:** Người dùng có thể thanh toán cho các gói dịch vụ cao cấp qua các phương thức thanh toán trực tuyến bao gồm ZaloPay, Paypal.
- **Tìm bài hát dựa trên giai điệu:** Người dùng có thể tìm bài hát thông qua việc ghi âm một đoạn nhạc nào đó, hệ thống sẽ hiển thị bài hát tương ứng nếu tìm ra.

2.1.2 Quản lý

- **Quản lý người dùng:** Quản lý có thể xem danh sách người dùng, thêm, cập nhật, và tìm kiếm người dùng theo tên, email, hoặc số điện thoại.
- **Quản lý danh sách phát chung:** Quản lý có thể tạo và duy trì các danh sách phát chung cho người dùng.
- **Thông kê:** Quản lý có thể thống kê số lượng người dùng, lượt nghe, và các bài hát phổ biến theo thời gian.
- **Quản lý album:** Quản lý có thể xem và quản lý thông tin các album.
- **Quản lý nghệ sĩ:** Quản lý có thể xem và quản lý thông tin các nghệ sĩ.

2.2 Yêu cầu phi chức năng

Hiệu suất

- **Tìm kiếm nhanh chóng:** Các thao tác tìm kiếm phải có thời gian phản hồi dưới 5 giây.
- **Xử lý giao dịch đồng thời:** Hệ thống phải có khả năng xử lý 500 giao dịch đồng thời mà không gặp phải độ trễ.

Tính bảo mật

- **Mã hóa SSL cho thanh toán:** Đảm bảo các giao dịch tài chính được bảo vệ thông qua SSL/TLS.
- **Xác thực hai yếu tố (2FA):** Sử dụng xác thực OTP hoặc sinh trắc học để tăng cường bảo mật khi người dùng thực hiện giao dịch nhạy cảm.

Tính sẵn có

- **Hoạt động 24/7:** Hệ thống cần phải luôn sẵn sàng với mức độ uptime 99.9

Khả năng sử dụng

- **Giao diện thân thiện:** Giao diện người dùng phải đơn giản và dễ sử dụng, với các tính năng quan trọng hiển thị rõ ràng và dễ tìm.
- **Hỗ trợ đa ngôn ngữ:** Hệ thống cần cung cấp hỗ trợ cho nhiều ngôn ngữ và đảm bảo khả năng truy cập cho người dùng có nhu cầu đặc biệt.

Tính tương thích

- **Tương thích với các trình duyệt:** Ứng dụng cần phải tương thích với các trình duyệt web phổ biến như Chrome, Cốc cốc, và Edge.
- **Tương thích với thiết bị di động:** Giao diện cần phải responsive và tương thích với các hệ điều hành di động (iOS, Android).

Tính bảo trì

- **Dễ bảo trì và mở rộng tính năng:** Cần áp dụng kiến trúc modular để dễ dàng thêm hoặc thay đổi chức năng mà không ảnh hưởng đến toàn bộ hệ thống.
- **CI/CD:** Sử dụng quy trình CI/CD để tự động triển khai và cập nhật hệ thống mà không cần downtime.

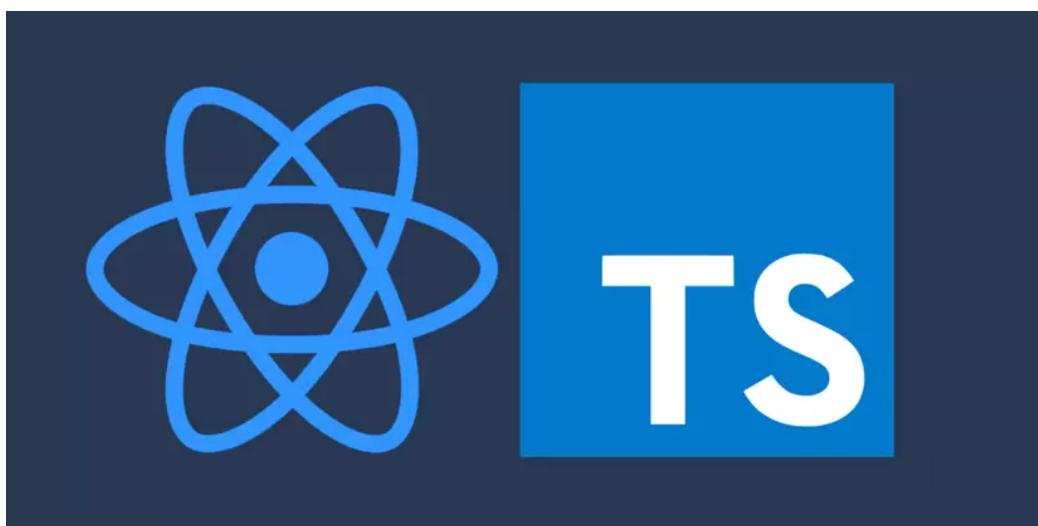
Chương 3

KIẾN TRÚC PHẦN MỀM

3.1 Mô hình ứng dụng

Frontend:

- **ReactJS:** React là một thư viện JavaScript mã nguồn mở, được phát triển bởi Facebook vào năm 2013, nhằm xây dựng giao diện người dùng cho các ứng dụng web. React cho phép các lập trình viên phát triển các thành phần giao diện (components) một cách hiệu quả, dễ bảo trì và tái sử dụng. Với React, người dùng có thể trải nghiệm giao diện mượt mà và dễ dàng tương tác với ứng dụng.



Backend:

- **Python - Django:** Django là một framework Python mạnh mẽ giúp phát triển các ứng dụng web nhanh chóng và dễ dàng hơn. Django cung

cấp các tính năng như ORM (Object Relational Mapping) để quản lý cơ sở dữ liệu, giúp việc phát triển các API RESTful trở nên đơn giản và thuận tiện. Nó cũng hỗ trợ bảo mật và quản lý người dùng, giúp dễ dàng xử lý các yêu cầu API của người dùng.



Hệ quản trị cơ sở dữ liệu:

- **MySQL:** là một hệ thống quản trị cơ sở dữ liệu mã nguồn mở (Relational Database Management System, viết tắt là RDBMS), thuộc quyền sở hữu của Oracle, được sử dụng để quản lý và lưu trữ dữ liệu. Nó sử dụng SQL (Structured Query Language) làm ngôn ngữ chính để truy vấn và thao tác với cơ sở dữ liệu. MySQL phổ biến trong các ứng dụng web, đặc biệt là các ứng dụng sử dụng kiến trúc LAMP (Linux, Apache, MySQL, PHP/Python/Perl). Các ứng dụng web lớn nhất như Facebook, Twitter, YouTube, Google, và Yahoo! đều dùng MySQL cho mục đích lưu trữ dữ liệu. Nó đã tương thích với nhiều hệ tầng máy tính quan trọng như Linux, macOS, Microsoft Windows, và Ubuntu.



Giao thức truyền thông:

- **WebSocket:** là một giao thức truyền thông giúp cho việc thiết lập kênh truyền thông hai chiều giữa máy chủ và máy khách. WebSocket hoạt động bằng cách thiết lập kết nối HTTP liên tục với máy chủ và sau đó nâng cấp nó lên kết nối websocket hai chiều bằng cách gửi Upgrade header. WebSocket được hỗ trợ trong hầu hết các trình duyệt web hiện đại và cho các trình duyệt không hỗ trợ, chúng tôi có các thư viện cung cấp dự phòng cho các kỹ thuật khác như Comet và HTTP Long Polling.



3.2 Công cụ

PyCharm:

- PyCharm là một môi trường phát triển tích hợp (Integrated Development Environment - IDE) được thiết kế chuyên biệt cho lập trình Python, do JetBrains phát triển và ra mắt lần đầu tiên vào tháng 2 năm 2010. PyCharm cung cấp các tính năng mạnh mẽ như gợi ý mã thông minh, kiểm tra lỗi thời gian thực, tích hợp công cụ quản lý phiên bản (Git, SVN), và hỗ trợ phát triển web với các framework như Django và Flask. IDE này có hai phiên bản: Community (miễn phí, mã nguồn mở) và Professional (trả phí, với nhiều tính năng nâng cao). PyCharm được sử dụng rộng rãi trong các dự án phát triển phần mềm, đặc biệt là các ứng dụng Python, và tương thích với nhiều nền tảng như Windows, macOS, và Linux.



Visual Studio Code (VS Code):

- Visual Studio Code là một trình soạn thảo mã nguồn nhẹ nhưng mạnh mẽ, hỗ trợ đa nền tảng như Windows, macOS và Linux. Với các tính năng như hoàn thành mã thông minh, tích hợp Git, và hỗ trợ nhiều tiện ích mở rộng, VS Code được sử dụng để phát triển frontend của dự án Instagram clone với ReactJS. Công cụ này giúp lập trình viên dễ dàng viết mã, gõ lỗi, và quản lý các thư viện như Ant Design, Redux, và TailwindCSS một cách hiệu quả.

Postman:

- Postman là một công cụ phổ biến được sử dụng để kiểm thử và làm việc với các API, đặc biệt là API kiểu REST. API đóng vai trò quan trọng trong việc kết nối các thành phần của ứng dụng, và Postman giúp đơn giản hóa quá trình gọi và kiểm tra API mà không cần viết mã. Công cụ này hỗ trợ tất cả các phương thức HTTP như GET, POST, PUT, PATCH, DELETE, và cho phép lưu lại lịch sử các yêu cầu (request) để sử dụng lại khi cần. Trong dự án Instagram clone, Postman được sử dụng để kiểm tra các API như đăng bài post (POST /api/post), lấy danh sách bài post (GET /api/post), và gửi tin nhắn.



MySQL Workbench:

- MySQL Workbench là một công cụ quản lý cơ sở dữ liệu MySQL, cung cấp giao diện đồ họa để phát triển, thiết kế và quản lý cơ sở dữ liệu một cách trực quan. Công cụ này cho phép dễ dàng tạo, chỉnh sửa cơ sở dữ liệu, thực hiện các thao tác như đảo ngược (reverse engineering) để tạo mô hình từ cơ sở dữ liệu hiện có, và chuyển tiếp (forward engineering) để triển khai mô hình thành cơ sở dữ liệu. Trong dự án Instagram clone, MySQL Workbench được sử dụng để thiết kế và quản lý cơ sở dữ liệu MySQL, bao gồm các bảng như users, posts, comments, và messages.

GitHub:

- GitHub là một nền tảng quản lý mã nguồn phổ biến, cho phép các lập trình viên chia sẻ, cộng tác và quản lý phiên bản mã nguồn một cách hiệu quả. Sự phát triển của GitHub bắt đầu vào ngày 19 tháng 10 năm 2007, và trang web chính thức được ra mắt vào tháng 4 năm 2008

bởi **Tom Preston-Werner**, **Chris Wanstrath**, và **PJ Hyett**. Microsoft đã mua lại GitHub vào tháng 6 năm 2018, giúp nền tảng này có thêm nhiều nguồn lực và tích hợp sâu hơn với các sản phẩm của Microsoft, như Azure và Visual Studio.

- GitHub cung cấp không chỉ một kho lưu trữ mã nguồn mà còn là một công cụ mạnh mẽ để cộng tác và chia sẻ mã nguồn giữa các lập trình viên trên toàn cầu. Git, hệ thống quản lý phiên bản mà GitHub sử dụng, cho phép các lập trình viên theo dõi và kiểm soát sự thay đổi trong mã nguồn theo thời gian, giúp họ có thể quay lại các phiên bản trước nếu cần thiết và làm việc đồng thời mà không gặp phải vấn đề xung đột dữ liệu.
- GitHub cung cấp các tính năng như **branches** (nhánh) để làm việc song song trên các tính năng hoặc sửa lỗi mà không làm ảnh hưởng đến mã nguồn chính. Người dùng có thể **fork** (tạo bản sao) các dự án từ người khác để làm việc trên đó và sau đó tạo **pull request** (yêu cầu hợp nhất) để đóng góp các thay đổi của mình vào dự án gốc.
- GitHub cũng cung cấp các công cụ hỗ trợ như **issue tracking** (theo dõi vấn đề), **project boards** (bảng dự án), **actions** (tự động hóa quy trình phát triển), và **wikis** để tài liệu hóa các dự án, giúp các nhóm lập trình viên dễ dàng hợp tác, theo dõi tiến độ, và quản lý các vấn đề phát sinh trong suốt quá trình phát triển phần mềm.
- Ngoài ra, GitHub còn có tính năng **GitHub Pages**, cho phép người dùng lưu trữ các trang web tĩnh trực tiếp từ kho GitHub của mình. GitHub còn hỗ trợ **private repositories** (kho lưu trữ riêng tư), giúp các lập trình viên hoặc tổ chức bảo vệ mã nguồn của họ khỏi việc chia sẻ công khai.
- Nói chung, GitHub là một công cụ cực kỳ quan trọng trong cộng đồng lập trình viên, giúp tăng cường sự hợp tác, bảo vệ mã nguồn và thúc đẩy quy trình phát triển phần mềm nhanh chóng và hiệu quả.

Giấy phép mã nguồn mở (Open Source License):

- Giấy phép Mã nguồn mở (Open Source License) là các giấy phép cho phép người dùng tự do truy cập, sử dụng, sửa đổi và phân phối phần mềm, miễn là họ tuân theo các điều kiện của giấy phép. Các giấy phép này giúp phần mềm được phát triển và chia sẻ rộng rãi, thúc đẩy cộng đồng đóng góp vào việc phát triển phần mềm. Một số giấy phép mã nguồn mở phổ biến bao gồm: MIT License, GPL (General Public

License), và Apache License 2.0. MIT License là một trong những giấy phép mã nguồn mở đơn giản nhất, cho phép người dùng sử dụng, sao chép, sửa đổi và phân phối phần mềm miễn phí, nhưng yêu cầu phải bao gồm bản quyền và thông báo giấy phép. GPL yêu cầu phần mềm phát hành dưới mã nguồn mở và bất kỳ phần mềm phát triển từ phần mềm này cũng phải tiếp tục là mã nguồn mở. Apache License 2.0 tương đối thoải mái nhưng yêu cầu bảo vệ các quyền sở hữu trí tuệ liên quan đến phần mềm. Các giấy phép mã nguồn mở thúc đẩy sự phát triển của phần mềm thông qua cộng đồng và giúp bảo vệ quyền lợi của các tác giả phần mềm.

