

**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 Lăng Phiêu

TP.HCM, 2025

# Mục lục

<b>Lời cảm ơn</b>	<b>3</b>
<b>1 GIỚI THIỆU</b>	<b>4</b>
1.1 Giới thiệu sơ lược về đề tài . . . . .	4
1.2 Lý do chọn đề tài . . . . .	4
1.3 Mục tiêu . . . . .	5
<b>2 ĐẶC TẢ YÊU CẦU PHẦN MỀM</b>	<b>6</b>
2.1 Yêu cầu chức năng . . . . .	6
2.1.1 Người dùng . . . . .	6
2.1.2 Quản lý . . . . .	7
2.2 Yêu cầu phi chức năng . . . . .	7
2.2.1 Hiệu suất . . . . .	7
2.2.2 Tính bảo mật . . . . .	8
2.2.3 Tính sẵn có . . . . .	8
2.2.4 Khả năng sử dụng . . . . .	8
2.2.5 Tính tương thích . . . . .	8
2.2.6 Tính bảo trì . . . . .	8
<b>3 KIẾN TRÚC PHẦN MỀM</b>	<b>9</b>
3.1 Thành phần phần mềm . . . . .	9
3.1.1 Hệ quản trị cơ sở dữ liệu: . . . . .	9
3.1.2 Giao thức truyền thông: . . . . .	10
3.2 Công cụ . . . . .	10
3.2.1 IntelliJ IDEA: . . . . .	10
3.2.2 Visual Studio Code (VS Code): . . . . .	10
3.2.3 Postman: . . . . .	10
3.2.4 MySQL Workbench: . . . . .	10
3.2.5 GitHub: . . . . .	11
3.2.6 Deploy: Amazon EC2 và VPS: . . . . .	11
3.3 Kiến trúc phần mềm . . . . .	11

3.4	Sử dụng LaTeX để viết báo cáo . . . . .	12
<b>4</b>	<b>CƠ SỞ DỮ LIỆU</b>	<b>13</b>

# 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 Lăng 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

rước 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.

## Chương 2

# ĐẶC TẢ YÊU CẦU PHẦN 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.
- **Khám phá nội dung:** Người dùng có thể xem các bài hát, album, và nghệ sĩ mới, cũng như tìm kiếm các bài hát theo thể loại.
- **Đánh giá nghệ sĩ:** Người dùng có thể đánh giá và để lại bình luận về các nghệ sĩ hoặc bài hát.

- **Quản lý thông tin cá nhân:** Người dùng có thể xem và chỉnh sửa thông tin cá nhân của mình.
- **Theo dõi nghệ sĩ:** Người dùng có thể theo dõi các nghệ sĩ yêu thích để nhận thông báo về các bài hát mới.
- **Thanh toán:** Người dùng có thể thanh toán cho các gói dịch vụ cao cấp qua các phương thức thanh toán trực tuyến.

### 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 nội dung:** Quản lý có thể thêm, sửa, xóa các bài hát, album, và danh sách phát, đồng thời quản lý các nghệ sĩ.
- **Quản lý danh sách phát chung:** Quản lý có thể tạo và duy trì các danh sách phát chung cho người dùng.
- **Thống kê:** Quản lý có thể thống kê số lượng người dùng, lượt nghe, và các bài hát phổ biến theo thời gian.
- **Quản lý hệ thống:** Quản lý có thể xem và quản lý thông tin của hệ thống, bao gồm các cấu hình và cài đặt.
- **Quản lý báo cáo:** Quản lý có thể tạo và xem các báo cáo chi tiết về hoạt động của người dùng và hệ thống.

## 2.2 Yêu cầu phi chức năng

### 2.2.1 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.



### 2.2.2 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.

### 2.2.3 Tính sẵn có

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

### 2.2.4 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.

### 2.2.5 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).

### 2.2.6 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 Thành phần phần mềm

#### 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.

#### 3.1.1 Hệ quản trị cơ sở dữ liệu:

- **MySQL:** MySQL là một hệ quản trị cơ sở dữ liệu mã nguồn mở, sử dụng SQL để truy vấn và thao tác với dữ liệu. Đây là lựa chọn phổ biến cho các ứng dụng web vì tính ổn định và khả năng tương thích với nhiều hệ điều hành khác nhau. Trong ứng dụng Clone Spotify, MySQL sẽ lưu trữ dữ liệu người dùng, danh sách bài hát, album, playlist, cũng như các dữ liệu liên quan đến người dùng và nhạc.

### 3.1.2 Giao thức truyền thông:

- **WebSocket:** WebSocket là một giao thức truyền thông giúp thiết lập kênh truyền thông hai chiều giữa máy chủ và máy khách. WebSocket cung cấp kết nối liên tục, giúp việc phát trực tiếp và truyền tải dữ liệu âm nhạc giữa máy chủ và người dùng hiệu quả, tối ưu hóa trải nghiệm nghe nhạc và giảm độ trễ khi phát nhạc.

