**Tài liệu Kỹ thuật – Ứng dụng MinCloset**

Phiên bản: 3.0

Ngày: 19 tháng 6 năm 2025

Tác giả: MinVN

**Mục lục**

1. **Giới thiệu**
   * 1.1. Mục đích Tài liệu
   * 1.2. Tổng quan Ứng dụng
2. **Kiến trúc Hệ thống**
   * 2.1. Tổng quan
   * 2.2. Quản lý Trạng thái: Riverpod
   * 2.3. Tầng Repository
   * 2.4. Tầng Use Case (Clean Architecture)
   * 2.5. Lưu trữ Dữ liệu
   * 2.6. Cấu trúc Thư mục
3. **Phân tích Chức năng Chi tiết**
   * 3.1. Quản lý Tủ đồ & Vật phẩm
   * **3.2. Thêm đồ hàng loạt & Phân loại bằng AI (Tính năng mới)**
   * 3.3. Gợi ý Trang phục bằng AI
   * 3.4. Xưởng Phối đồ & Trình duyệt Vật phẩm
   * 3.5. Thư viện Bộ đồ đã lưu
   * 3.6. Trang Cá nhân & Cài đặt
4. **Các Thành phần Cốt lõi**
   * 4.1. Lớp Mô hình Dữ liệu (Data Models)
   * 4.2. Lớp Tương tác CSDL (Database Helper)
   * 4.3. Widget Tái sử dụng
5. **Hướng dẫn Cài đặt & Môi trường Dev**
6. **Lộ trình Phát triển & Đề xuất**

**1. Giới thiệu**

**1.1. Mục đích Tài liệu**

Tài liệu này cung cấp một cái nhìn tổng quan chi tiết về kiến trúc kỹ thuật, các chức năng, và phương pháp xây dựng của ứng dụng MinCloset1. Mục tiêu là làm cơ sở cho việc bảo trì, mở rộng và chuyển giao dự án2.

**1.2. Tổng quan Ứng dụng**

MinCloset là ứng dụng di động đa nền tảng được xây dựng bằng Flutter, hoạt động như một trợ lý tủ đồ thông minh3. Ứng dụng cho phép người dùng số hóa tủ quần áo, tạo các bộ trang phục ảo, và nhận gợi ý phối đồ hàng ngày được cá nhân hóa bởi A.I4.

**2. Kiến trúc Hệ thống**

**2.1. Tổng quan**

Ứng dụng tuân thủ các nguyên tắc của Clean Architecture để tối ưu cho việc bảo trì, kiểm thử và mở rộng5. Luồng dữ liệu chính tuân theo mô hình:

UI (Widget) → Logic Giao diện (Notifier) → Logic Nghiệp vụ (Use Case) → Repository → Nguồn Dữ liệu (Data Source) 6

**2.2. Quản lý Trạng thái (State Management): Riverpod**

* **Công nghệ**: Riverpod được sử dụng làm giải pháp quản lý trạng thái chính7.
* **Triển khai**:
  + StateNotifierProvider: Dùng để quản lý các trạng thái phức tạp và tương tác của người dùng (ví dụ: HomePage, AddItemPage, OutfitBuilderPage)8.
  + FutureProvider: Dùng để cung cấp dữ liệu bất đồng bộ một lần (ví dụ: closetsProvider)9.
  + **StateProvider**: Dùng cho các trạng thái đơn giản, toàn cục. Ví dụ tiêu biểu là mainScreenIndexProvider được dùng để điều khiển tab đang được chọn của thanh điều hướng chính từ bất kỳ đâu trong ứng dụng.
  + .family modifier: Cho phép tạo các provider có tham số, tạo ra các state độc lập (ví dụ: itemFilterProvider)10.
  + .autoDispose modifier: Tự động giải phóng bộ nhớ cho các provider không cần tồn tại lâu dài11.
* **Nguyên tắc**: Giao diện (Widgets) hoàn toàn mang tính khai báo 12, chỉ "theo dõi" (ref.watch) trạng thái và "gọi" (ref.read) các phương thức từ Notifier13. Logic được đặt trong Notifier hoặc Use Case14.

