

REQUIREMENT:

Project: Mini DLNA stream music (fangnhip)

Project leader: Nguyễn Thành Nam

Product Owner (PO): Nguyễn Thành Nam

Business Analyst (BA): Nguyễn Thành Nam

Backend Developer (Express): Nguyễn Thành Nam

Frontend Developer (React): Nguyễn Thành Nam

UX/UI Designer : Nguyễn Thành Nam

Tester: Nguyễn Thành Nam

Researcher: Nguyễn Thành Nam

1. Mục tiêu sản phẩm

- Khai thác thư viện nhạc từ MiniDLNA trong mạng LAN
- Hiển thị danh mục (tracks, album, artist, playlists, album art, encoding,...) cho người dùng
- Cho phép phát nhạc qua trình phát web (React front-end, Express back-end), Android (có thể là Flutter), với play/ pause / next / previous/ favorite/ volume/ repeat/ shuffle/ hiển thị lyrics
- MVP tập trung vào: scan nội dung DLNA, danh sách + tìm kiếm, phát nhạc, hiển thị metadata, trải nghiệm người dùng
- Có các chức năng: user có thể yêu cầu bài nhạc lossless để thêm vào DLNA (gửi form cho admin), người dùng có thể đóng góp nhạc lossless bằng cách gửi form cho admin kèm bài nhạc (có thể kèm lyrics(file .lrc hoặc .txt))

2. Chức năng chính

2.1 Khám phá & đồng bộ thư viện

- Kết nối tới miniDLNA server: đọc danh sách containers/items qua UPnP/ HTTP (browse/metadata)

- Lưu cache (database.json hoặc MongoDB) các metadata đã lấy (trackid, title, artist, album, duration, url, genre, nrAudioChannels, albumArtUrl, quality (encoding, label, Tier, bitDepth, sampleRate), date, composer, lyrics)

2.2 Duyệt & tìm kiếm

- List artists, album, playlists, tracks, quality, lyrics
- Lọc theo artists, tracks, albums, quality

2.3 Phát nhạc

- Làm nổi bật trình phát nhạc, tối ưu trải nghiệm nghe nhạc của người dùng (hiển thị thumbnail như 1 khung tranh bao kín màn hình (dành cho TV, tablet))

2.4 Metadata & album art

- Lấy album art từ DLNA

3. Non-functional requirements:

- Performance: Lần quét đầu sẽ hơi lâu, nhưng những lần sau tốc độ phản hồi để phát nhạc <1s, trải nghiệm mượt mà
- Scalability: Thư viện ban đầu là 400 tracks, có thể mở rộng lên nhiều
- Security: chỉ cho phép truy cập từ LAN, nếu ở ngoài vào thì phải dùng VPN
- Compatibility: Client sử dụng HTML5, backend hỗ trợ range request & CORS

4. Use case:

- US-1: là user, tôi muốn xem danh sách album để chọn phát
 - o AC: trả lại album với id, title, artist, albumArt, trackCount
- US-2: Là user, tôi muốn phát 1 track trên trình duyệt
 - o AC: audio tag có thể play, pause, repeats, shuffle, tua, tương tác volume