## 3.2 Công cụ

### 3.2.1 IntelliJ IDEA:

- IntelliJ IDEA là IDE đầu tiên của JetBrains. Nó chủ yếu nhắm vào các công nghệ dựa trên JVM như Java, Groovy, Kotlin và Scala nhưng nó cũng hỗ trợ Java EE và phát triển web. Phiên bản đầu tiên của IntelliJ IDEA được phát hành vào tháng 1 năm 2001. Ở đây công cụ này được dùng để code spring boot API.

### 3.2.2 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 với ReactJS trong dự án Clone Spotify.

### 3.2.3 Postman:

- Postman là một trong những công cụ phổ biến nhất được sử dụng trong thử nghiệm các API. Như ta đã biết, API chịu trách nhiệm kết nối các ứng dụng với nhau, có Postman sẽ giúp cho thao tác với API này trở nên dễ dàng hơn. Thông thường, Postman sẽ được dùng cho API kiểu REST. Với Postman, ta có thể gọi Rest API mà không cần viết dòng code nào. Postman hỗ trợ tất cả các phương thức HTTP (GET, POST, PUT, PATCH, DELETE, ...). Bên cạnh đó, nó còn cho phép lưu lại lịch sử các lần request, rất tiện cho việc sử dụng lại khi cần.

### 3.2.4 MySQL Workbench:

- MySQL Workbench là công cụ giúp quản lý cơ sở dữ liệu MySQL, cho phép phát triển mô hình dữ liệu và quản lý cơ sở dữ liệu MySQL. Nó

cung cấp giao diện đồ họa cho phép dễ dàng tạo, chỉnh sửa cơ sở dữ liệu và thực hiện các thao tác như đảo ngược (reverse engineering) và chuyển tiếp (forward engineering) cơ sở dữ liệu.

### 3.2.5 GitHub:

- **GitHub** là công cụ quản lý mã nguồn phổ biến, cho phép các lập trình viên chia sẻ, cộng tác và quản lý phiên bản mã nguồn. Sự phát triển của nền tảng GitHub bắt đầu vào ngày 19 tháng 10 năm 2007. Trang web được đưa ra vào tháng 4 năm 2008 do Tom Preston-Werner, Chris Wanstrath, và P.J. Hyett. Microsoft đã mua GitHub vào tháng 6 năm 2018.

### 3.2.6 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

### WebSocket:

- **Khái niệm:** WebSocket là một giao thức truyền thông hai chiều giúp duy trì kết nối liên tục giữa máy khách và máy chủ. WebSocket rất hữu ích trong các ứng dụng yêu cầu giao tiếp thời gian thực, như ứng dụng Clone Spotify để truyền tải dữ liệu nhạc, thông báo, và các sự kiện trong thời gian thực mà không cần phải gửi lại các yêu cầu HTTP mới.

### Database Architecture:

- **MySQL:** MySQL sẽ được sử dụng để lưu trữ dữ liệu liên quan đến người dùng, bài hát, album, playlist, và các thông tin khác. Cơ sở dữ liệu này sẽ hỗ trợ các chức năng như:
  - Quản lý người dùng (đăng ký, đăng nhập, thông tin cá nhân).
  - Quản lý bài hát và album.
  - Quản lý playlist và bài hát yêu thích.

## 3.4 Sử dụng LaTeX để viết báo cáo

LaTeX là một hệ thống soạn thảo tài liệu mạnh mẽ, được sử dụng rộng rãi trong các lĩnh vực khoa học và kỹ thuật để tạo ra các tài liệu có chất lượng cao. Là 1 ngôn ngữ định dạng (document markup language) giống như Word, Powerpoint ở chỗ là đều cho ra những văn bản được định dạng theo mong muốn của người sử dụng nhưng có sự khác biệt lớn. Word thì gõ chữ cái và sử dụng công cụ đồ họa để thực hiện việc định dạng văn bản, còn LaTeX là "lập trình" ra văn bản.

- Tiền thân là TeX, một ngôn ngữ định dạng do Donald Knuth phát minh, rất khó sử dụng.
- LaTeX được phát minh bởi Leslie Lamport, dựa trên TeX nhưng dễ sử dụng hơn rất nhiều. Cho ra chất lượng bản in cực cao cùng cấu trúc văn bản rất logic và đồng bộ.
- Tự động hóa rất nhiều so với làm bằng tay của Word.

### Các bước cơ bản:

- Cài đặt phần mềm LaTeX như MiKTeX (Windows) hoặc TeXLive (Linux/MacOS).
- Cài đặt Overleaf (nếu bạn muốn sử dụng LaTeX trực tuyến).
- Sử dụng các môi trường như `document`, `itemize`, `section` để tạo cấu trúc cho tài liệu.
- Sử dụng `graphicx` để chèn hình ảnh, `hyperref` để chèn liên kết.
- Tạo bảng, biểu đồ hoặc đồ họa thông qua các gói tiện ích như `tabular`, `tikz` hoặc `pgfplots`.

# Chương 4

## CƠ SỞ DỮ LIỆU