Windows Subsystem for Linux (WSL):

- Windows Subsystem for Linux (WSL) là một tính năng trên Windows 10 và Windows Server 2019, cho phép người dùng chạy môi trường Linux trực tiếp trên Windows mà không cần phải cài đặt một máy ảo hoặc hệ điều hành dual-boot. WSL giúp các nhà phát triển sử dụng các công cụ và phần mềm Linux, ngay cả khi họ đang làm việc trên hệ điều hành Windows. WSL cung cấp một trải nghiệm người dùng rất tiện lợi vì nó cho phép sử dụng các công cụ dòng lệnh của Linux, như Bash, grep, sed, awk, và nhiều phần mềm khác mà thường chỉ có sẵn trên Linux. Điều này rất hữu ích đối với các nhà phát triển web và phần mềm, những người cần phải phát triển trên môi trường Linux mà không muốn chuyển sang hệ điều hành khác. WSL có hai phiên bản: WSL 1 và WSL 2. WSL 1 tạo ra một lớp tương thích giữa Windows và Linux, cho phép chạy các lệnh Linux trực tiếp trên Windows mà không cần máy ảo. WSL 2 cải thiện hiệu suất với một nhân Linux thực sự, giúp hỗ trợ đầy đủ các hệ thống file của Linux và tăng tốc độ đáng kể, đặc biệt là đối với các ứng dụng yêu cầu hệ thống file hoặc nhân Linux đầy đủ.

Amazon Web Services (AWS):

- Amazon Web Services (AWS) là nền tảng dịch vụ đám mây của Amazon, cung cấp các dịch vụ như tính toán, lưu trữ, cơ sở dữ liệu, phân tích dữ liệu và các công cụ phát triển khác. AWS giúp các công ty và lập trình viên xây dựng và vận hành các ứng dụng mà không cần phải quản lý cơ sở hạ tầng vật lý. AWS cung cấp các dịch vụ đám mây quy mô lớn, đáng tin cậy, bảo mật và chi phí hiệu quả, phục vụ nhiều loại ứng dụng từ ứng dụng web đến ứng dụng di động, từ cơ sở hạ tầng IT cho đến AI và học máy. Một số dịch vụ chính của AWS bao gồm Amazon EC2 (Elastic Compute Cloud), Amazon S3 (Simple Storage

Service), AWS Lambda, và Amazon RDS (Relational Database Service). Amazon EC2 cung cấp các máy ảo để bạn chạy ứng dụng, server và dịch vụ của mình. Amazon S3 là dịch vụ lưu trữ dữ liệu đám mây có khả năng mở rộng cao, giúp bạn lưu trữ và truy xuất dữ liệu một cách dễ dàng. AWS Lambda là dịch vụ tính toán không máy chủ, cho phép bạn chạy mã mà không cần phải quản lý máy chủ. Amazon RDS cung cấp các dịch vụ cơ sở dữ liệu như MySQL, PostgreSQL, và Oracle trên đám mây, giúp việc quản lý cơ sở dữ liệu trở nên dễ dàng và tiết kiệm chi phí. AWS có mô hình thanh toán theo mức sử dụng, giúp các doanh nghiệp chỉ trả tiền cho những dịch vụ mà họ sử dụng, từ đó tối ưu hóa chi phí và tài nguyên cho các công ty và tổ chức.



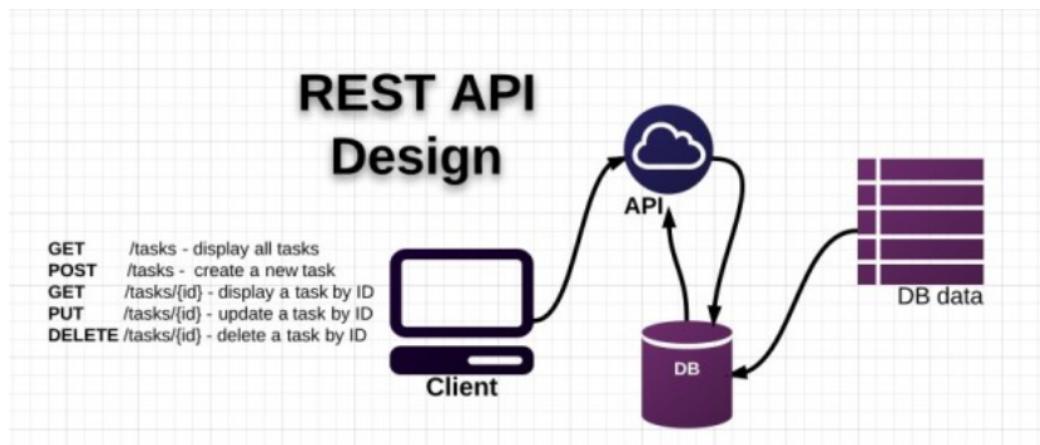
Deploy: Amazon EC2 và VPS:

- **Amazon EC2:** Amazon EC2 là dịch vụ đám mây của Amazon Web Services (AWS), cho phép người dùng thuê máy chủ ảo (instances) để triển khai và chạy ứng dụng. EC2 mang lại sự linh hoạt, khả năng mở rộng và tính sẵn sàng cao cho các ứng dụng web. Với Clone Spotify, EC2 sẽ được sử dụng để triển khai phần backend và giúp ứng dụng có thể mở rộng khi lượng người dùng tăng.
- **VPS:** VPS là một máy chủ ảo được phân chia từ một máy chủ vật lý duy nhất. Mỗi VPS có hệ điều hành riêng biệt, giúp triển khai ứng dụng và quản lý tài nguyên như một máy chủ độc lập. Với Clone Spotify, VPS sẽ được sử dụng cho việc triển khai các dịch vụ khác ngoài backend, như các ứng dụng phụ trợ hoặc lưu trữ dữ liệu không đụng đến hệ thống chính.

3.3 Kiến trúc phần mềm

RESTful API:

- **Khái niệm:** RESTful API là một tiêu chuẩn được sử dụng trong việc thiết kế API cho các ứng dụng web (thiết kế Web services) nhằm mục đích quản lý các tài nguyên (resource) một cách hiệu quả. Nó tập trung vào các tài nguyên hệ thống như tệp văn bản, hình ảnh, âm thanh, video, hoặc dữ liệu động, bao gồm các trạng thái tài nguyên được định dạng và truyền tải thông qua giao thức HTTP. RESTful API cho phép các ứng dụng giao tiếp với nhau thông qua các phương thức HTTP chuẩn, giúp đơn giản hóa việc truy cập và xử lý dữ liệu trên các nền tảng khác nhau.



Điễn giải các thành phần

- **API (Application Programming Interface):** API là một tập hợp các quy tắc và cơ chế cho phép một ứng dụng hoặc thành phần tương tác với một ứng dụng hoặc thành phần khác. API đóng vai trò như một cầu nối, giúp các ứng dụng trao đổi dữ liệu với nhau một cách dễ dàng. Dữ liệu trả về từ API thường được định dạng ở các kiểu phổ biến như JSON hoặc XML, phù hợp để tích hợp vào các ứng dụng web hoặc di động.
- **REST (REpresentational State Transfer):** REST là một kiểu kiến trúc (architectural style) được sử dụng để thiết kế API, dựa trên việc chuyển đổi trạng thái biểu diễn của tài nguyên. REST tận dụng các phương thức HTTP đơn giản như GET, POST, PUT, DELETE để thực hiện các thao tác trên tài nguyên. Thay vì sử dụng một URL để xử lý thông tin phức tạp, REST gửi các yêu cầu HTTP đến một URL

cụ thể nhằm truy xuất hoặc chỉnh sửa dữ liệu, giúp giao tiếp giữa các hệ thống trở nên trực quan và hiệu quả.

- **RESTful API:** RESTful API là một tiêu chuẩn thiết kế API cho các ứng dụng web, tập trung vào việc quản lý các tài nguyên (resource) và cho phép các ứng dụng khác nhau (web, mobile, v.v.) giao tiếp với nhau. Chức năng quan trọng nhất của RESTful API là quy định cách sử dụng các phương thức HTTP (như GET để lấy dữ liệu, POST để tạo mới, PUT để cập nhật, DELETE để xóa) và cách định dạng URL để truy cập các tài nguyên. RESTful API không quy định logic mã nguồn của ứng dụng và không bị giới hạn bởi ngôn ngữ lập trình, do đó bất kỳ ngôn ngữ hoặc framework nào (như Java với Spring Boot, JavaScript với Node.js) cũng có thể được sử dụng để thiết kế một RESTful API.

Chương 4

SƠ ĐỒ ERD

4.1 Mô hình Dữ liệu

Dưới đây là các mô hình dữ liệu chính trong hệ thống:

4.1.1 Mô hình Người Dùng (NguoiDung)

Mô hình NguoiDung lưu trữ thông tin về người dùng của hệ thống. Các trường chính của entity này bao gồm:

- **nguo_i_dung_id**: Khóa chính, định danh duy nhất cho mỗi người dùng.
 - **email**: Địa chỉ email của người dùng, được sử dụng để đăng nhập.
 - **so_dien_thoai**: Số điện thoại của người dùng.

- **ten_hien_thi**: Tên hiển thị của người dùng trên hệ thống.
- **gioi_tinh**: Giới tính của người dùng, có thể là "Nam" hoặc "Nữ".
- **avatar_url**: URL của ảnh đại diện người dùng.
- **ngay_sinh**: Ngày sinh của người dùng.
- **quoc_gia**: Quốc gia của người dùng.
- **la_premium**: Trường boolean xác định xem người dùng có phải là người dùng Premium không.
- **google_id** và **facebook_id**: Các trường chứa ID của người dùng từ Google hoặc Facebook (nếu đăng nhập qua các nền tảng này).
- **ngay_tao** và **ngay_cap_nhat**: Thời gian tạo và cập nhật thông tin người dùng.
- **is_active** và **is_staff**: Các trường quản lý trạng thái hoạt động và quyền hạn của người dùng (staff).

Chức năng của mô hình này là quản lý thông tin người dùng, cung cấp cơ sở dữ liệu cho việc đăng nhập, tạo và cập nhật thông tin người dùng, cũng như phân quyền cho người dùng.

4.1.2 Mô hình Nghệ Sĩ (NgheSi)

Mô hình NgheSi lưu trữ thông tin về nghệ sĩ. Các trường chính bao gồm:

- **nghe_si_id**: Khóa chính, định danh nghệ sĩ.
- **ten_nghe_si**: Tên nghệ sĩ.
- **tieu_su**: Thông tin mô tả về nghệ sĩ.
- **anh_dai_dien**: URL của ảnh đại diện nghệ sĩ.
- **ngay_sinh**: Ngày sinh của nghệ sĩ.
- **quoc_gia**: Quốc gia của nghệ sĩ.
- **is_active**: Trạng thái hoạt động của nghệ sĩ.
- **created_at** và **updated_at**: Thời gian tạo và cập nhật thông tin nghệ sĩ.

Chức năng của mô hình này là lưu trữ và quản lý thông tin của các nghệ sĩ, giúp người dùng có thể tìm kiếm và theo dõi các nghệ sĩ yêu thích.

4.1.3 Mô hình Album

Mô hình **Album** lưu trữ thông tin về các album nhạc. Các trường chính bao gồm:

- **album_id**: Khóa chính, định danh album.
- **ten_album**: Tên album.
- **nghe_si**: Khóa ngoại liên kết đến mô hình **NgheSi**, chỉ ra nghệ sĩ sở hữu album.
- **anh_bia**: URL của ảnh bìa album.
- **ngay_phat_hanh**: Ngày phát hành album.
- **the_loai**: Thể loại của album.
- **is_active**: Trạng thái hoạt động của album.
- **created_at** và **updated_at**: Thời gian tạo và cập nhật album.

Chức năng của mô hình này là quản lý thông tin về các album mà nghệ sĩ phát hành, giúp người dùng dễ dàng tìm kiếm và nghe nhạc theo album.

4.1.4 Mô hình Bài Hát (BaiHat)

Mô hình **BaiHat** lưu trữ thông tin về các bài hát trong hệ thống. Các trường chính bao gồm:

- **bai_hat_id**: Khóa chính, định danh bài hát.
- **ten_bai_hat**: Tên bài hát.
- **nghe_si**: Khóa ngoại liên kết đến mô hình **NgheSi**, chỉ ra nghệ sĩ thể hiện bài hát.
- **album**: Khóa ngoại liên kết đến mô hình **Album**, chỉ ra album mà bài hát thuộc về.
- **the_loai**: Thể loại của bài hát.
- **file_bai_hat**: Đường dẫn đến tệp bài hát (mp3, mp4, v.v.).
- **duong_dan**: URL của bài hát, giúp người dùng phát bài hát từ hệ thống.

- **loi_bai_hat**: Lời bài hát.
- **thoi_luong**: Thời gian bài hát (tính bằng giây).
- **ngay_phat_hanh**: Ngày phát hành bài hát.

Chức năng của mô hình này là lưu trữ thông tin bài hát, bao gồm các tệp âm thanh và lời bài hát, giúp người dùng nghe và tìm kiếm bài hát.

4.1.5 Mô hình Danh Sách Phát (DanhSachPhat)

Mô hình DanhSachPhat lưu trữ thông tin về các danh sách phát nhạc của người dùng. Các trường chính bao gồm:

- **danh_sach_phat_id**: Khóa chính, định danh danh sách phát.
- **nguo_dung_id**: Khóa ngoại liên kết đến mô hình NguoiDung, chỉ ra người dùng sở hữu danh sách phát.
- **ten_danh_sach**: Tên danh sách phát.
- **mo_ta**: Mô tả về danh sách phát.
- **la Cong_khai**: Trạng thái công khai của danh sách phát.
- **ngay_tao**: Thời gian tạo danh sách phát.
- **tong_thoi_luong**: Tổng thời gian của tất cả bài hát trong danh sách phát (tính bằng giây).
- **so_thu_tu**: Thứ tự hiển thị của danh sách phát.
- **anh_danh_sach**: URL của ảnh đại diện cho danh sách phát.
- **so_nguo_theo_doi**: Số người theo dõi danh sách phát.

Chức năng của mô hình này là quản lý các danh sách phát của người dùng, giúp người dùng tạo và chia sẻ các danh sách nhạc.

4.1.6 Mô hình Thanh Toán (ThanhToan)

Mô hình ThanhToan lưu trữ thông tin về các giao dịch thanh toán của người dùng đối với các gói Premium. Các trường chính bao gồm:

- **thanh_toan_id**: Khóa chính, định danh giao dịch thanh toán.