**2.3. Tầng Repository**

Tầng này tạo một lớp trừu tượng giữa logic nghiệp vụ và nguồn dữ liệu (SQLite, API...), cho phép thay đổi nguồn dữ liệu mà không ảnh hưởng đến logic15. Mỗi domain có một Repository riêng (ClosetRepository, ClothingItemRepository,...)16.

**2.4. Tầng Use Case (Clean Architecture)**

Đóng gói các quy trình nghiệp vụ phức tạp, đặc biệt là những quy trình cần điều phối nhiều Repository17.

* GetOutfitSuggestionUseCase: Điều phối việc lấy thời tiết, vật phẩm và gọi AI để tạo gợi ý18.
* SaveOutfitUseCase: Xử lý logic chụp ảnh màn hình và lưu một bộ đồ19.
* **AnalyzeItemUseCase (Mới)**: Đóng gói logic nghiệp vụ cho việc gọi AI để phân tích và phân loại một hình ảnh vật phẩm.

**2.5. Lưu trữ Dữ liệu**

* **CSDL Chính**: SQLite được sử dụng thông qua gói sqflite cho các dữ liệu có cấu trúc20.
* **Lưu trữ Đơn giản**: shared\_preferences được dùng cho các dữ liệu nhỏ, cài đặt người dùng21.

**2.6. Cấu trúc Thư mục (lib)**

Cấu trúc được tổ chức theo chức năng và tầng kiến trúc: screens, models, states, notifiers, providers, domain/use\_cases, repositories, services, widgets, helpers, constants22.

**3. Phân tích Chức năng Chi tiết**

**3.1. Quản lý Tủ đồ & Vật phẩm**

* **Mô tả**: Cho phép người dùng quản lý tủ đồ và vật phẩm23.
* **Thành phần Kỹ thuật**:
  + UI: ClosetsPage với 2 tab "Tất cả vật phẩm" và "Theo Tủ đồ"24. ClosetDetailPage hiển thị các vật phẩm trong một tủ đồ25. Nút + trên AppBar của ClosetDetailPage đã được loại bỏ để tránh trùng lặp chức năng.
  + Điều hướng: Nút "Xem tất cả" ở trang chủ giờ đây sử dụng mainScreenIndexProvider để tự động chuyển người dùng đến tab "Tủ đồ".

**3.2. Thêm đồ hàng loạt & Phân loại bằng AI (Tính năng mới)**

* **Mô tả**: Đây là một luồng chức năng lớn và mới, cho phép người dùng thêm nhiều vật phẩm cùng lúc và được hỗ trợ bởi AI.
* **Thành phần Kỹ thuật**:
  + **Lối vào**: Nút + toàn cục (GlobalAddButton) giờ đây sẽ hiển thị một menu lựa chọn: "Chụp ảnh mới" hoặc "Chọn từ Album".
  + **Chọn nhiều ảnh**: Nếu người dùng chọn "Chọn từ Album", ứng dụng sẽ gọi pickMultiImage, giới hạn ở **10 ảnh**. Nếu người dùng chọn nhiều hơn, một SnackBar sẽ thông báo và chỉ 10 ảnh đầu tiên được xử lý.
  + **Phân loại tự động**:
    - **Kích hoạt**: Ngay sau khi người dùng chọn ảnh (dù là một hay nhiều), ảnh sẽ được gửi đến Gemini 1.5 Flash thông qua AnalyzeItemUseCase.
    - **Logic**: AI được hướng dẫn bởi một prompt chi tiết để phân tích ảnh và trả về một đối tượng JSON.
    - **Kết quả**: AI có khả năng tự động điền các trường:
      * **Danh mục**: Phân loại theo cấu trúc 2 tầng (ví dụ: "Áo > Áo thun"). Nếu không nhận diện được danh mục con, nó sẽ trả về "Khác" (ví dụ: "Áo > Khác").
      * **Màu sắc**: Có thể nhận diện và chọn nhiều màu.
      * **Chất liệu & Họa tiết**: Chọn một giá trị phù hợp nhất hoặc trả về "Khác".
    - **UI**: Trong lúc AI phân tích, một lớp phủ loading sẽ hiển thị trên ảnh vật phẩm.
  + **Giao diện Thêm hàng loạt**:
    - Nếu người dùng chọn nhiều hơn 1 ảnh, họ sẽ được điều hướng đến màn hình BatchAddItemScreen.
    - Màn hình này sử dụng PageView cho phép lướt qua lại giữa các món đồ.
    - Tiêu đề AppBar hiển thị tiến trình (ví dụ: "Thêm đồ (2/10)").
    - Nút "Sau" ở thanh điều hướng dưới cùng sẽ tự động chuyển thành nút "Lưu tất cả" khi người dùng ở món đồ cuối cùng.
  + **Tái cấu trúc**: Form nhập liệu được tách thành widget tái sử dụng ItemDetailForm. AddItemScreen (thêm một món đồ) và BatchAddItemScreen đều sử dụng widget này để đảm bảo sự nhất quán.

