

MINI-SRS - Hệ thống Quản Lý Chi Tiêu Junkio (Personal & Family Expense Management)

0) Thông tin tài liệu

- Tên dự án: Web App Quản Lý Chi Tiêu Cá Nhân & Gia Đình
 - Kiến trúc bắt buộc: **Client–Server**, FE/BE tách biệt; FE dạng **SPA**; BE dạng **RESTful API trả JSON**
 - CSDL bắt buộc: **RDBMS** (MySQL/SQL Server/PostgreSQL), dùng **Migrations**; có **Seeder/Factory** tạo dữ liệu giả lập (tối thiểu **500–1000 bản ghi chính**)
 - Deploy bắt buộc: **Docker + docker-compose.yml** chạy toàn bộ dự án bằng **1 câu lệnh**
-

1) Scope (Phạm vi)

1.1 Mục tiêu

Hệ thống giúp người dùng:

- Ghi nhận và quản lý **thu/chi** theo thời gian, danh mục, ví (cá nhân/nhóm gia đình).
- Theo dõi **ngân sách** (budget) và **báo cáo – thống kê** bằng dashboard & biểu đồ.
- Hỗ trợ **logic nghiệp vụ phức tạp** (vượt CRUD), ví dụ: giao dịch định kỳ, đơn giản hóa nợ trong nhóm.

1.2 In-scope

- Xác thực & phân quyền (RBAC).
- CRUD cho các đối tượng chính (transactions, wallets, categories, budgets, families...).
- Tìm kiếm/ lọc/ sắp xếp/ phân trang danh sách (bắt buộc).
- Dashboard thống kê + tối thiểu 2 biểu đồ.
- 1 mô-đun logic phức tạp “định” (chọn triển khai thật): **Recurring Engine** hoặc **Debt Simplification**.

1.3 Out-of-scope

- Tích hợp ngân hàng/Open Banking, OCR hoá đơn
-

2) Roles (Vai trò) & quyền hạn (RBAC)

Ánh xạ theo ngữ cảnh quản lý chi tiêu gia đình:

R1 - System Admin (tương ứng Admin)

- Quyền: quản trị hệ thống, quản lý user (khóa/mở), cấu hình tham số hệ thống, xem thống kê tổng quan (không can thiệp dữ liệu tài chính nếu không cần).

R2 - Family Manager (tương ứng Staff)

- Quyền: tạo “Family Group”, mời thành viên, thiết lập ngân sách tổng, phê duyệt chi vượt hạn mức, xem báo cáo toàn bộ gia đình.

R3 - Family Member (tương ứng Customer)

- Quyền: ghi giao dịch cá nhân/chung, xem báo cáo cá nhân, xem ngân sách chung ở chế độ read-only, đề xuất khoản chi lớn.
-

3) Danh sách yêu cầu (FR/NFR)

3.1 Functional Requirements (FR)

Nhóm FR-AUTH (Xác thực & tài khoản)

- FR-AUTH-01:** Đăng nhập/đăng xuất.
- FR-AUTH-02:** Quên mật khẩu (gửi email reset).
- FR-AUTH-03:** JWT cho API; yêu cầu token cho các API protected.

Nhóm FR-RBAC (Phân quyền)

- FR-RBAC-01:** Hệ thống có ≥ 3 roles.
- FR-RBAC-02:** Admin có thể gán quyền/role cho Staff (Family Manager).
- FR-RBAC-03:** Middleware kiểm tra quyền truy cập theo role (các endpoint admin/manager/member khác nhau).

Nhóm FR-CORE (Nghiệp vụ chính – CRUD)

- FR-CORE-01:** CRUD (thêm/xem/sửa/xóa hoặc ẩn) cho các đối tượng chính.
- FR-CORE-02:** Quản lý **Transactions (giao dịch)**: amount, date, category, wallet, mô tả... (đối tượng trung tâm).
- FR-CORE-03:** Quản lý **Wallets (ví)**: ví riêng (user) / ví chung (family).

- **FR-CORE-04:** Quản lý **Categories (danh mục)**: income/expense; có thể hỗ trợ danh mục con.
- **FR-CORE-05:** Quản lý **Budgets (ngân sách)** theo khoảng thời gian và (tuỳ chọn) theo danh mục.
- **FR-CORE-06:** Quản lý **Families & FamilyMembers**: tạo nhóm gia đình, mời/join, trạng thái pending/active.

Nhóm FR-SEARCH (Tìm kiếm, lọc, phân trang)

- **FR-SEARCH-01:** Danh sách giao dịch có **lọc đa tiêu chí** (ngày, danh mục, ví...), **sắp xếp**, và **phân trang**.
- **FR-SEARCH-02:** Trang quản trị user / danh mục / ví... cũng hỗ trợ phân trang khi danh sách lớn (phục vụ seed 500–1000 records).