- **nguoit dung id**: Khóa ngoại liên kết đến mô hình NguoiDung, chỉ ra người dùng thực hiện giao dịch.
- **so_tien**: Số tiền đã thanh toán.
- **loai_thanh_toan**: Loại giao dịch (ví dụ: thẻ tín dụng, PayPal).
- **ngay_thanh_toan**: Ngày thanh toán.
- **trang_thai_thanh_toan**: Trạng thái của giao dịch (thành công, thất bại).

Chức năng của mô hình này là quản lý thông tin giao dịch thanh toán của người dùng để cấp quyền Premium cho người dùng.

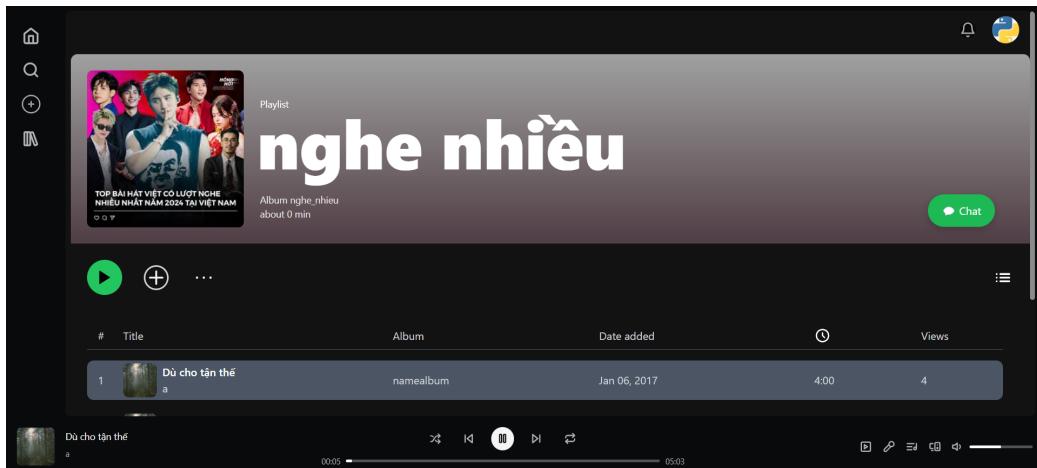
Chương 5

BÁO CÁO KẾT QUẢ

Trong quá trình phát triển phần mềm **Spotify Clone**, nhóm đã hiện thực các tính năng chính như sau:

5.1 Chức năng phát nhạc

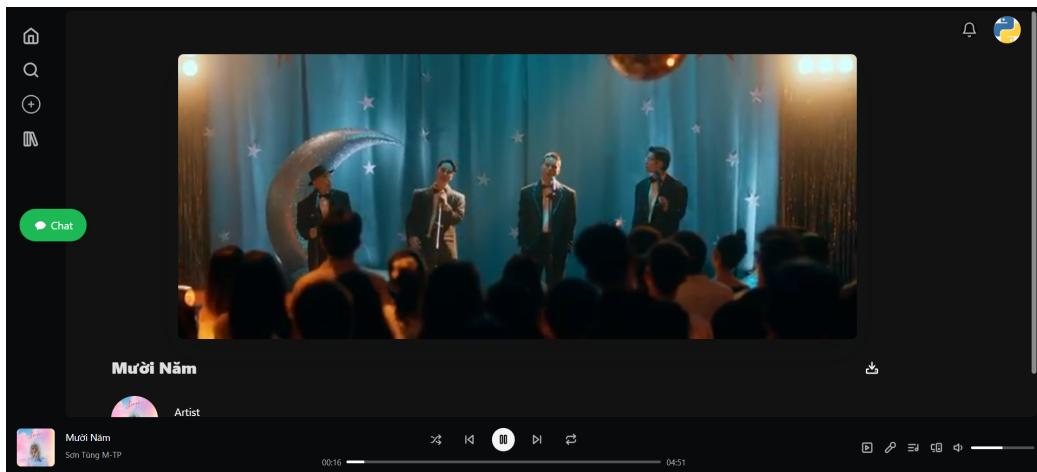
Người dùng có thể chọn và phát các bài hát từ danh sách bài hát có sẵn. Tính năng này hỗ trợ phát nhạc liên tục, phát ngẫu nhiên và lặp lại bài hát.



Hình 5.1: Giao diện chức năng phát nhạc

5.2 Chức năng phát video âm nhạc

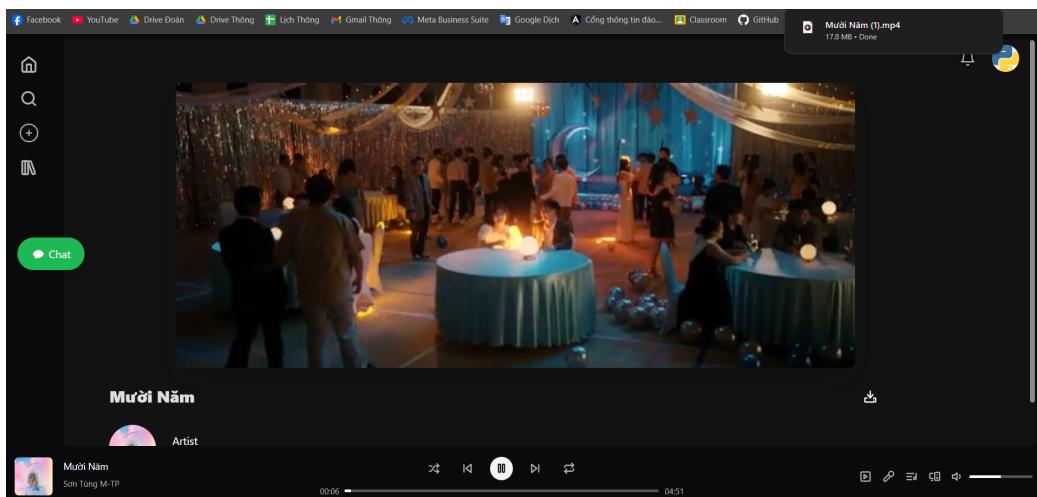
Người dùng có thể xem các video âm nhạc và có điều khiển phát lại.



Hình 5.2: Giao diện chức năng phát video âm nhạc

5.3 Chức năng tải video âm nhạc

Hệ thống cho phép người dùng tải video âm nhạc về thiết bị cá nhân thông qua nút tải xuống trong giao diện phát video.

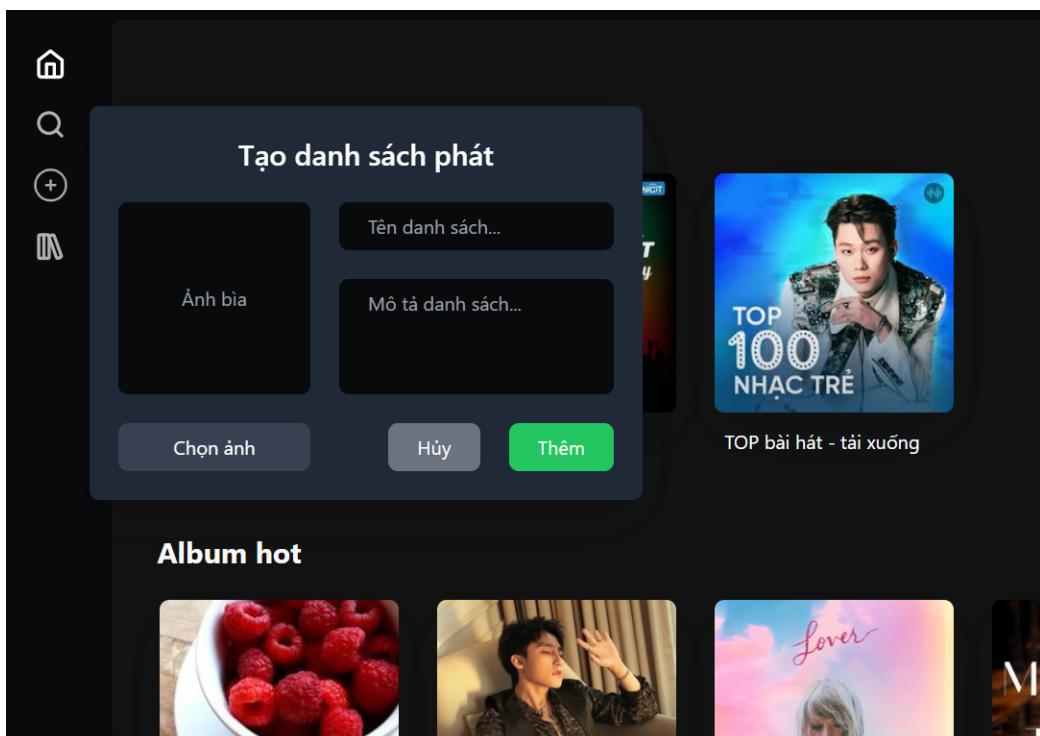


Hình 5.3: Chức năng tải video âm nhạc

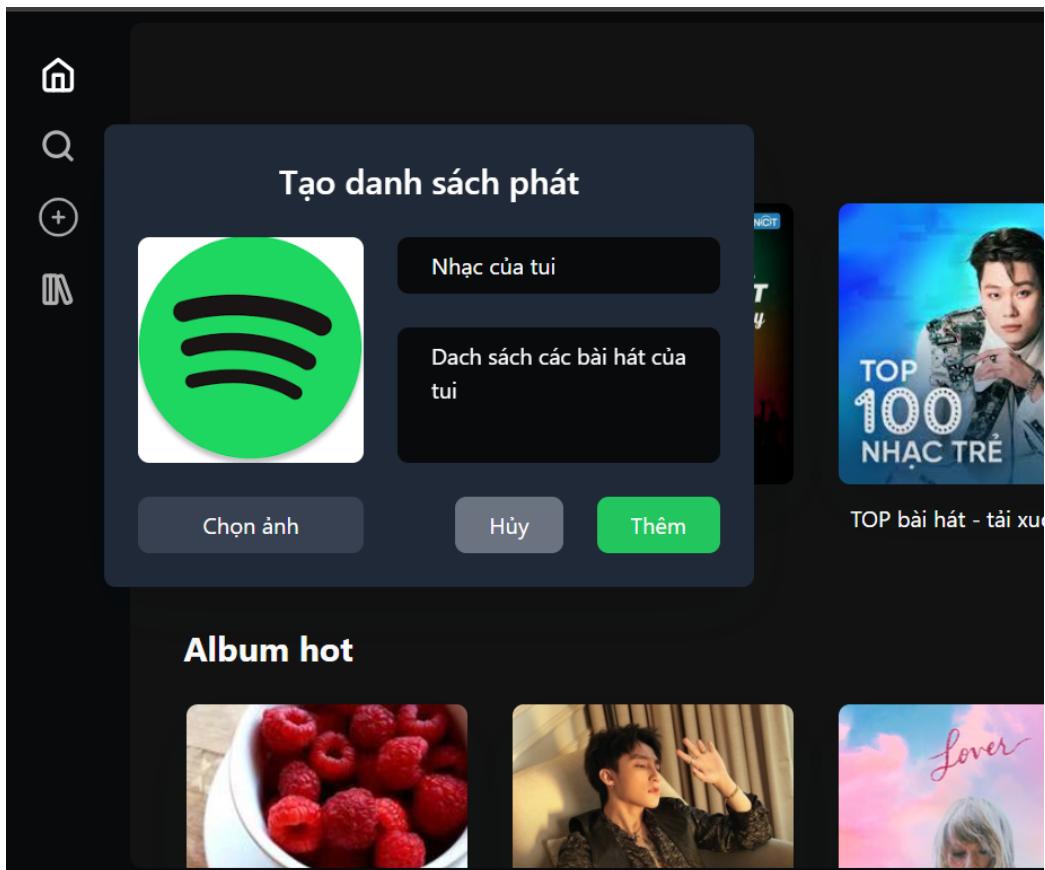
5.4 Chức năng tạo album và đánh dấu bài hát yêu thích

Người dùng có thể tạo album riêng để lưu trữ các bài hát yêu thích và dễ dàng truy cập lại. Ngoài ra, có thể đánh dấu các bài hát là “Yêu thích” để sắp xếp và lọc nhanh.