**3.3. Gợi ý Trang phục bằng AI**

Chức năng không thay đổi về mặt cốt lõi. Logic vẫn được đóng gói trong GetOutfitSuggestionUseCase26.

**3.4. Xưởng Phối đồ & Trình duyệt Vật phẩm**

* **Mô tả**: Canvas sáng tạo để tạo bộ đồ ảo27.
* **Thành phần Kỹ thuật**:
  + **UI**:
    - Layout của DraggableScrollableSheet đã được sửa lỗi bằng CustomScrollView và SliverPersistentHeader, cho phép kéo từ mọi vị trí và "ghim" thanh tìm kiếm.
    - Tỉ lệ khung hình của canvas phối đồ đã được **cố định là 3:4** và có **nền trắng**, đảm bảo sự nhất quán với ảnh thumbnail và trang chi tiết.
    - **Badge số đếm (Mới)**: Các vật phẩm trong panel lựa chọn giờ đây sẽ hiển thị một ô tròn số đếm ở góc, cho biết vật phẩm đó đã được thêm vào canvas bao nhiêu lần.

**3.5. Thư viện Bộ đồ đã lưu**

* **Mô tả**: Trang OutfitsHubPage hiển thị thư viện các bộ đồ đã tạo28.
* **Thành phần Kỹ thuật**: GridView hiển thị các bộ đồ với tỉ lệ 3:4, nhất quán với "Xưởng phối đồ"29. Trang chi tiết bộ đồ cũng có nền trắng để đảm bảo tính đồng bộ.

**3.6. Trang Cá nhân & Cài đặt**

Chức năng không thay đổi đáng kể so với phiên bản trước30.

**4. Các Thành phần Cốt lõi**

* **Data Models** (lib/models/): Các lớp ClothingItem, Closet, Outfit, OutfitFilter định nghĩa cấu trúc dữ liệu31. AddItemState đã được thêm cờ isAnalyzing. Lớp BatchAddItemState mới được tạo để phục vụ tính năng thêm hàng loạt.
* **Database Helper** (lib/helpers/db\_helper.dart): Lớp cấp thấp thực thi các câu lệnh SQL32. Đã được bổ sung phương thức insertBatchItems để tối ưu hiệu năng khi lưu nhiều vật phẩm.
* **Reusable Widgets** (lib/widgets/): ItemBrowserView đã được nâng cấp với buildMode để có thể hiển thị linh hoạt dưới dạng Box hoặc Sliver. ItemDetailForm là một widget tái sử dụng mới, được tách ra từ AddItemScreen.

**5. Hướng dẫn Cài đặt & Môi trường Dev**

Các yêu cầu không thay đổi so với trước33.

**6. Lộ trình Phát triển & Đề xuất**

Kiến trúc hiện tại đã rất vững chắc và sẵn sàng cho các tính năng phức tạp hơn.

* **Các bước tiếp theo đã định sẵn**: "Lịch mặc đồ", "Đồng bộ Đám mây", "Thống kê Nâng cao" vẫn là những mục tiêu rất giá trị34.
* **Các ý tưởng mở rộng mới**:
  + **Gợi ý Mua sắm thông minh**: Dựa trên tủ đồ hiện tại, AI có thể gợi ý các món đồ nên mua để tối ưu hóa khả năng phối đồ.
  + **Tương tác Cộng đồng**: Cho phép người dùng chia sẻ bộ đồ của mình và tham gia các thử thách phối đồ.
  + **Đóng gói cho Chuyến đi (Packing Lists)**: Gợi ý trang phục cần mang theo dựa trên điểm đến và dự báo thời tiết.