# BÁO CÁO THIẾT KẾ CƠ SỞ DỮ LIỆU DỰ ÁN ZENDO

* Tên đề tài: ZenDo – Focus on what truly matters
* Sinh viên thực hiện: Nguyễn Bình Minh
* Giảng viên hướng dẫn: Nguyễn Hữu Trung
* Trường: VD

TP. Hồ Chí Minh, tháng 10 năm 2025

# 1. Giới thiệu tổng quan hệ thống

ZenDo là ứng dụng quản lý công việc cá nhân và nhóm, được phát triển bằng Flutter (frontend) và Supabase (backend – cơ sở dữ liệu PostgreSQL, có tích hợp realtime, auth, và storage). Ứng dụng giúp người dùng quản lý dự án, tạo nhiệm vụ, đặt nhắc nhở, gắn thẻ, bình luận và chia sẻ trong nhóm. Hệ thống cơ sở dữ liệu được thiết kế hướng đối tượng, hỗ trợ AI phân tích thói quen và sinh insight làm việc hiệu quả.

# 2. Thiết kế cơ sở dữ liệu

Hệ thống ZenDo sử dụng PostgreSQL làm cơ sở dữ liệu chính, được triển khai qua Supabase. Tất cả dữ liệu được lưu trong schema `public`, với tổng cộng 13 bảng chính cùng các trigger và chính sách bảo mật (RLS).

## 2.1 Danh sách các bảng trong hệ thống

* 1. profiles
* 2. projects
* 3. project\_members
* 4. tasks
* 5. subtasks
* 6. task\_tags
* 7. task\_tag\_map
* 8. attachments
* 9. comments
* 10. reminders
* 11. notifications
* 12. ai\_insights
* 13. v\_membership (view hỗ trợ policy)

## 2.2 Mô tả chi tiết các bảng

### Bảng: profiles

Lưu thông tin người dùng, liên kết trực tiếp với hệ thống xác thực (auth.users) của Supabase.

### Bảng: projects

Lưu thông tin dự án của người dùng. Mỗi người dùng có thể tạo nhiều dự án.

### Bảng: project\_members

Liên kết giữa người dùng và dự án (thành viên). Mỗi dự án có thể có nhiều thành viên với vai trò khác nhau.

### Bảng: tasks

Chứa thông tin từng nhiệm vụ (task) thuộc dự án.

### Bảng: subtasks

Chứa các công việc con thuộc một task. Mỗi task có thể có nhiều subtasks.

### Bảng: task\_tags

Lưu thông tin các thẻ (tag) trong mỗi dự án, phục vụ gắn nhãn cho task.

### Bảng: task\_tag\_map

Bảng liên kết n-n giữa task và tag.

### Bảng: attachments

Lưu thông tin tệp đính kèm của task, như hình ảnh, tài liệu, hoặc file âm thanh.

### Bảng: comments

Lưu bình luận của người dùng trong mỗi task. Khi có comment mới, trigger sẽ tạo thông báo cho các thành viên khác.

### Bảng: reminders

Lưu các nhắc nhở cho task, dùng để gửi thông báo hoặc email đến người dùng khi đến hạn.

### Bảng: notifications

Lưu thông tin thông báo gửi đến người dùng khi có sự kiện (comment mới, task hoàn thành...).

### Bảng: ai\_insights

Lưu kết quả phân tích AI: ví dụ gợi ý ưu tiên công việc, tóm tắt ngày làm việc, hoặc phân tích thói quen.

### Bảng: v\_membership

View hỗ trợ xác định user nào thuộc project nào. Dùng trong chính sách RLS.

# 3. Mối quan hệ giữa các bảng

- profiles 1–n projects  
- projects 1–n tasks  
- tasks 1–n subtasks, 1–n comments, 1–n attachments  
- projects n–n profiles thông qua project\_members  
- tasks n–n tags thông qua task\_tag\_map

# 4. Trigger và Row Level Security (RLS)

Supabase sử dụng RLS để bảo mật từng hàng dữ liệu. Người dùng chỉ có thể truy cập các bản ghi thuộc về họ hoặc dự án mà họ là thành viên. Các policy quan trọng gồm:  
- profiles: chỉ chính chủ có quyền đọc/sửa  
- projects: owner tạo, sửa, xóa; members chỉ xem  
- tasks: member có thể đọc, editor/owner được phép cập nhật hoặc xóa

Trigger quan trọng:  
- handle\_new\_user(): tự động tạo profile khi có user mới  
- set\_updated\_at(): tự động cập nhật thời gian sửa task  
- notify\_comment(): tạo thông báo cho các thành viên khi có bình luận mới

# 5. Kết nối Flutter và Supabase

Ứng dụng Flutter sử dụng gói `supabase\_flutter` để kết nối với cơ sở dữ liệu này. Các chức năng chính bao gồm đăng nhập, CRUD dữ liệu, realtime và upload file.

# 6. Kết luận và hướng phát triển

Thiết kế cơ sở dữ liệu ZenDo đảm bảo tính mở rộng, bảo mật và tương thích cao. Cấu trúc này đủ để xây dựng một hệ thống quản lý công việc hoàn chỉnh, đồng bộ realtime, và dễ mở rộng với các module AI sau này (gợi ý, phân tích thói quen, nhắc nhở thông minh...).