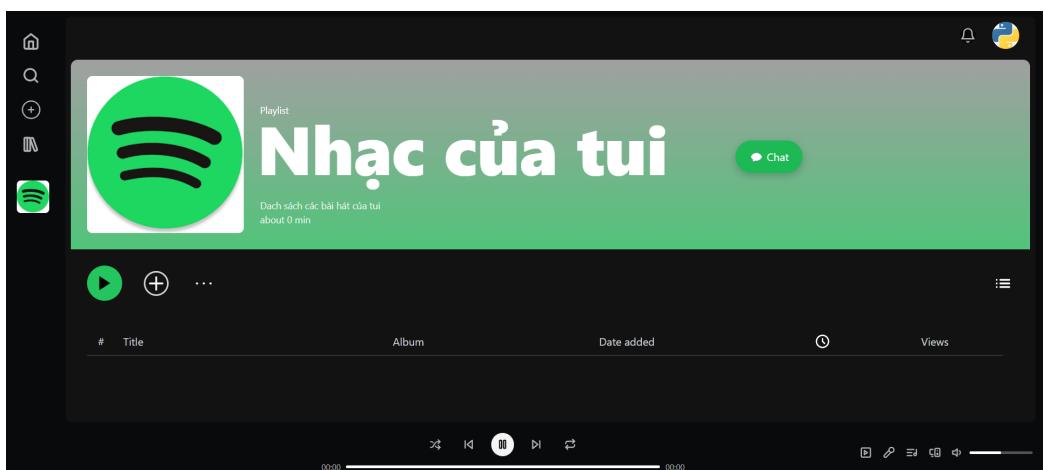
5.4.1 Chức năng tạo album



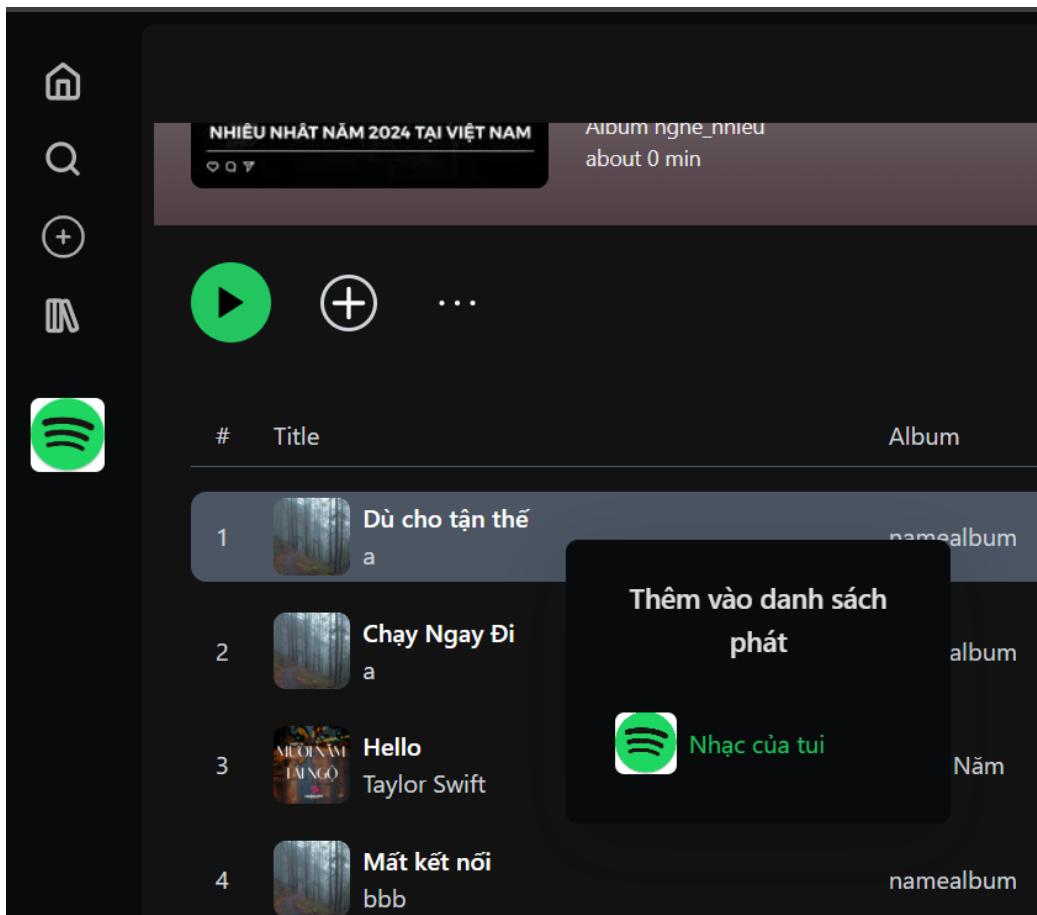
Hình 5.4: Giao diện hiện biểu mẫu tạo album



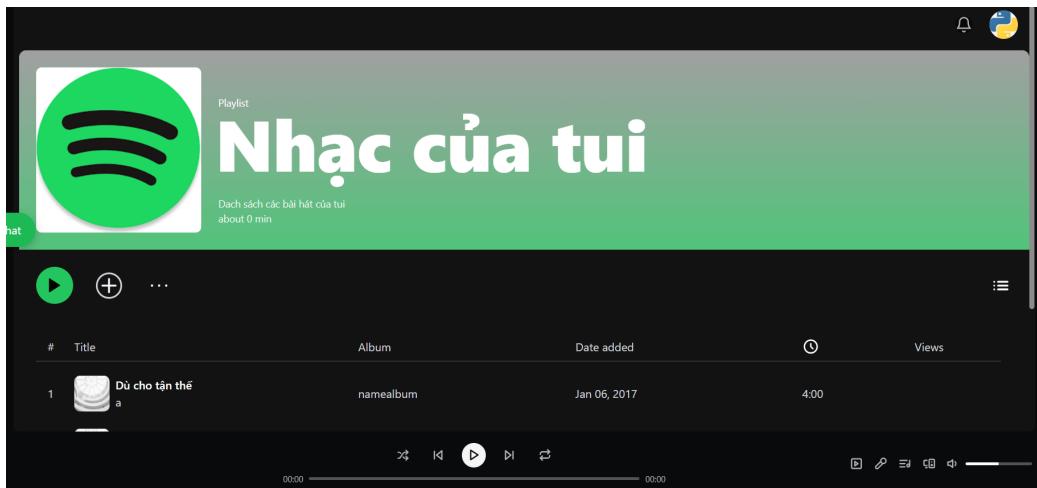
Hình 5.5: Giao diện điền thông tin tạo album



Hình 5.6: Giao diện tạo album thành công



Hình 5.7: Giao diện thêm bài hát vào album



Hình 5.8: Giao diện album đã bài hát thành công

5.4.2 Chức năng đánh dấu bài hát yêu thích

5.5 Quản trị

5.5.1 Quản lý người dùng

- Hiển thị danh sách người dùng

Quản lý người dùng							
STT	Ảnh đại diện	Tên hiển thị	Email	Số điện thoại	Trạng thái	Vai trò	Hành động
12		Lý Thị K	user10@example.com	0901234570	Hoạt động	Người dùng	Khóa Cấp quyền Admin
13		Nguyễn Đình Thông	nguyendinhthongd@gmail.com	0865824502	Hoạt động	Quản trị viên	Xem chi tiết Khóa
14		NguyenDinhThong	nguyenthong24446666@gmail.com		Hoạt động	Người dùng	Xem chi tiết Khóa Cấp quyền Admin

Hình 5.9: Hiển thị danh sách người dùng

- Xem chi tiết thông tin người dùng

Hình 5.10: Xem chi tiết thông tin người dùng

- Khóa tài khoản người dùng

Hình 5.11: Khóa tài khoản người dùng

Hình 5.12: Khóa tài khoản người dùng thành công

- Mở khóa tài khoản người dùng

The screenshot shows a user management interface. At the top, there's a navigation bar with links like Dashboard, Users, Song, Play List, Album, Artist, and Typy Song. Below that is a breadcrumb trail: Home / Dashboard. The main title is 'Quản lý người dùng'. A table lists two users: user9@example.com (Vũ Văn I) and user10@example.com (Lý Thị K). For user9@example.com, the status is 'Đã khóa' (Locked), and a modal window is open asking 'Bạn có chắc chắn muốn mở khóa tài khoản này không?' (Are you sure you want to unlock this account?). The modal has buttons 'Không' (No) and 'Có' (Yes). To the right of the table, there are buttons for 'Mở khóa' (Unlock) and 'Cấp quyền Admin' (Grant Admin Privileges). Below the table, there are 'Xem chi tiết' (View details) and 'Khóa' (Lock) buttons.

Hình 5.13: Mở khóa tài khoản người dùng

This screenshot shows the same user management interface after the unlock operation. The modal from the previous screenshot is now closed, and a success message 'Mở tài khoản thành công' (Account unlocked successfully) is displayed at the top. The user table remains the same, but the status for user9@example.com is now 'Hoạt động' (Active). The 'Xem chi tiết' and 'Khóa' buttons are still present to the right of the table.

Hình 5.14: Mở khóa tài khoản người dùng thành công

- Cấp quyền quản trị cho tài khoản

Quản lý người dùng							
STT	Ảnh đại diện	Tên hiển thị	Email	Số điện thoại	Trạng thái	Vai trò	Hành động
		m			Khóa		Cấp quyền Admin
12		Lý Thị K	user10@example.com	0901234570	Hoạt động	Người dùng	Cấp quyền Admin
13		Nguyễn Đinh Thông	nguyendinhthongd hsg@gmail.com	0865824502	Hoạt động	Quản trị viên	Xem chi tiết Khóa

Hình 5.15: Cấp quyền quản trị cho tài khoản



Quản lý người dùng							
STT	Ảnh đại diện	Tên hiển thị	Email	Số điện thoại	Trạng thái	Vai trò	Hành động
		m			Khóa		Xem chi tiết Khóa
12		Lý Thị K	user10@example.com	0901234570	Hoạt động	Quản trị viên	Xem chi tiết Khóa
13		Nguyễn Đinh Thông	nguyendinhthongd hsg@gmail.com	0865824502	Hoạt động	Quản trị viên	Xem chi tiết Khóa

Hình 5.16: Cấp quyền quản trị cho tài khoản thành công

5.5.2 Quản lý bài hát

- Danh sách bài hát

The screenshot shows a dark-themed dashboard with a sidebar on the left containing links for Dashboard, User, Song (selected), Playlist, Album, Artist, and TypeSong. The main content area is titled 'Danh sách bài hát' (Song List) and features a search bar and a 'Thêm bài hát' (Add Song) button. Below is a table with the following data:

ID	Tên bài hát	Thể loại	Thời lượng	ID nghệ sĩ	Ngày phát hành	Trạng thái	Hành động
1	Mặt kết nối	Pop	4 phút	2	6/1/2017	Active	Sửa Khóa
2	Giờ thi	Pop	4 phút	2	6/1/2017	Active	Sửa Khóa
3	Thích quá rồi nà	Pop	4 phút	1	6/1/2017	Active	Sửa Khóa
4	Dù cho tận thế	Pop	4 phút	1	6/1/2017	Active	Sửa Khóa
5	Lẽ đường	Pop	4 phút	1	6/1/2017	Active	Sửa Khóa

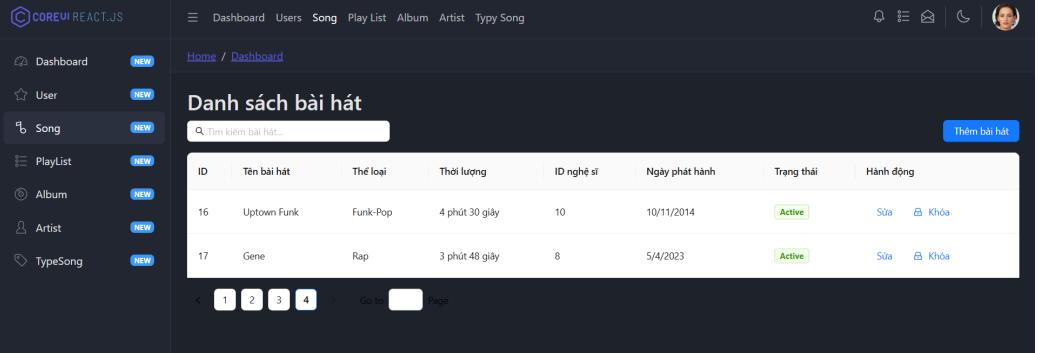
Pagination at the bottom shows pages 1-4, with a 'Go to' input field and a 'page' dropdown.

Hình 5.17: Danh sách bài hát

- Thêm bài hát mới

The screenshot shows the 'Thêm bài hát' (Add Song) modal dialog. It includes fields for 'Tên bài hát' (Name), 'Thể loại' (Genre), 'Ngày phát hành' (Release Date), 'ID nghệ sĩ' (Artist ID), and 'Tệp bài hát' (File). The 'Tệp bài hát' field shows a file selection button and the path 'Gene.mp3'. At the bottom are 'Cancel' and 'OK' buttons.

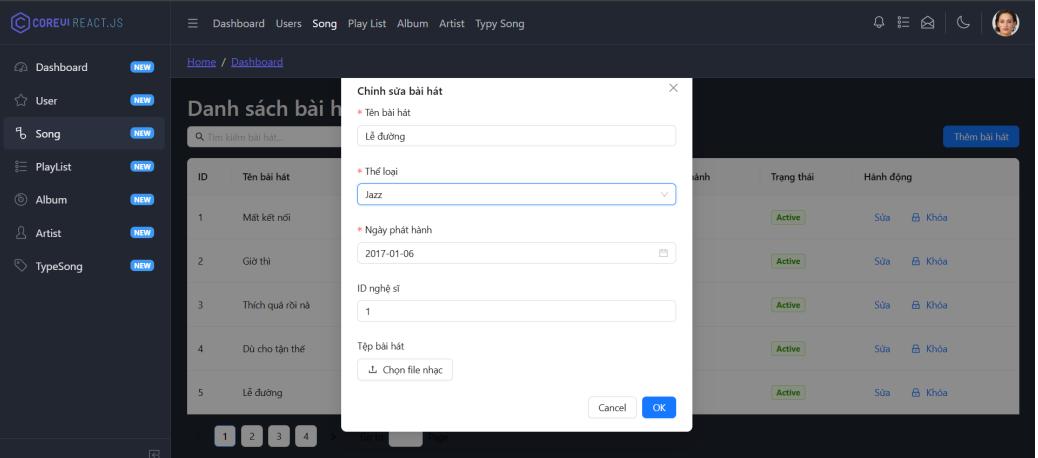
Hình 5.18: Thêm bài hát mới



The screenshot shows a dark-themed web application interface for managing songs. On the left, there's a sidebar with navigation links: Dashboard, User, Song (selected), Playlist, Album, Artist, and TypeSong. The main content area has a title 'Danh sách bài hát' (Song List) and a search bar. A table displays two songs: 'Uptown Funk' and 'Gene'. Both songs have an 'Active' status, 'Sửa' (Edit) and 'Khóa' (Lock) buttons, and a 'Hành động' (Action) column. At the top right of the table is a blue button labeled 'Thêm bài hát' (Add song). Below the table is a pagination control with pages 1-4 and a 'Go to page' input field.

Hình 5.19: Thêm bài hát mới thành công

- Sửa bài hát



This screenshot shows the same application interface as Figure 5.19, but with an edit dialog box overlaid on the 'Song' list. The dialog is titled 'Chỉnh sửa bài hát' (Edit song) and contains fields for 'Tên bài hát' (Song name) set to 'Lê đường', 'Thể loại' (Genre) set to 'Jazz', 'Ngày phát hành' (Release date) set to '2017-01-06', 'ID nghệ sĩ' (Artist ID) set to '1', and a file upload field for 'Tệp bài hát' (Song file) with the placeholder 'Chọn file nhạc' (Select music file). At the bottom of the dialog are 'Cancel' and 'OK' buttons.

Hình 5.20: Sửa bài hát

The screenshot shows a dark-themed dashboard with a sidebar containing links like Dashboard, User, Song, Playlist, Album, Artist, and TypeSong. The 'Song' link is highlighted with a blue badge labeled 'NEW'. The main content area displays a success message 'Cập nhật bài hát thành công!' (Song update successful!). Below it is a table titled 'Danh sách bài hát' (Song list) with columns: ID, Tên bài hát, Thể loại, Thời lượng, ID nghệ sĩ, Ngày phát hành, Trạng thái, and Hành động. The table contains five rows of song data. A search bar and a 'Thêm bài hát' (Add song) button are also visible.