Nhóm FR-DASH (Dashboard & báo cáo)

- **FR-DASH-01:** Dashboard tổng hợp số liệu (ví dụ tổng thu, tổng chi theo tháng).
- **FR-DASH-02:** Tối thiểu **2 biểu đồ** (cột/tròn/đường...) từ dữ liệu tổng hợp.

Nhóm FR-COMPLEX (Logic phức tạp)

- **FR-COMPLEX-01:** Có ít nhất 1 chức năng xử lý logic nghiệp vụ “không chỉ nhập-xuất”.
Sẽ thực hiện 1 trong 2 (hoặc cả 2 nếu kíp):
 - **Option A -Recurring Transaction Engine:** tự tạo giao dịch cho khoản chi định kỳ theo frequency + ngày bắt đầu (scheduler/cron).
 - **Option B -Debt Simplification:** tính “dòng tiền ròng” và đề xuất giao dịch tối thiểu để thanh toán nợ trong nhóm.

Nhóm FR-BONUS

- **FR-BONUS-01:** Real-time notification/chat (Socket.io/SignalR).
- **FR-BONUS-02:** Export/Import Excel/PDF.

3.2 Non-Functional Requirements (NFR)

NFR-TECH (Kiến trúc & công nghệ)

- **NFR-TECH-01:** FE/BE tách biệt; FE là SPA; BE là REST API trả JSON.

- **NFR-TECH-02:** RDBMS + migrations; không nộp file .sql export thủ công thay cho migrations.
- **NFR-TECH-03:** Seeder/Factory tạo dữ liệu giả lập tối thiểu 500–1000 bản ghi chính.
- **NFR-TECH-04:** Docker hoá dự án; có docker-compose chạy App + DB bằng 1 lệnh.

NFR-UI (UI/UX)

- **NFR-UI-01:** Responsive (desktop & mobile/tablet).
- **NFR-UI-02:** Validation frontend (bắt lỗi nhập liệu); thông báo rõ ràng khi thành công/thất bại (toast/alert).
- **NFR-UI-03:** Giao diện hiện đại, bố cục rõ ràng; khuyến khích dùng UI Library.

NFR-SEC (Bảo mật)

- **NFR-SEC-01:** JWT; mật khẩu không lưu plain text, phải hash (bcrypt/argon2).
- **NFR-SEC-02:** Chống SQL Injection (ORM/prepared statements).
- **NFR-SEC-03:** Chống XSS khi hiển thị dữ liệu người dùng nhập.
- **NFR-SEC-04:** Validation chặt chẽ ở backend (không chỉ tin frontend).

NFR-PROCESS (Quy trình & chất lượng mã)

- **NFR-PROC-01:** Quản lý mã nguồn bằng Git trên GitHub/GitLab; commit lịch sử rõ ràng; có nhánh Main/Dev.
- **NFR-PROC-02:** Tuân thủ coding convention; tổ chức code theo MVC/cấu trúc framework.
- **NFR-PROC-03:** Không lộ thông tin nhạy cảm; dùng .env.

4) Business Rules (Quy tắc nghiệp vụ)

- **BR-01 (Wallet ownership):** Một ví phải thuộc về **User hoặc Family**, không được “cả hai” hoặc “không thuộc ai”.
- **BR-02 (Budget evaluation):** Budget không FK trực tiếp với transactions; tổng chi trong khoảng thời gian + danh mục sẽ được so sánh với hạn mức.

- **BR-03 (Recurring):** Giao dịch định kỳ được tạo tự động theo frequency/next_run_date; hệ thống chạy scheduler kiểm tra hằng ngày.
 - **BR-04 (Debt simplification):** Hệ thống tính net flow để đề xuất danh sách thanh toán tối thiểu thay vì nhiều giao dịch vòng.
 - **BR-05 (RBAC enforcement):** Family Member không được thay đổi cấu trúc ngân sách tổng; Family Manager có quyền thiết lập/phê duyệt.
-

5) Glossary (Thuật ngữ)

- **SPA:** Single Page Application (ứng dụng 1 trang, chuyển trang bằng routing phía client).
- **RESTful API:** API theo chuẩn REST, dùng HTTP methods (GET/POST/PUT/DELETE), trả JSON.
- **RBAC:** Role-Based Access Control (phân quyền theo vai trò).
- **JWT:** JSON Web Token, dùng cho xác thực/ủy quyền API.
- **Migration:** Script quản lý phiên bản schema CSDL (tạo/sửa bảng theo version).
- **Seeder/Factory:** Sinh dữ liệu giả phục vụ test/phân trang/hiệu năng.
- **Transaction:** Giao dịch thu/chi (đối tượng trung tâm).
- **Wallet:** Ví tiền (cá nhân hoặc gia đình).
- **Budget:** Ngân sách giới hạn theo thời gian/danh mục.
- **Recurring Pattern:** Mẫu định kỳ để tạo giao dịch tự động.