

# NHẬP MÔN CÔNG NGHỆ PHẦN MỀM

## KIỂM TRA GIỮA KỲ

Thời gian 90 phút

### **Đề bài: Ứng dụng web phát nhạc / streaming nhạc (Music Streaming Web App)**

Xây dựng hệ thống nghe nhạc trực tuyến cho phép người dùng tìm kiếm/bật nhạc, tạo playlist, theo dõi nghệ sĩ, lưu yêu thích, xem lời bài hát và thống kê lượt nghe. Phát nhạc theo chuẩn HLS (HTTP Live Streaming) and DASH (Dynamic Adaptive Streaming over HTTP) (mô phỏng ở mức prototype).

#### **1) Chức năng cốt lõi (Functional Scope)**

- Đăng ký/đăng nhập (email/OTP hoặc social SSO).
- Tìm kiếm bài hát/nghệ sĩ/album/playlist.
- Phát nhạc (play/pause/seek/next/previous, shuffle/repeat).
- Trang bài hát: ảnh bìa, metadata, lời bài hát (lyrics), đề xuất liên quan.
- Playlist cá nhân: tạo/sửa/xóa, thêm bớt bài, chia sẻ (public/private).
- Theo dõi nghệ sĩ/album, “Yêu thích” bài hát (like/favorite).
- Lịch sử nghe (play history) + thống kê cơ bản (top 10 gần đây).
- Quản trị nội dung (admin): CRUD nghệ sĩ/album/bài hát/thể loại; upload file demo (mock) và đường dẫn streaming.
- (Tùy chọn) Gói thuê bao (subscription) và hạn ngạch/chất lượng.
- (Tùy chọn) Báo cáo lượt nghe theo ngày/tuần.

#### **2) Danh sách 10 thực thể & quan hệ (ER – Entity Relationship)**

1. **User**(UserID, Name, Email, PasswordHash, Role, CreatedAt, Status)
2. **Artist**(ArtistID, Name, Bio, Country, DebutYear, AvatarURL, Status)
3. **Album**(AlbumID, ArtistID, Title, ReleaseDate, CoverURL, Status)
4. **Track**(TrackID, AlbumID?, Title, Duration, AudioURL, Lyrics, GenreID, Explicit, PublishStatus)
5. **Genre**(GenreID, Name, Description).
6. **Playlist**(PlaylistID, OwnerUserID, Title, Description, Visibility[public/private], CoverURL, CreatedAt)
7. **PlaylistTrack**(PlaylistID, TrackID, SortOrder, AddedAt)
8. **Like**(UserID, TrackID, LikedAt)
9. **Follow**(UserID, ArtistID, FollowedAt)
10. **PlayHistory**(HistoryID, UserID, TrackID, PlayedAt, Device, PositionSec)

#### **Ghi chú:**

Tùy chọn (không bắt buộc trong 10 thực thể): **Subscription**(gói), **Payment**, **Report** (thống kê), **ContentSource** (nguồn file, CDN).

### 3) Yêu cầu sinh viên – Deliverables (10 mục)

#### 3.1 Thiết kế sơ đồ Use Case

- Diễn tả đầy đủ actor: *User, Admin, Audio CDN/Player (external)*.
- Nhóm UC chính: *Đăng nhập, Tìm kiếm, Phát nhạc, Quản lý Playlist, Yêu thích, Theo dõi Nghệ sĩ, Xem Lịch sử*.
- Nhóm UC quản trị: *Quản lý Artist/Album/Track/Genre*.
- Mỗi UC có *Brief Description + Pre/Post-condition + Basic/Alternate Flows* (bảng 1–2 trang).

#### 3.2 Thiết kế Sequence Diagram (tối thiểu 2 kịch bản)

- **SD-01: Phát nhạc một bài** – User → WebApp → Player → TrackService → (giả lập) Audio CDN.  
Thông điệp: *selectTrack(), getTrackMeta(), initStream(), play(), scrobble()*.
- **SD-02: Thêm bài vào Playlist** – User → WebApp → PlaylistService → TrackService → DB.  
Thông điệp: *addTrack(playlistId, trackId)*, kiểm tra trùng, cập nhật *SortOrder*.

#### 3.3 Thiết kế sơ đồ Thực thể–Kết hợp (ER)

- Vẽ diagram gồm **10 thực thể** ở phần (2), thể hiện đầy đủ cardinality (1–N, N–N), khóa chính/phụ, thuộc tính quan trọng, ràng buộc duy nhất (unique).

#### 3.4 Thiết kế sơ đồ CSDL (ERD – Logical/Physical)

- Chuyển ER sang **ERD quan hệ**: bảng, khóa, kiểu dữ liệu gợi ý (PostgreSQL/MySQL), chỉ mục (INDEX) cho cột tìm kiếm (*Track.Title, Artist.Name*), khóa tổng hợp cho bảng nối (*PlaylistTrack, Like, Follow*).
- Ghi chú ràng buộc: FK ON DELETE CASCADE với *PlaylistTrack/Like/Follow*; unique (*User.Email*), check (*Track.Duration > 0*).

#### 3.5 Thiết kế giao diện Prototype (wireframe/mockup)

- Tối thiểu 5 màn hình:  
A) **Home/Discover**, B) **Player mini + trang Track**, C) **Search**,  
D) **Playlist detail (drag-drop sắp xếp)**, E) **Artist/Album page**,  
    ◦ trang **Admin** (CRUD nội dung) ở mức khung.
- Dùng Figma/Photoshop/Whimsical hoặc HTML tĩnh (khuyến khích làm **prototype click-through**).

#### 3.6 Quy tắc nghiệp vụ & Test case

- Một bài hát chỉ xuất hiện **một lần** trong playlist (ràng buộc duy nhất).
- Lịch sử nghe được ghi khi **play**  $\geq 30$  **giây** hoặc *user presses next after  $\geq 30s$* .
- Test case mẫu (ít nhất 10 TC): thêm bài trùng playlist, like cùng track 2 lần, tìm kiếm không dấu/tiếng Việt, sắp xếp playlist, ghi nhận lịch sử.

### 3.7. Demo mô phỏng

- HTML prototype có **audio player** dùng `<audio>` hoặc HLS.js (mock HLS).
- Dữ liệu mẫu (seed):  $\geq 10$  nghệ sĩ,  $\geq 20$  bài hát,  $\geq 5$  album,  $\geq 5$  playlist.
- Ghi lại **video demo 3–5 phút** (screen record) minh họa các luồng chính.

### 3.8. Nộp báo cáo dưới dạng file HTML trên GitHub (bắt buộc)

- Tạo repo công khai **github.com/<username>/music-streaming-spec**.
- File **index.html** (hoặc GitHub Pages) tổng hợp:
  - Mục lục; ảnh Use Case, Sequence, ER, ERD; link Figma/Prototype; bảng API; test cases; video demo.
- Kèm **README.md** hướng dẫn chạy prototype (nếu có).

## Thang điểm:

- Use Case + mô tả luồng (15%)
- Sequence Diagrams (10%)
- ER + ERD (25%)
- Prototype UI (20%)
- Test cases (20%)
- Trình bày & GitHub HTML/Pages (10%)

---- Hết ----