ID	Tên bài hát	Thể loại	Thời lượng	ID nghệ sĩ	Ngày phát hành	Trạng thái	Hành động
1	Mặt kết nối	Pop	4 phút	2	6/1/2017	Active	Sửa <input type="checkbox"/> Khóa
2	Giờ thi	Pop	4 phút	2	6/1/2017	Active	Sửa <input type="checkbox"/> Khóa
3	Thích quá rồi nà	Pop	4 phút	1	6/1/2017	Active	Sửa <input type="checkbox"/> Khóa
4	Dù cho tần thẻ	Pop	4 phút	1	6/1/2017	Active	Sửa <input type="checkbox"/> Khóa
5	Lê đường	Jazz	4 phút	1	6/1/2017	Active	Sửa <input type="checkbox"/> Khóa

Hình 5.21: Sửa bài hát thành công

- Khóa bài hát

This screenshot is similar to Figure 5.21 but shows a modal dialog asking 'Bạn có chắc muốn khóa bài hát này?' (Are you sure you want to lock this song?). The user has selected 'Có' (Yes). The main table and sidebar are identical to the previous figure.

ID	Tên bài hát	Thể loại	Thời lượng	ID nghệ sĩ	Ngày phát hành	Trạng thái	Hành động
1	Mặt kết nối	Pop	4 phút	2	6/1/2017	Active	Sửa <input type="checkbox"/> Khóa
2	Giờ thi	Pop	4 phút	2	6/1/2017	Active	Sửa <input type="checkbox"/> Khóa
3	Thích quá rồi nà	Pop	4 phút	1	6/1/2017	Active	Sửa <input type="checkbox"/> Khóa

Hình 5.22: Khóa bài hát

This screenshot shows the same interface as Figure 5.21 and 5.22, but the first song in the list now has its status changed to 'inactive' (không hoạt động) in the 'Trạng thái' column. The other songs remain active.

ID	Tên bài hát	Thể loại	Thời lượng	ID nghệ sĩ	Ngày phát hành	Trạng thái	Hành động
1	Mặt kết nối	Pop	4 phút	2	6/1/2017	inactive	Sửa <input type="checkbox"/> Mở khóa
2	Giờ thi	Pop	4 phút	2	6/1/2017	Active	Sửa <input type="checkbox"/> Khóa
3	Thích quá rồi nà	Pop	4 phút	1	6/1/2017	Active	Sửa <input type="checkbox"/> Khóa

Hình 5.23: Khóa bài hát thành công

- Mở khóa bài hát

Danh sách bài hát

ID	Tên bài hát	Thể loại	Thời lượng	ID nghệ sĩ	Ngày phát hành	Trạng thái	Hành động
1	Mắt két nỗi	Pop	4 phút	2	6/1/2017	Inactive	Sửa Mở khóa
2	Giờ thì	Pop	4 phút	2	6/1/2017	Active	Sửa Khóa
3	Thích quá rồi nà	Pop	4 phút	1	6/1/2017	Active	Sửa Khóa
4	Dù cho tận thế	Pop	4 phút	1	6/1/2017	Active	Sửa Khóa
5	Lẽ đường	Jazz	4 phút	1	6/1/2017	Active	Sửa Khóa

1 2 3 4 > Go to page

Hình 5.24: Mở khóa bài hát

Danh sách bài hát

ID	Tên bài hát	Thể loại	Thời lượng	ID nghệ sĩ	Ngày phát hành	Trạng thái	Hành động
1	Mắt két nỗi	Pop	4 phút	2	6/1/2017	Active	Sửa Khóa
2	Giờ thì	Pop	4 phút	2	6/1/2017	Active	Sửa Khóa
3	Thích quá rồi nà	Pop	4 phút	1	6/1/2017	Active	Sửa Khóa
4	Dù cho tận thế	Pop	4 phút	1	6/1/2017	Active	Sửa Khóa
5	Lẽ đường	Jazz	4 phút	1	6/1/2017	Active	Sửa Khóa

Hình 5.25: Mở khóa bài hát thành công

- Tìm kiếm bài hát

ID	Tên bài hát	Thể loại	Thời lượng	ID nghệ sĩ	Ngày phát hành	Trạng thái	Hành động
2	Giờ thi	Pop	4 phút	2	6/1/2017	Active	Sửa Khóa

Hình 5.26: Tìm kiếm bài hát

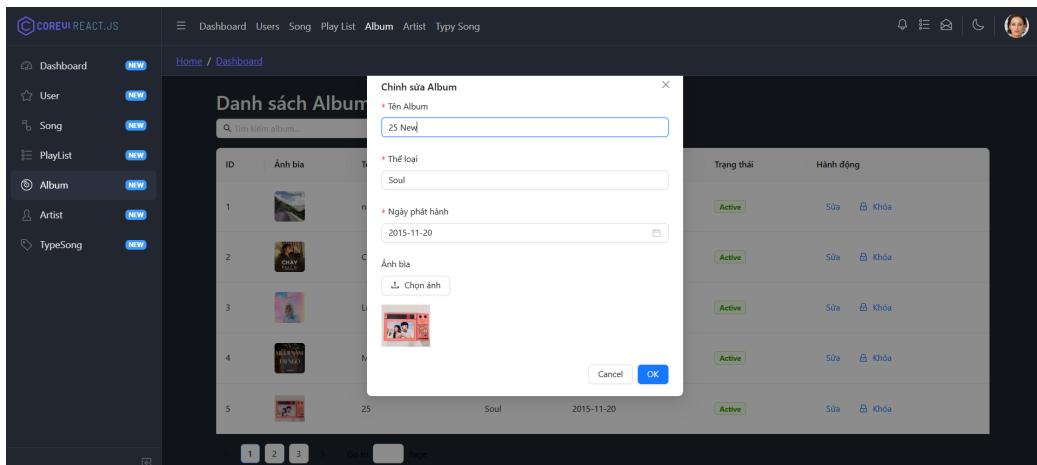
5.5.3 Quản lý album

- Danh sách album

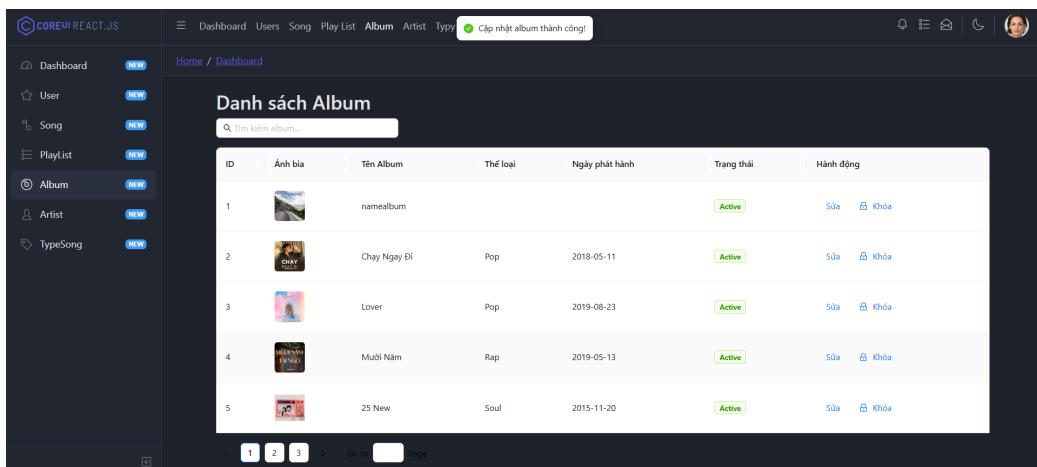
ID	Ảnh bìa	Tên Album	Thể loại	Ngày phát hành	Trạng thái	Hành động
1		namealbum	Pop	2018-05-11	Active	Sửa Khóa
2		Chay Ngay Đì	Pop	2018-05-11	Active	Sửa Khóa
3		Lover	Pop	2019-08-23	Active	Sửa Khóa
4		Mười Năm	Rap	2019-05-13	Active	Sửa Khóa
5		25	Soul	2015-11-20	Active	Sửa Khóa

Hình 5.27: Danh sách album

- Sửa album



Hình 5.28: Sửa album



Hình 5.29: Sửa album thành công

- Khóa album

ID	Ảnh bìa	Tên Album	Thể loại	Ngày phát hành	Trạng thái	Hành động
1		namealbum			Active	Sửa ⚒ Khóa
2		Chạy Ngay Đi	Pop	2018-05-11	Active	Bạn có chắc chắn muốn khóa album này? Không Có
3		Lover	Pop	2019-08-23	Active	Sửa ⚒ Khóa
4		Mười Năm	Rap	2019-05-13	Active	Sửa ⚒ Khóa
5		25 New	Soul	2015-11-20	Active	Sửa ⚒ Khóa

Hình 5.30: Khóa album

ID	Ảnh bìa	Tên Album	Thể loại	Ngày phát hành	Trạng thái	Hành động
1		namealbum			Active	Sửa ⚒ Khóa
2		Chạy Ngay Đi	Pop	2018-05-11	Active	Sửa ⚒ Khóa
3		Lover	Pop	2019-08-23	Inactive	Sửa ⚒ Mở khóa
4		Mười Năm	Rap	2019-05-13	Active	Sửa ⚒ Khóa

Hình 5.31: Khóa album thành công

- Mở khóa album

Danh sách Album						
Tim kiem album...						
ID	Ảnh bìa	Tên Album	Thể loại	Ngày phát hành	Trạng thái	Hành động
1		namealbum			Active	Sửa Khóa
2		Chạy Ngay Đi	Pop	2018-05-11	Active	Bạn có chắc muốn mở khóa album này? Không Có
3		Lover	Pop	2019-08-23	Inactive	Sửa Mở khóa
4		Mười Năm	Rap	2019-05-13	Active	Sửa Khóa
5		25 New	Soul	2015-11-20	Active	Sửa Khóa

1 2 3 > Go to Page

Hình 5.32: Mở khóa album

Danh sách Album						
Tim kiem album...						
ID	Ảnh bìa	Tên Album	Thể loại	Ngày phát hành	Trạng thái	Hành động
1		namealbum			Active	Sửa Khóa
2		Chạy Ngay Đi	Pop	2018-05-11	Active	Sửa Khóa
3		Lover	Pop	2019-08-23	Active	Sửa Khóa
4		Mười Năm	Rap	2019-05-13	Active	Sửa Khóa
5		25 New	Soul	2015-11-20	Active	Sửa Khóa

1 2 3 > Go to Page

Hình 5.33: Mở khóa album thành công

- Tìm kiếm album

ID	Ánh bìa	Tên Album	Thể loại	Ngày phát hành	Trạng thái	Hành động
5		25 New	Soul	2015-11-20	Active	Sửa Khóa

Hình 5.34: Tìm kiếm album

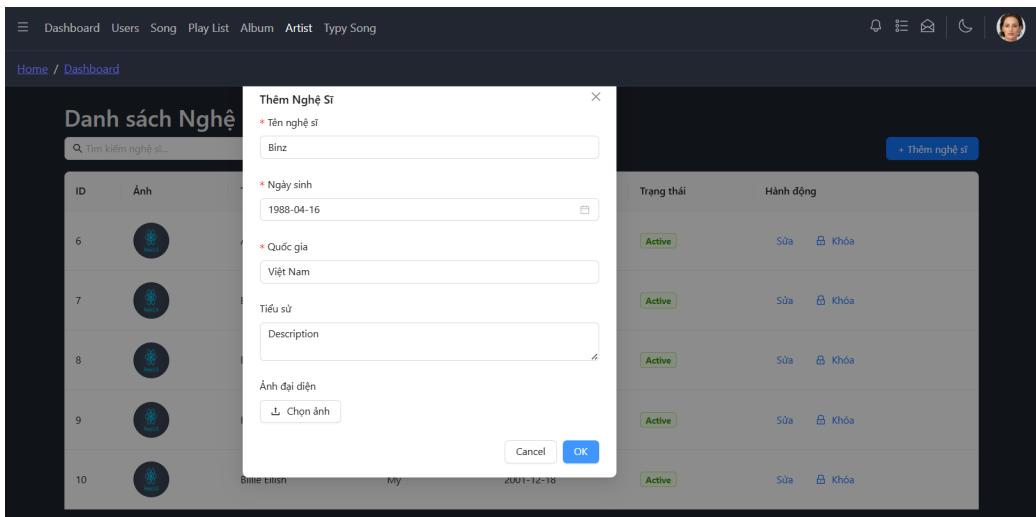
5.5.4 Quản lý nghệ sĩ

- Danh sách nghệ sĩ

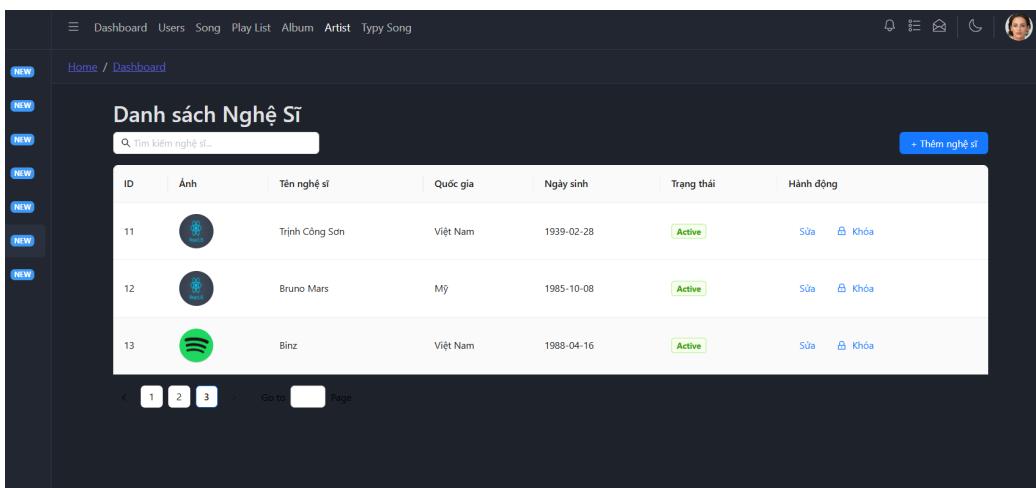
ID	Ảnh	Tên nghệ sĩ	Quốc gia	Ngày sinh	Trạng thái	Hành động
6		Adele	Anh	1988-05-05	Active	Sửa Khóa
7		Bích Phương	Việt Nam	1989-09-30	Active	Sửa Khóa
8		Ed Sheeran	Anh	1991-02-17	Active	Sửa Khóa
9		Hà Anh Tuấn	Việt Nam	1984-12-17	Active	Sửa Khóa
10		Billie Eilish	Mỹ	2001-12-18	Active	Sửa Khóa

