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

Đồ án tổng hợp - CNPM (CO3103)

Báo cáo

Ứng dụng nghe nhạc BKSound

GVHD: Trần Trương Tuấn Phát

Sinh viên: Lư Chấn Vũ - 2313955 *(Lớp L04 - Nhóm 122, **Leader**)*
Vũ Minh Sang - 2312944 *(Lớp L04 - Nhóm 122)*
Nguyễn Quang Huy - 2311202 *(Lớp L01 - Nhóm 122)*
Lê Minh Khoa - 2311593 *(Lớp L04 - Nhóm 122)*
Lê Minh Trí - 2313593 *(Lớp L04 - Nhóm 122)*

TP. HỒ CHÍ MINH, 12/2024

Mục lục

Danh sách kí hiệu

$\mathbb{N}$ Tập hợp số tự nhiên

Danh sách từ viết tắt

CSP Cutting Stock Problem

FFD First Fit Decreasing

GA Genetic Algorithm

LP Linear Programming

Danh sách hình ảnh

Danh sách bảng biểu

Danh sách thành viên và nhiệm vụ

STT	Họ và tên	MSSV	Nhiệm vụ	% hoàn thành
1	Lư Chấn Vũ	2313955	- Code: GA. - Báo cáo: Mục 5.3.	100%
2	Vũ Minh Sang	2312944	- Báo cáo: Mục 3, 5.1. - Tổng hợp và chỉnh sửa báo cáo.	100%
3	Nguyễn Quang Huy	2311202	- Code: FFD. - Báo cáo: Mục 4.1, 5.2.	100%
4	Lê Minh Khoa	2311593	- Code: GA. - Báo cáo: Mục 2, 4.2.	100%
5	Lê Minh Trí	2313593	- Code: FFD. - Báo cáo: Mục 1, 6, 7.	100%

Bảng 1: Danh sách thành viên và nhiệm vụ

1 Tổng quan về đề tài

1.1 Mô tả đề tài

1.2 Mục tiêu đề tài

1.3 Phạm vi đề tài

2 Functional Requirements - Non-functional Requirements

2.1 Functional Requirements

2.1.1 Non-interactive Requirements

2.1.1.a Tính năng Streaming ở nhiều mức bitrate

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

- **Cách hoạt động:**
  - Khi Người dùng phát một bài hát, hệ thống sẽ cung cấp nhiều lựa chọn bitrate khác nhau (64kbps, 128kbps, 320kbps).
    - 64kbps: Phù hợp cho kết nối mạng yếu, tiết kiệm dữ liệu.
    - 128kbps: Chất lượng tiêu chuẩn, cân bằng giữa tốc độ tải và chất lượng âm thanh.
    - 320kbps: Chất lượng cao, dành cho Người dùng muốn trải nghiệm âm nhạc tốt nhất.
  - Người dùng có thể chọn thủ công mức bitrate mong muốn, phù hợp với chất lượng mạng và nhu cầu nghe nhạc.
  - Hệ thống hỗ trợ **Adaptive Streaming**: trong quá trình nghe, nếu mạng yếu hoặc không ổn định, bitrate sẽ tự động hạ xuống để tránh gián đoạn; khi mạng mạnh trở lại, bitrate được nâng lên để đảm bảo chất lượng âm thanh tốt nhất.
  - Các phiên bản nhạc ở nhiều bitrate được tạo sẵn trong quá trình xử lý upload, do đó việc chuyển đổi diễn ra nhanh chóng và mượt mà.
- **Input:**
  - File nhạc gốc (định dạng hợp lệ: MP3, WAV, FLAC,...).
  - Metadata bài hát (tên, nghệ sĩ, thể loại, ảnh bìa).
  - Thông tin cấu hình hệ thống (các mức bitrate cần tạo).
- **Output:**
  - Các phiên bản bài hát ở nhiều mức bitrate (64kbps, 128kbps, 320kbps).
  - Đường dẫn hoặc ID truy cập các file đã xử lý để phát trực tuyến.
- **Tiêu chí hoàn thành:**
  - Hệ thống tạo thành công ít nhất 3 mức bitrate cho mỗi bài hát đã upload.

  - Streaming ổn định, hỗ trợ adaptive bitrate khi tốc độ mạng thay đổi.
  - Các file nhạc được lưu trữ an toàn, có thể truy xuất đúng khi Người dùng phát bài hát.

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

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

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

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

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

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

- **Tiêu chí hoàn thành:**
  - Hệ thống luôn cập nhật gợi ý dựa trên các lượt nghe gần nhất (ví dụ: 20–30 bài gần đây).
  - Danh sách gợi ý chính xác, không trùng lặp quá nhiều với các bài đã nghe.
  - Tốc độ trả về gợi ý nhanh, không gây gián đoạn trải nghiệm Người dùng.
  - Cho phép cá nhân hoá theo Nghệ sĩ, thể loại hoặc mood.

