**1. Kiến Trúc Phần Mềm**

**🔹 Mô hình kiến trúc:**

Sử dụng **kiến trúc Microservices** để đảm bảo tính linh hoạt, dễ bảo trì và mở rộng.

**🏛️ Tổng quan kiến trúc:**

➡️ **Frontend:**

* Sử dụng công cụ như **Visual Code, Intellij, MySQL**
* Ngôn ngữ lập trình sử dụng html, css, javascript

➡️ **Backend:**

* **Node.js (Express.js)** hoặc **Python (Django)** → Xử lý logic nghiệp vụ
* **Socket.IO** hoặc **Firebase** → Xử lý thông báo thời gian thực
* **JWT (JSON Web Token)** → Xác thực và phân quyền người dùng
* **Worker Service** → Xử lý nền cho các công việc phức tạp (cron job)

**2. Thiết Kế Giao Diện (UI/UX)**

**🔹 Nguyên tắc thiết kế:**

✅ Đơn giản, dễ sử dụng, trực quan  
✅ Ưu tiên UX (trải nghiệm người dùng) trên các thiết bị di động và máy tính

**🖥️ 2.1. Màn Hình Đăng Nhập**

* Trường nhập: Tên đăng nhập, mật khẩu
* Chức năng:
  + Đăng nhập qua OAuth (Google, Microsoft)
  + Quên mật khẩu
* Thông báo: Thông báo lỗi khi đăng nhập sai

**🏠 2.2. Trang Chủ**

➡️ Hiển thị các thông tin:

* Tiến độ dự án (theo phần trăm, màu sắc)
* Số lượng công việc hoàn thành
* Danh sách các vấn đề cần giải quyết

➡️ Các thành phần:  
✅ Biểu đồ Gantt → Tiến độ theo thời gian  
✅ Biểu đồ thanh → Hiệu quả sử dụng tài nguyên  
✅ Thông báo real-time → Dùng WebSocket