Hình 5.35: Danh sách nghệ sĩ

- Thêm nghệ sĩ mới

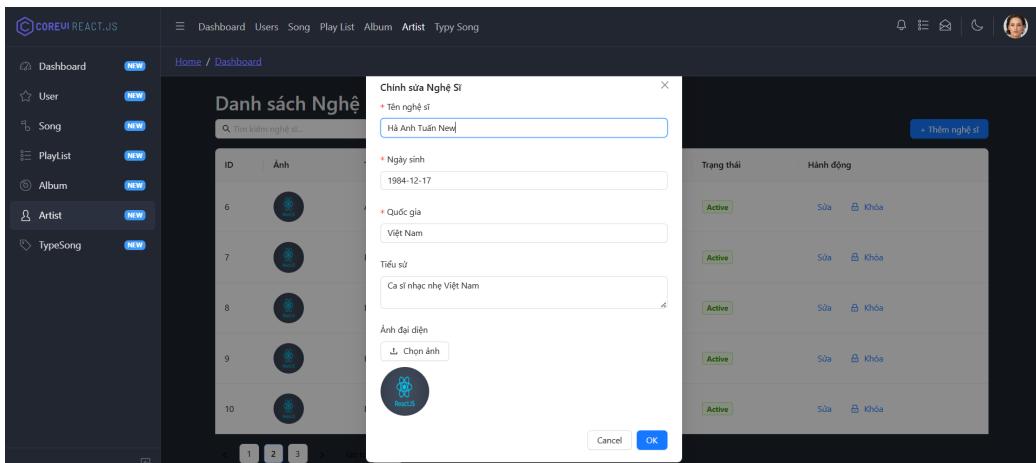


Hình 5.36: Thêm nghệ sĩ mới



Hình 5.37: Thêm nghệ sĩ mới thành công

- Sửa nghệ sĩ



Hình 5.38: Sửa nghệ sĩ

ID	Ảnh	Tên nghệ sĩ	Nationality	Ngày sinh	Trạng thái	Hành động
6		Adele	Anh	1988-05-05	Active	Sửa Khóa
7		Bích Phương	Việt Nam	1989-09-30	Active	Sửa Khóa
8		Ed Sheeran	Anh	1991-02-17	Active	Sửa Khóa
9		Hà Anh Tuấn New	Việt Nam	1984-12-17	Active	Sửa Khóa
10		Billie Eilish	Mỹ	2001-12-18	Active	Sửa Khóa

Hình 5.39: Sửa nghệ sĩ thành công

- Khóa nghệ sĩ

The screenshot shows a dark-themed web application interface. At the top, there is a navigation bar with links: Dashboard, Users, Song, Play List, Album, Artist, Typy Song, and a user profile icon. Below the navigation is a breadcrumb trail: Home / Dashboard. The main title is "Danh sách Nghệ Sĩ". A search bar and a "Thêm nghệ sĩ" button are visible. The table has columns: ID, Ánh (Thumbnail), Tên nghệ sĩ, Quốc gia, Ngày sinh, Trạng thái, and Hành động. The rows show artists Adele, Bích Phương, Ed Sheeran, Hà Anh Tuấn New, and Billie Eilish. For Bích Phương (ID 7), a modal window appears asking if the user wants to lock the artist, with options "Không" (No) and "Có" (Yes). The "Có" button is highlighted in blue.

ID	Ánh	Tên nghệ sĩ	Quốc gia	Ngày sinh	Trạng thái	Hành động
6		Adele	Anh	1988-05-05	Active	Bạn có chắc muốn khóa nghệ sĩ này? Không Có
7		Bích Phương	Việt Nam	1989-09-30	Active	Sửa Khóa
8		Ed Sheeran	Anh	1991-02-17	Active	Sửa Khóa
9		Hà Anh Tuấn New	Việt Nam	1984-12-17	Active	Sửa Khóa
10		Billie Eilish	Mỹ	2001-12-18	Active	Sửa Khóa

Hình 5.40: Khóa nghệ sĩ

This screenshot shows the same application interface after locking the artist Bích Phương. The modal from the previous screenshot has disappeared. The status for Bích Phương is now "Inactive" (magenta border). The message "Cập nhật trạng thái thành công!" (Update status successful!) is displayed at the top right. The rest of the table and interface remain the same.

ID	Ánh	Tên nghệ sĩ	Quốc gia	Ngày sinh	Trạng thái	Hành động
6		Adele	Anh	1988-05-05	Active	Sửa Khóa
7		Bích Phương	Việt Nam	1989-09-30	Inactive	Sửa Mở khóa
8		Ed Sheeran	Anh	1991-02-17	Active	Sửa Khóa
9		Hà Anh Tuấn New	Việt Nam	1984-12-17	Active	Sửa Khóa

Hình 5.41: Khóa nghệ sĩ thành công

- Mở khóa nghệ sĩ

ID	Ảnh	Tên nghệ sĩ	Quốc gia	Ngày sinh	Trạng thái	Hành động
6		Adele	Anh	1988-05-05	Active	Bạn có chắc muốn mở khóa nghệ sĩ này? Không Có
7		Bích Phương	Việt Nam	1989-09-30	Inactive	Sửa Mở khóa
8		Ed Sheeran	Anh	1991-02-17	Active	Sửa Khóa
9		Hà Anh Tuấn New	Việt Nam	1984-12-17	Active	Sửa Khóa

Hình 5.42: Mở khóa nghệ sĩ

ID	Ảnh	Tên nghệ sĩ	Quốc gia	Ngày sinh	Trạng thái	Hành động
6		Adele	Anh	1988-05-05	Active	Sửa Khóa
7		Bích Phương	Việt Nam	1989-09-30	Active	Sửa Khóa
8		Ed Sheeran	Anh	1991-02-17	Active	Sửa Khóa
9		Hà Anh Tuấn New	Việt Nam	1984-12-17	Active	Sửa Khóa
10		Billie Eilish	Mỹ	2001-12-18	Active	Sửa Khóa

Hình 5.43: Mở khóa nghệ sĩ thành công

- Tìm kiếm nghệ sĩ

ID	Ảnh	Tên nghệ sĩ	Nationality	Birth Date	Status	Action
11		Trịnh Công Sơn	Việt Nam	1939-02-28	Active	Sửa Xóa

Hình 5.44: Tìm kiếm nghệ sĩ

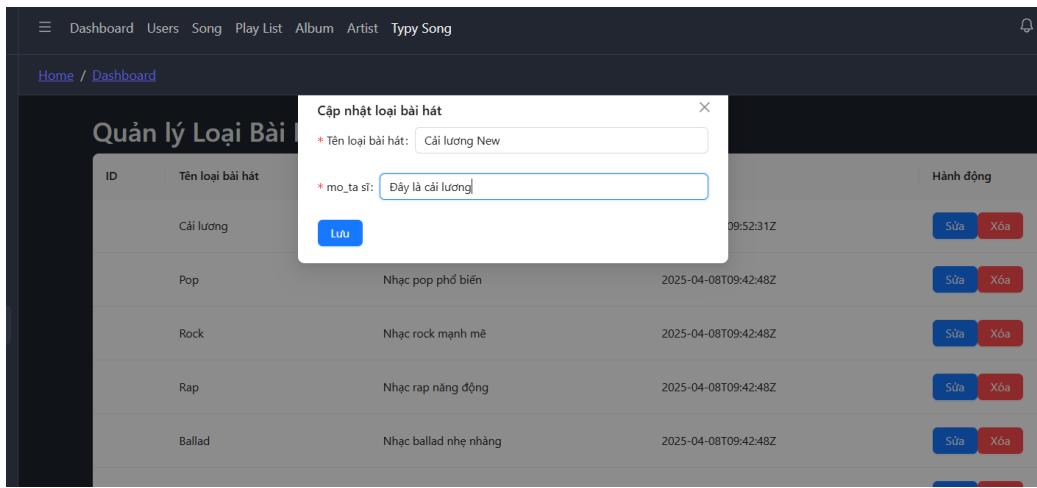
5.5.5 Quản lý loại bài hát

- Danh sách loại bài hát

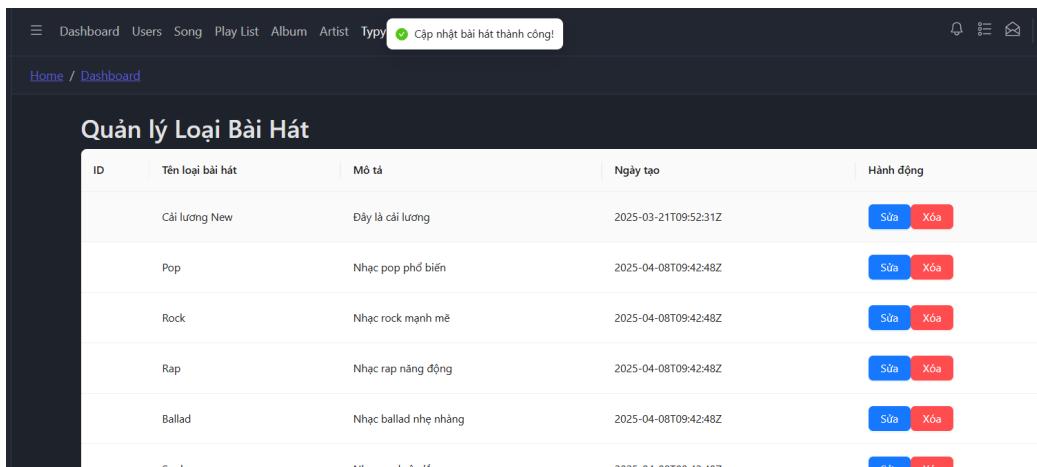
ID	Tên loại bài hát	Mô tả	Ngày tạo	Hành động
	Cải lương	Đây là cải lương nè	2025-03-21T09:52:31Z	Sửa Xóa
	Pop	Nhạc pop phổ biến	2025-04-08T09:42:48Z	Sửa Xóa
	Rock	Nhạc rock mạnh mẽ	2025-04-08T09:42:48Z	Sửa Xóa
	Rap	Nhạc rap năng động	2025-04-08T09:42:48Z	Sửa Xóa
	Ballad	Nhạc ballad nhẹ nhàng	2025-04-08T09:42:48Z	Sửa Xóa
	Soul	Nhạc soul sâu lắng	2025-04-08T09:42:48Z	Sửa Xóa
	Alternative	Nhạc thay thế độc đáo	2025-04-08T09:42:48Z	Sửa Xóa
	Nhạc Trịnh	Nhạc Trịnh Công Sơn	2025-04-08T09:42:48Z	Sửa Xóa

Hình 5.45: Danh sách loại bài hát

- Sửa loại bài hát



Hình 5.46: Sửa loại bài hát



Hình 5.47: Sửa loại bài hát

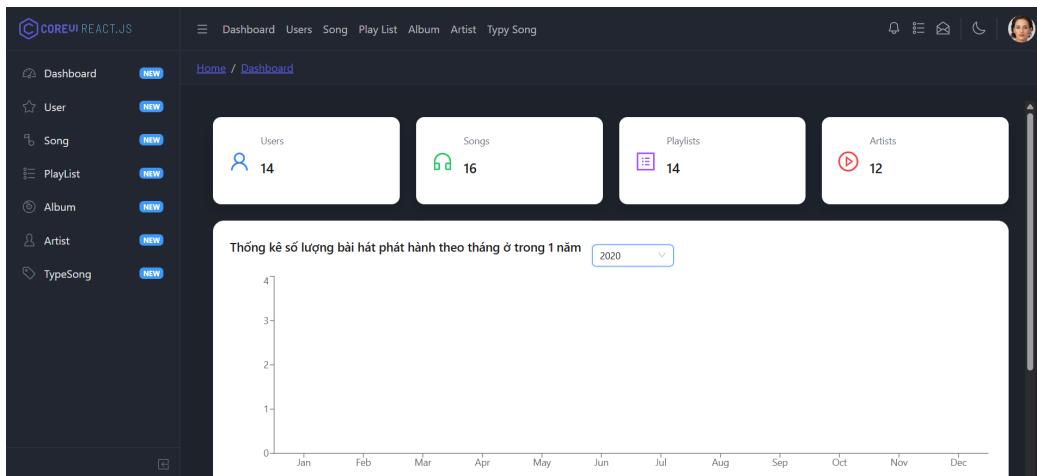
- Xóa loại bài hát

ID	Tên loại bài hát	Mô tả	Ngày tạo	Hành động
	Pop	Nhạc pop phổ biến	2025-04-08T09:42:48Z	<button>Sửa</button> <button>Xóa</button>
	Rock	Nhạc rock mạnh mẽ	2025-04-08T09:42:48Z	<button>Sửa</button> <button>Xóa</button>
	Rap	Nhạc rap năng động	2025-04-08T09:42:48Z	<button>Sửa</button> <button>Xóa</button>
	Ballad	Nhạc ballad nhẹ nhàng	2025-04-08T09:42:48Z	<button>Sửa</button> <button>Xóa</button>
	Soul	Nhạc soul sâu lắng	2025-04-08T09:42:48Z	<button>Sửa</button> <button>Xóa</button>
	Alternative	Nhạc thay thế độc đáo	2025-04-08T09:42:48Z	<button>Sửa</button> <button>Xóa</button>