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

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

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

- **Cách hoạt động:**
  - Hệ thống kiểm tra file ảnh cover kèm theo khi upload bài hát.
  - Nếu không có ảnh, hệ thống kích hoạt module sinh ảnh tự động.
  - Ảnh được tạo bằng cách:
    - Sinh ngẫu nhiên từ template mặc định theo thể loại nhạc.
    - Kết hợp text (tên bài hát, nghệ sĩ) với nền gradient hoặc ảnh mẫu.
  - Ảnh được gán cho bài hát và hiển thị trong thư viện, playlist, cũng như trình phát nhạc.
- **Input:**
  - File nhạc gốc upload lên hệ thống.
  - Metadata bài hát (tên, nghệ sĩ, thể loại).
  - Bộ template mặc định của hệ thống.
- **Output:**
  - Ảnh cover/thumbnail tự động sinh ra cho bài hát.
  - Đường dẫn hoặc ID của ảnh lưu trữ trên server.
- **Tiêu chí hoàn thành:**
  - Mỗi bài hát không có ảnh upload đều được gán một ảnh cover/thumbnail hợp lệ.
  - Ảnh có độ phân giải tối thiểu đáp ứng chuẩn hiển thị (ví dụ: 500x500px).
  - Quá trình sinh ảnh diễn ra nhanh, không làm chậm quá trình upload.
  - Ảnh đảm bảo tính thẩm mỹ và thông tin cơ bản (tên bài hát/nghệ sĩ nếu áp dụng).
- **Lợi ích:**
  - Đảm bảo giao diện ứng dụng đồng bộ, không có bài hát bị thiếu ảnh hiển thị.
  - Giúp Người dùng tiết kiệm thời gian chuẩn bị file ảnh trước khi upload.
  - Nâng cao trải nghiệm nghe nhạc với hình ảnh trực quan và thẩm mỹ.

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

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

- **Cách hoạt động:**

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

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

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

- **Tiêu chí hoàn thành:**
  - Không có tình trạng dừng nhạc đột ngột khi bài hát kết thúc (trừ khi danh sách rỗng và chế độ dừng được bật).
  - Chuyển bài mượt, độ trễ gần như bằng 0.
  - Tương thích với các chế độ phát (bình thường, lặp lại một bài, lặp lại danh sách, phát ngẫu nhiên).

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

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

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

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

- **Input:**
  - Danh sách Nghệ sĩ được Người dùng follow.

  - Dữ liệu phát hành bài hát/album mới của Nghệ sĩ.

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

- **Tiêu chí hoàn thành:**
  - Mỗi Nghệ sĩ được follow phát hành nhạc mới đều kích hoạt thông báo đến Người dùng.
  - Thông báo được gửi kịp thời, không chậm trễ quá lâu sau thời điểm phát hành.
  - Thông tin trong thông báo đầy đủ, rõ ràng, và dẫn đến đúng nội dung.
  - Người dùng có thể bật/tắt hoặc tùy chỉnh hình thức nhận thông báo.

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

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

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

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

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

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

- **Tiêu chí hoàn thành:**
  - Dữ liệu nghe nhạc được ghi nhận đầy đủ và chính xác theo từng Người dùng.

  - Thống kê có thể được lọc theo mốc thời gian (ngày/tuần/tháng/năm).
  - Giao diện hiển thị dễ hiểu, trực quan (ví dụ: biểu đồ tròn, biểu đồ cột).
  - Báo cáo tải nhanh, không gây chậm ứng dụng.

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

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

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

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

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

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

- **Tiêu chí hoàn thành:**
  - Lời bài hát được hiển thị đầy đủ, không thiếu dòng.
  - Đồng bộ chính xác với nhịp phát nhạc (sai lệch không quá 0.5 giây).
  - Giao diện hiển thị lyric mượt mà, không gây giật/lag.

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

2.2 Non-functional Requirements

3 Use-case Diagram và Use-case Scenario

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

Bảng 2: Mô tả usecase Upload bài hát/album

Tài liệu tham khảo

[1] Gilmore, P.C. and Gomory, R.E., 1961. A linear programming approach to the cutting stock problem. *Operations Research*, 9(6), pp.849-859.

[2] Dyckhoff, H., 1991. Cutting stock problems and solution procedures. *European Journal of Operational Research*, 44(2), pp.145-159.

[3] Cui, Z. and Zhang, L., 2000. A near-optimal solution to a two-dimensional cutting stock problem. *Journal of Optimization Theory and Applications*, 107(2), pp.393-408.

[4] Bennell, J.A., Oliveira, J.F. and Hitchen, M., 2021. Exact solution techniques for two-dimensional cutting and packing. *European Journal of Operational Research*, 293(3), pp.949-963.

[5] Lodi, A., Martello, S. and Vigo, D., 2021. A heuristic approach for two-dimensional rectangular cutting. *Computers & Operations Research*, 127, p.105121.

[6] Silva, R.A., Pinto, T.L. and Gomes, C.J., 2023. Approximation method for solving two-dimensional cutting stock problems. *Mathematical Programming*, 150(1), pp.195-220.

[7] Vanderbeck, F. and Wolsey, L.A., 1996. An exact algorithm for the two-dimensional cutting stock problem. *Computational Optimization and Applications*, 3(1), pp.123-143.

[8] Alvarez-Valdes, R., Parajon, A. and Tamarit, J.M., 2005. A tabu search algorithm for large-scale two-dimensional cutting stock problems. *Computers & Operations Research*, 32(5), pp.985-1007.

[9] Beasley, J.E., 1985. An exact two-dimensional non-guillotine cutting tree search procedure. *Operations Research*, 33(1), pp.49-64.

[10] Martello, S. and Toth, P., 1990. Knapsack Problems: Algorithms and Computer Implementations. Wiley-Interscience.

[11] GeeksforGeeks, 2024. Introduction to Greedy Algorithm - Data Structures and Algorithm Tutorials. Available at: https://www.geeksforgeeks.org/introduction-to-greedy-algorithm-data-structures-and-algorithm-tutorials/ [Accessed 25 Nov. 2024].

[12] OR-Library, Cutting Stock Problem Data. Available at: http://people.brunel.ac.uk/~mastjjb/jeb/orlib/cutinfo.html [Accessed 25 Nov. 2024].

[13] Dyckhoff, H., 1990. A typology of cutting and packing problems. *European Journal of Operational Research*, 44(2), pp.145-159.