**📋 2.3. Quản Lý Công Việc**

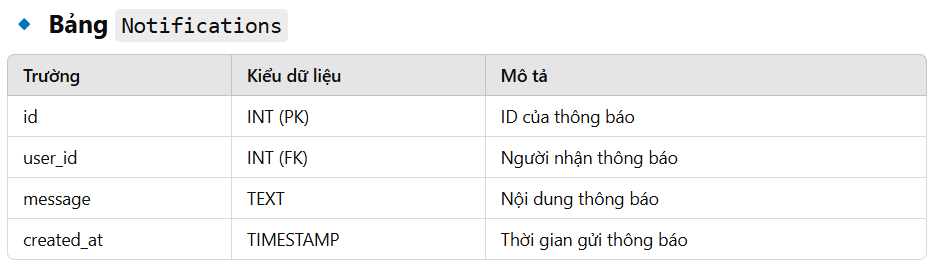
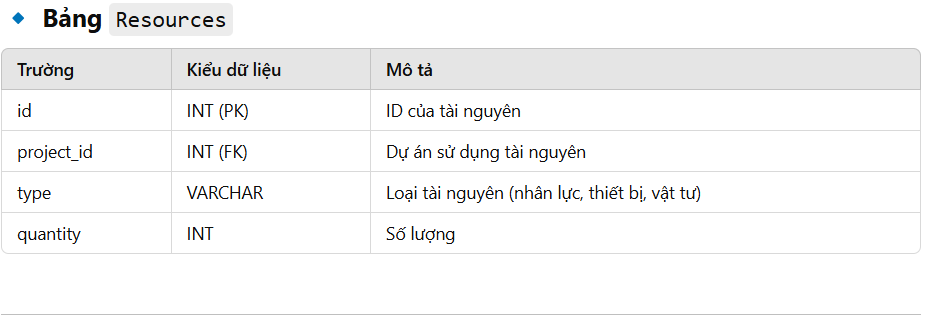
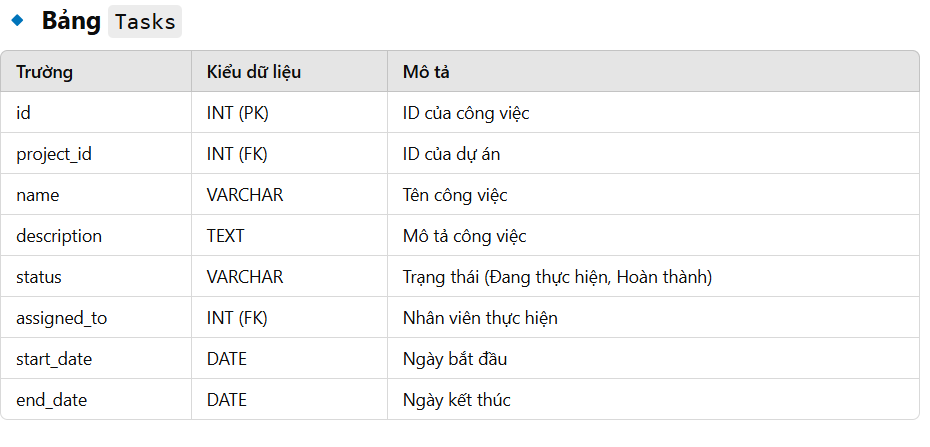
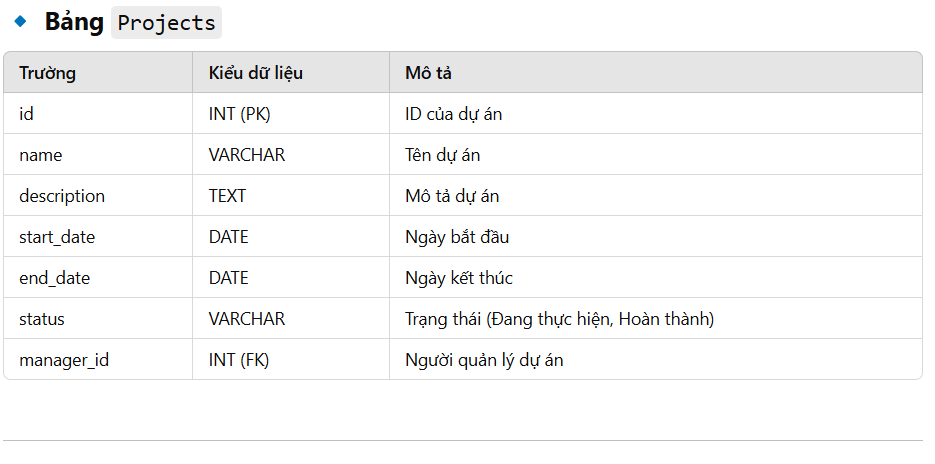
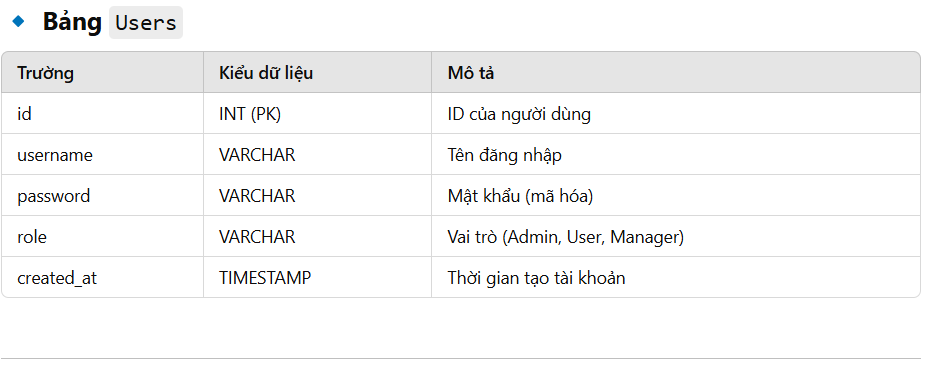
* Danh sách các công việc đang thực hiện

**📊 2.4. Báo Cáo**

* Báo cáo tiến độ
* Báo cáo hiệu suất sử dụng tài nguyên
* Xuất báo cáo PDF/Excel

**🔔 2.5. Hệ Thống Thông Báo**

* Thông báo real-time khi có công việc mới hoặc tiến độ bị chậm
* Hỗ trợ gửi qua email và ứng dụng

**3. Mô Hình Cơ Sở Dữ Liệu**