Hình 5.48: Xóa loại bài hát

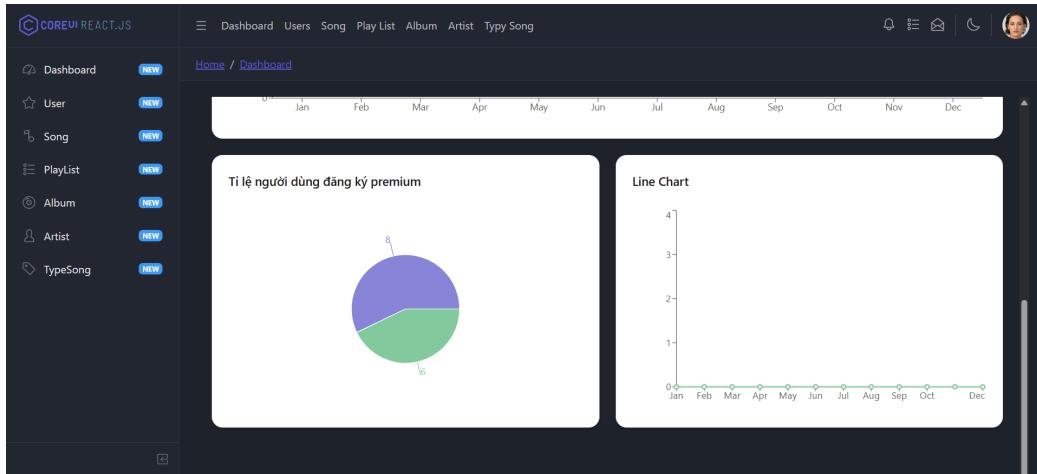
5.5.6 Thông kê

- Thông kê: Số lượng người dùng, bài hát, danh sách phát, nghệ sĩ, bài hát phát hành trong 1 năm



Hình 5.49: Thông kê: Số lượng người dùng, bài hát, danh sách phát, nghệ sĩ, bài hát phát hành trong 1 năm

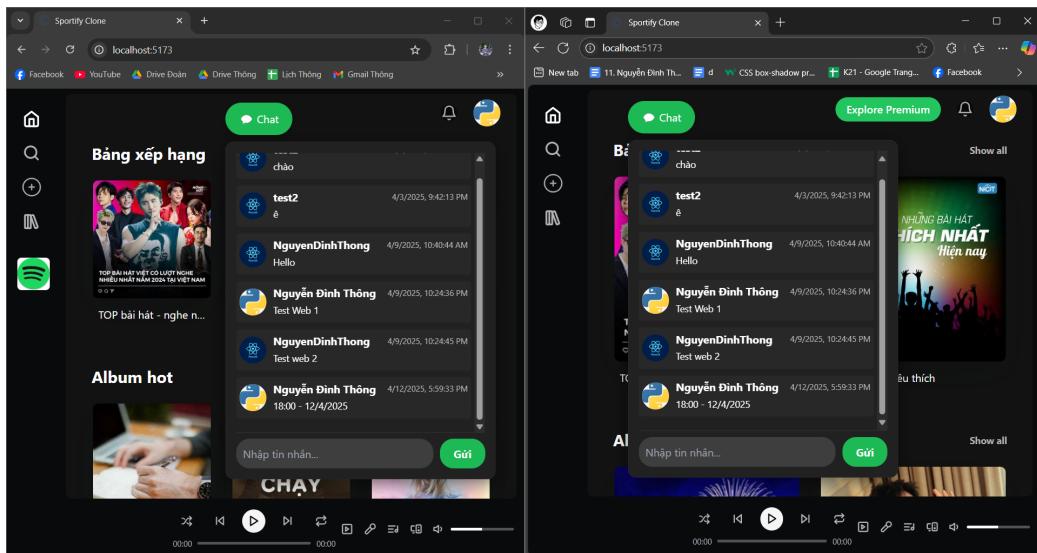
- Thông kê: Tỷ lệ người dùng đăng ký Premium theo biểu đồ tròn và biểu đồ đường



Hình 5.50: Thông kê: Tỷ lệ người dùng đăng ký Premium theo biểu đồ tròn và biểu đồ đường

5.6 Tính năng chat tích hợp

Ứng dụng hỗ trợ người dùng gửi tin nhắn trực tiếp trong giao diện web, giúp tăng cường tương tác cộng đồng. Tính năng có thể được tích hợp bằng WebSocket.

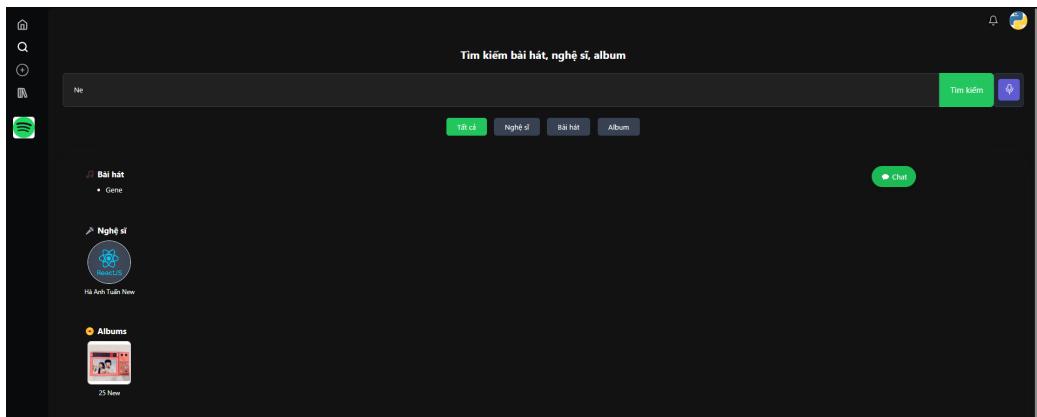


Hình 5.51: Giao diện tính năng chat

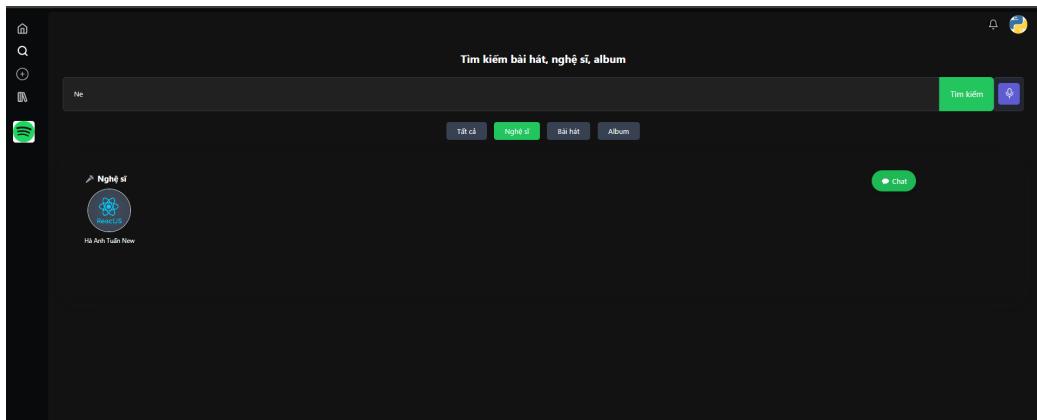
5.7 Chức năng tìm kiếm

5.7.1 Tìm kiếm theo từ khóa

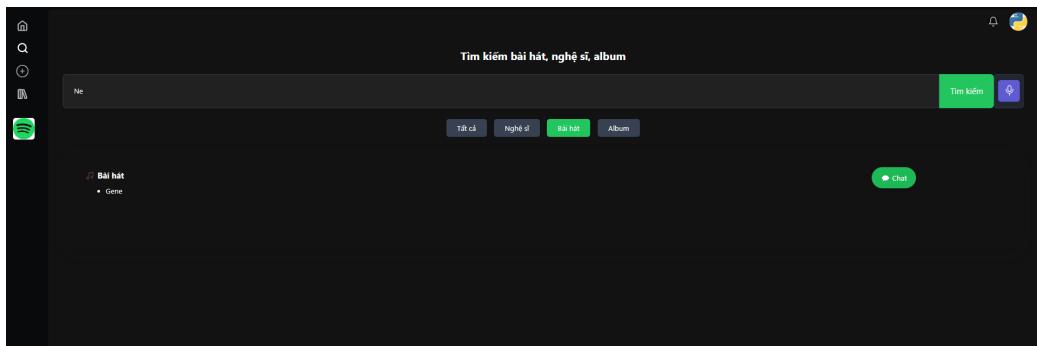
Chức năng cho phép người dùng nhập từ khóa (tên bài hát, ca sĩ, thể loại, v.v.) vào thanh tìm kiếm để nhận kết quả gợi ý phù hợp. Kết quả sẽ hiển thị danh sách các bài hát, ca sĩ, album tương ứng và cho phép lọc.



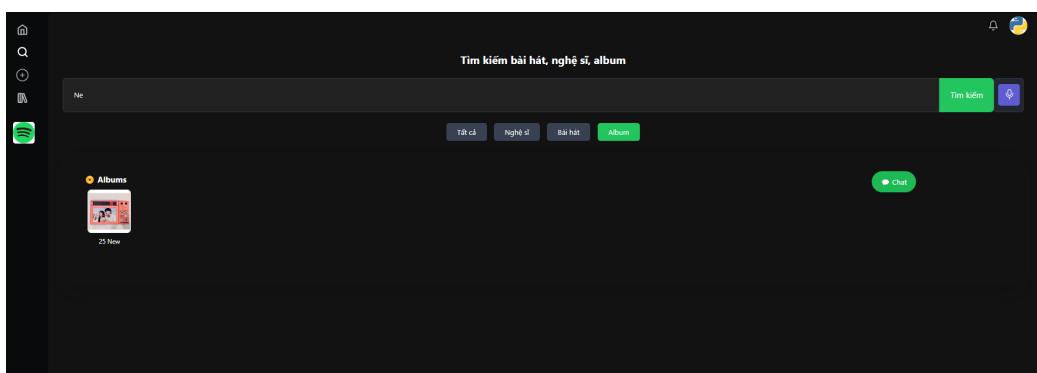
Hình 5.52: Giao diện tìm kiếm theo từ khóa



Hình 5.53: Giao diện tìm kiếm nghệ sĩ



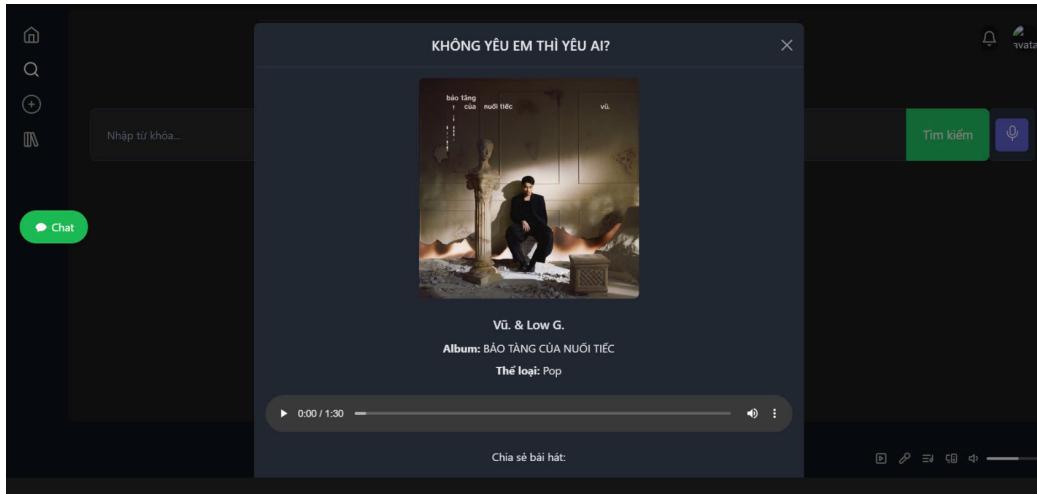
Hình 5.54: Giao diện tìm kiếm bài hát



Hình 5.55: Giao diện tìm kiếm bài hát

5.7.2 Tìm kiếm bài hát dựa trên tệp ghi âm

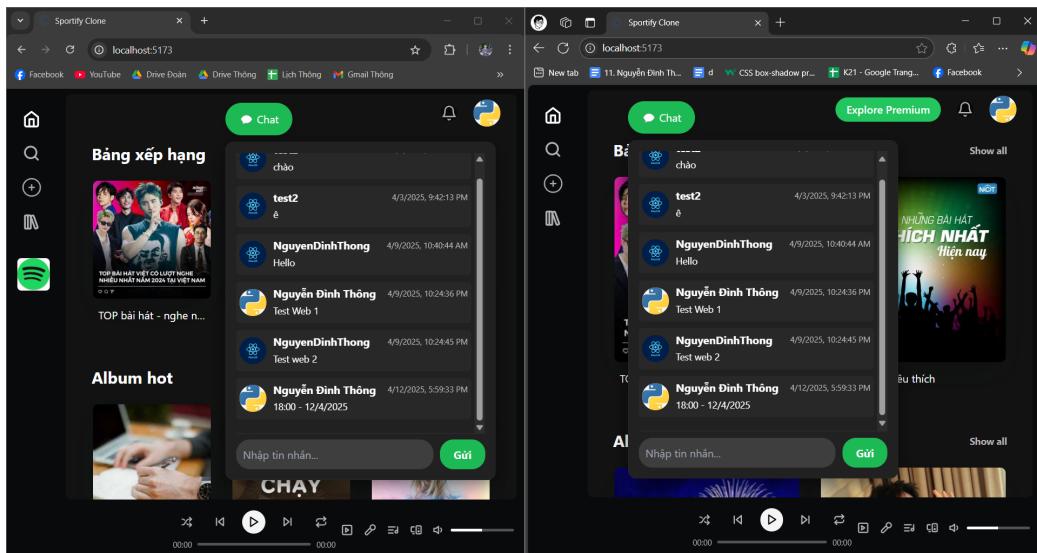
Người dùng có thể ghi âm đoạn bài hát. Hệ thống sử dụng mô hình mã nguồn mở Shazamio để phân tích và trả về dữ liệu của bài hát giống nhất.



Hình 5.56: Giao diện tìm kiếm bài hát dựa trên tệp ghi âm

5.8 Tính năng chat tích hợp

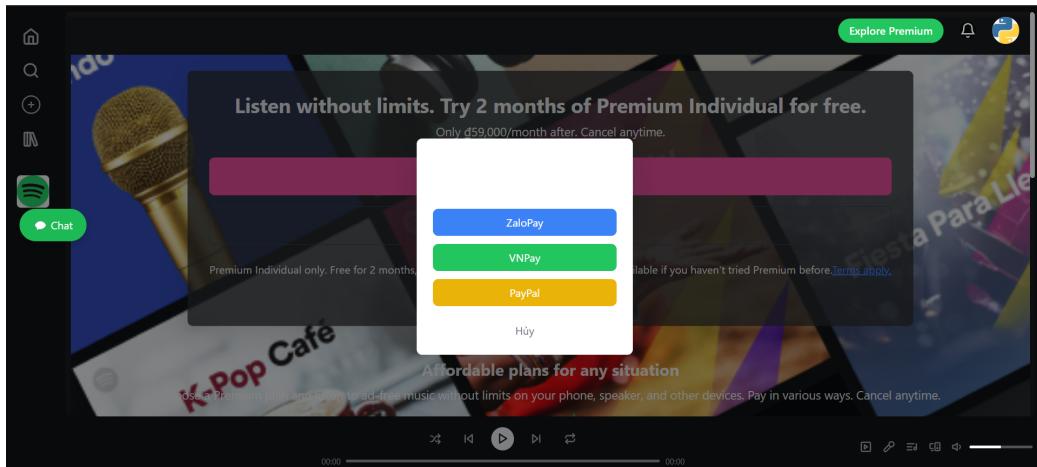
Ứng dụng hỗ trợ người dùng gửi tin nhắn trực tiếp trong giao diện web, giúp tăng cường tương tác cộng đồng. Tính năng có thể được tích hợp bằng WebSocket.



Hình 5.57: Giao diện tính năng chat

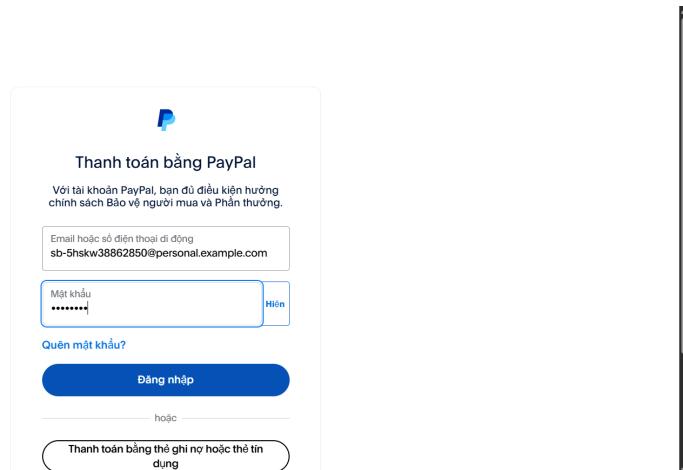
5.9 Premium

Khi người dùng mua Premium thì sẽ nghe nhạc mà không bị chèn quảng cáo giữa các bài.

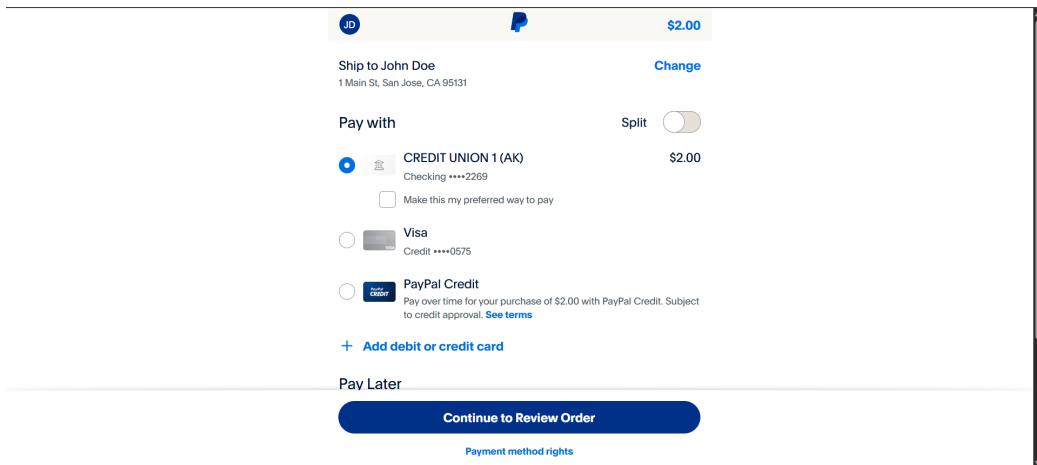


Hình 5.58: Giao diện mua premium

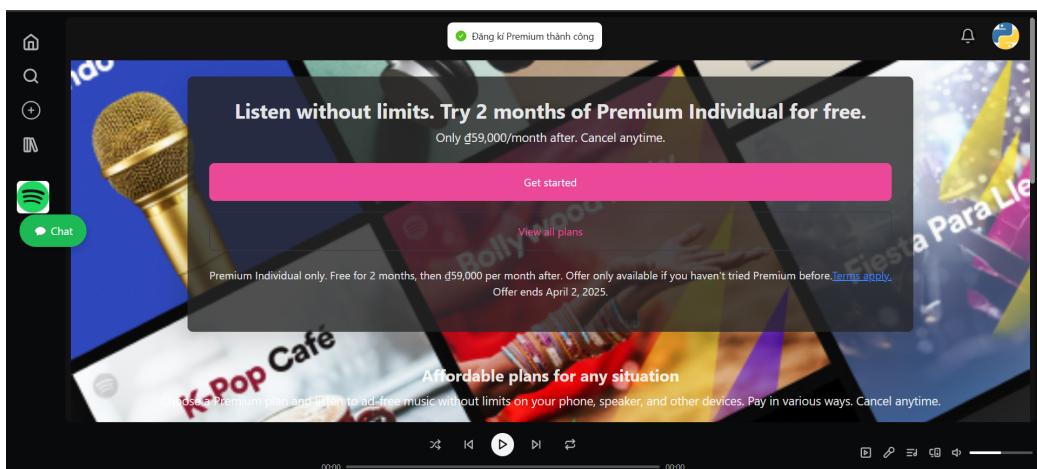
5.9.1 Mua Premium thanh toán bằng PayPal



Hình 5.59: Mua Premium thanh toán bằng PayPal

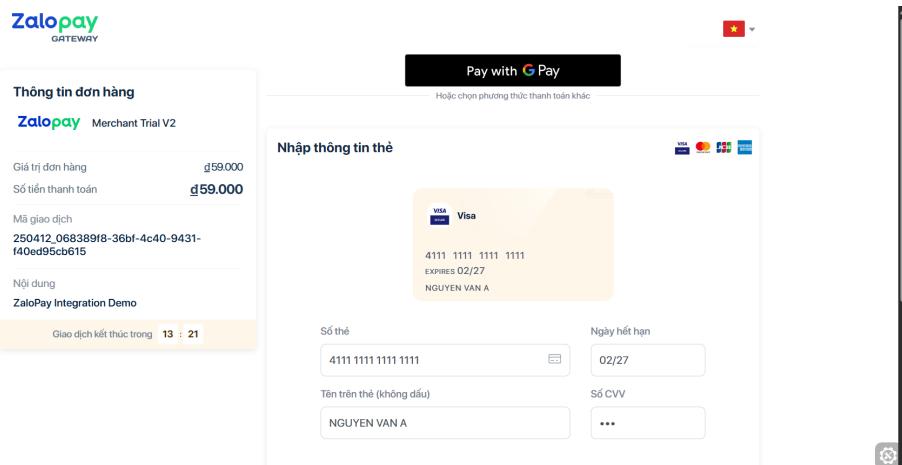


Hình 5.60: Mua Premium thanh toán bằng PayPal

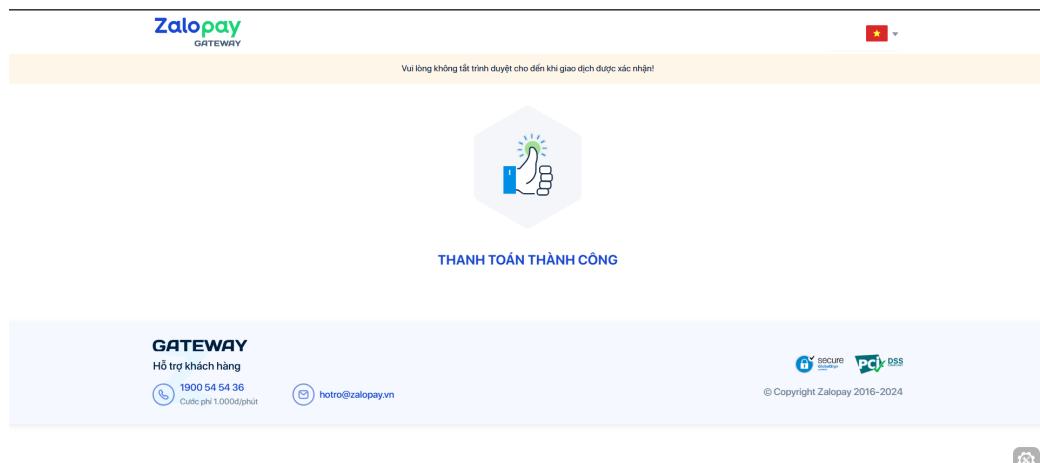


Hình 5.61: Mua Premium thanh toán bằng PayPal thành công

5.9.2 Mua Premium thanh toán bằng Zalo pay



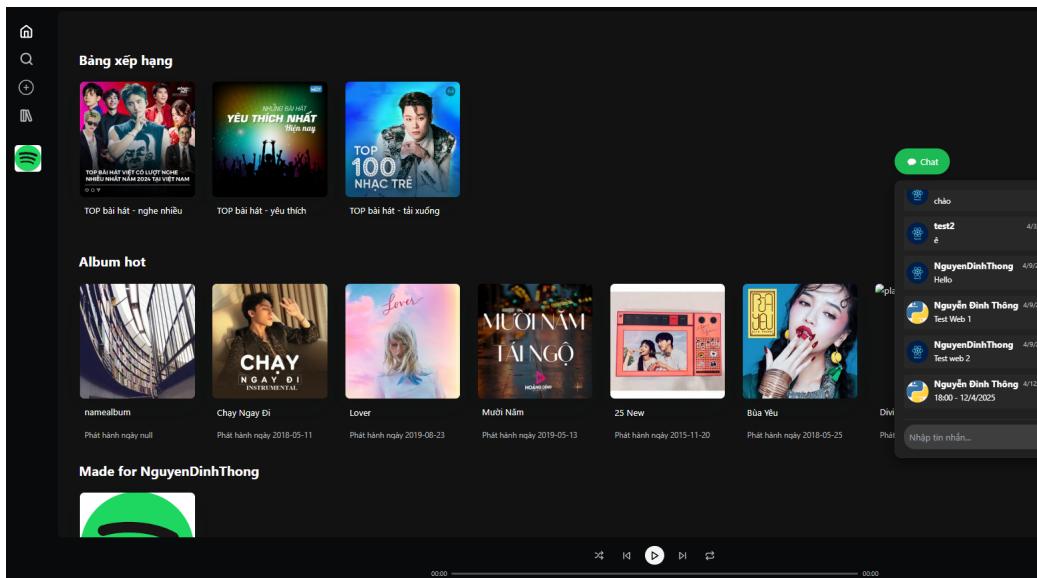
Hình 5.62: Mua Premium thanh toán bằng Zalo pay



Hình 5.63: Mua Premium thanh toán bằng Zalo pay thành công

5.10 Bảng xếp hạng và album hot

Cung cấp hệ thống bảng xếp hạng âm nhạc thịnh hành dựa vào lượt nghe.



Hình 5.64: Giao diện bảng xếp hạng và album hot

Chương 6

HƯỚNG PHÁT TRIỂN²

- **Tích hợp các dịch vụ phát nhạc trực tuyến:** Tích hợp các dịch vụ phát nhạc trực tuyến để cung cấp nguồn nhạc phong phú và chất lượng cao.
- **Cải thiện trải nghiệm người dùng (UX/UI):** Cải thiện trải nghiệm người dùng (UX/UI): Thiết kế giao diện hấp dẫn, thân thiện và dễ sử dụng hơn, đồng thời tối ưu hóa hiệu suất để người dùng có trải nghiệm mượt mà.
- **Phát triển các tính năng xã hội:** Phát triển các tính năng xã hội: Cho phép người dùng chia sẻ playlist, theo dõi bạn bè và xem hoạt động nghe nhạc của họ để tạo sự kết nối trong cộng đồng người dùng.
- **Hỗ trợ đa nền tảng:** Hỗ trợ đa nền tảng: Đảm bảo ứng dụng hoạt động trên nhiều thiết bị như iPad, di động và máy tính để bàn, đồng thời đồng bộ hóa dữ liệu người dùng giữa các thiết bị.
- **Tối ưu hóa cho di động:** Tối ưu hóa cho di động: Đảm bảo ứng dụng di động có hiệu suất tốt, giao diện thân thiện và tiết kiệm pin, mang lại trải nghiệm người dùng tối ưu trên các thiết bị di động.
- **Phân tích và đề xuất nhạc thông minh:** Phân tích và đề xuất nhạc thông minh: Sử dụng trí tuệ nhân tạo để phân tích thói quen nghe nhạc của người dùng và đề xuất các bài hát, playlist phù hợp, tạo sự cá nhân hóa trong trải nghiệm người dùng.
- **Đảm bảo tuân thủ bản quyền:** Đảm bảo tuân thủ bản quyền: Thiết lập các thỏa thuận với các nhà cung cấp nội dung và tuân thủ quy định về bản quyền âm nhạc để tránh các vấn đề pháp lý liên quan.

Chương 7

MÔI TRƯỜNG CHẠY ỨNG DỤNG VÀ CÁCH CÀI ĐẶT

7.1 Môi trường chạy ứng dụng

- **Backend - Django:** Sử dụng ngôn ngữ Python phiên bản 3.12.7, framework Django 5.1.6. Các gói phụ thuộc được liệt kê trong file `requirements.txt`.
- **Frontend - React:** Dựa trên ngôn ngữ JavaScript (ES6+), React phiên bản 18.3.1. Môi trường chạy gồm Node.js v20.16.0 và npm 10.8.1. Các gói phụ thuộc được khai báo trong file `package.json`.
- **Database - MySQL:** Sử dụng MySQL 8.0.30 và MySQL Workbench 8.0.30 để quản lý cơ sở dữ liệu.

7.2 Cách cài đặt

- Cài đặt Backend - Django:

```
git clone https://github.com/lamkbvn/BACKEND_SPOTIFY.git
cd BACKEND_SPOTIFY/my_project
pip install -r requirements.txt
python manage.py migrate
python manage.py runserver
```

- **Cài đặt Frontend - React:**

```
git clone https://github.com/duylam15/react-clone-spotify.git  
cd react-clone-spotify  
npm install  
npm run dev
```

- **Cài đặt Database - MySQL:** Nhập dữ liệu từ file `spotify.sql` vào cơ sở dữ liệu: