



**Học viện**  
**Công nghệ Bưu chính Viễn thông**

# Hệ thống bảng trắng Kanva

Trình bày: Phạm Minh Dương – B24CHHT064

Ma Công Thành – NCS2024.20

Vũ Quỳnh Anh – B24CHHT056

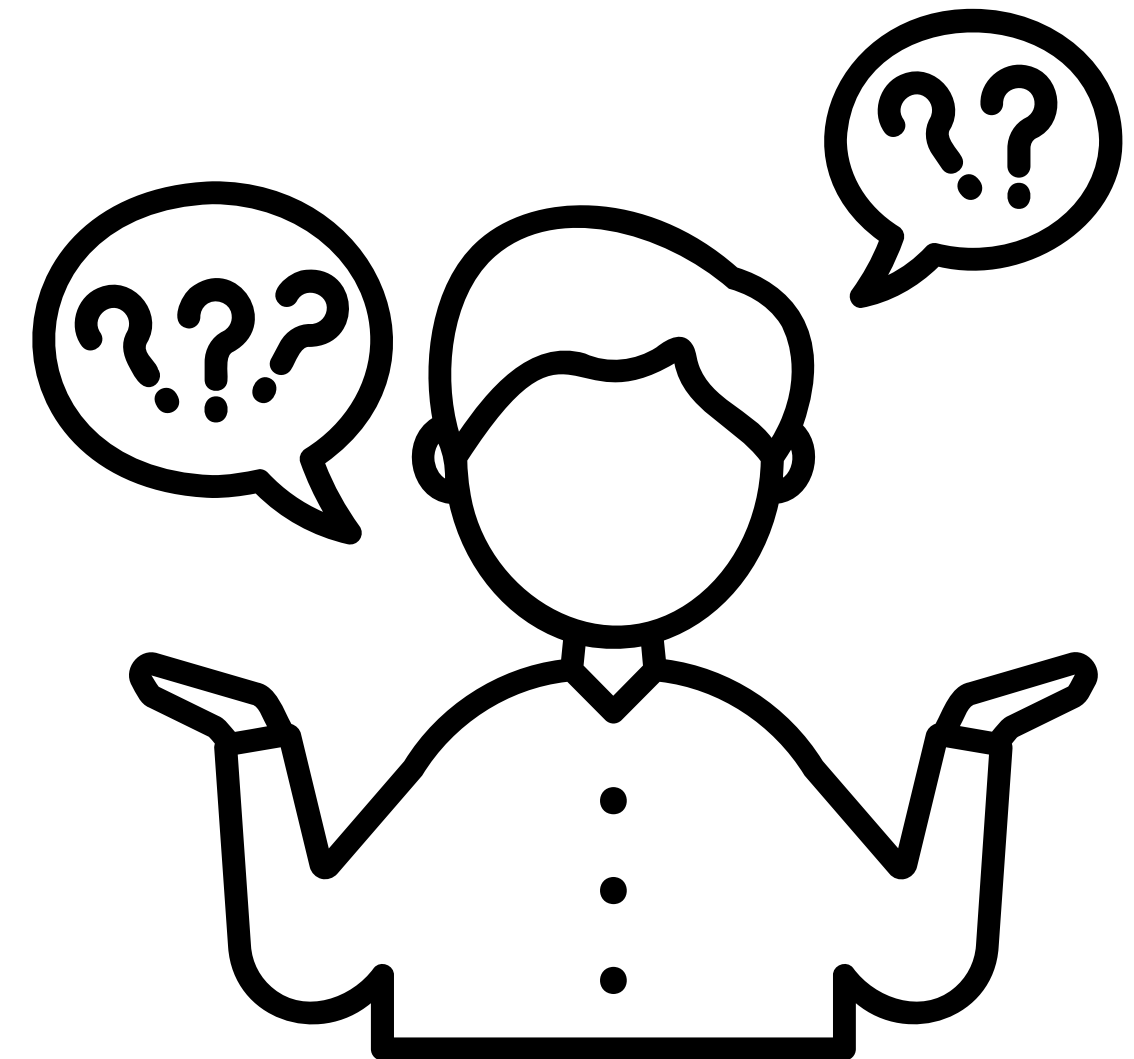
## Kiến trúc thiết kế hệ thống phân tán

- Kanva là một hệ thống whiteboard cộng tác thời gian thực được phát triển dựa trên kiến trúc microservices.
- Hệ thống cho phép nhiều người dùng vẽ, viết, thêm hình ảnh và tương tác trên một bảng trắng chia sẻ, với các thay đổi được đồng bộ gần như ngay lập tức.


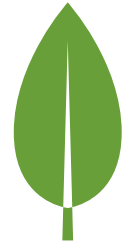
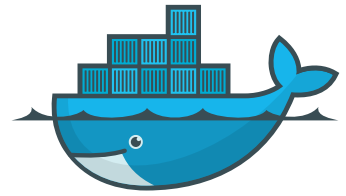



## Lý do chọn kiến trúc *Microservices*

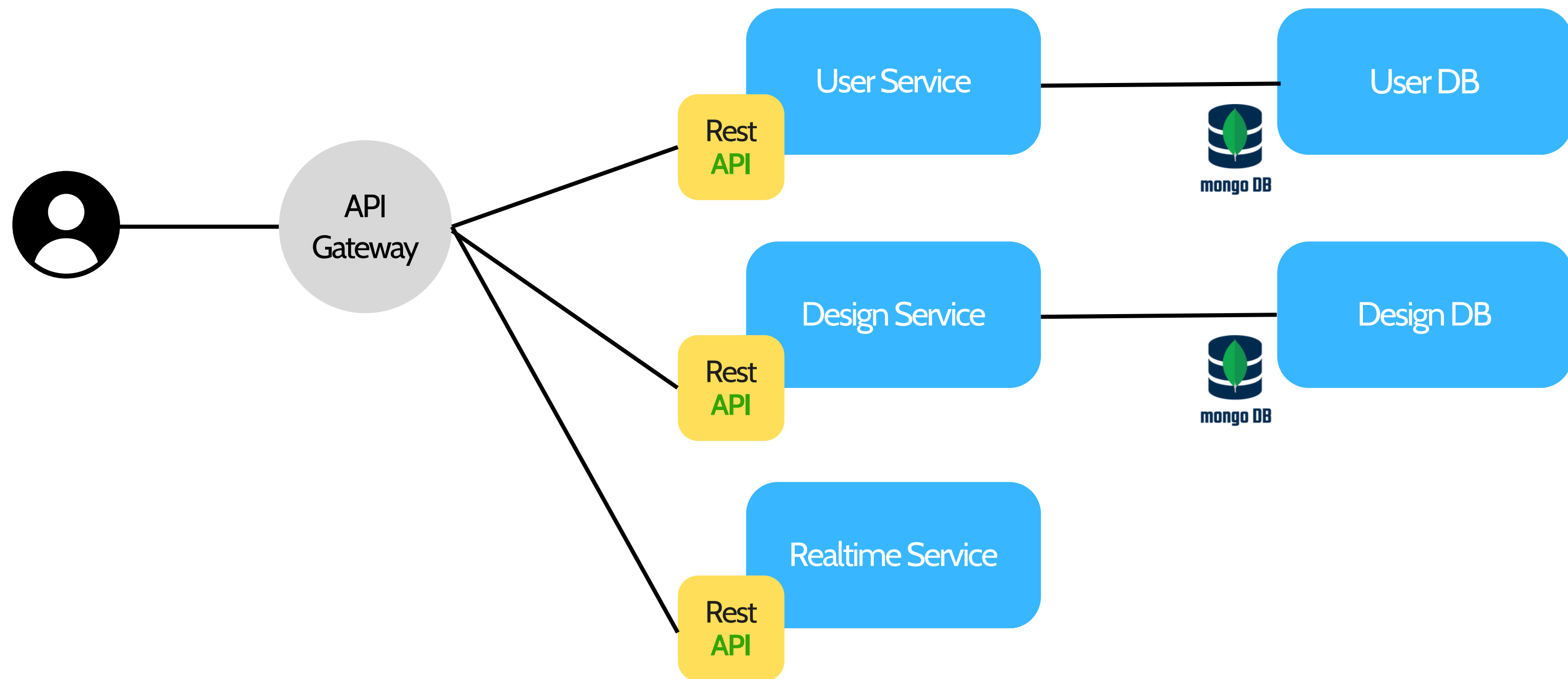
- Tính linh hoạt và dễ mở rộng
- Phát triển và triển khai độc lập
- Khả năng chịu lỗi cao



## Công nghệ sử dụng

- NestJS: Dùng để xây dựng các microservice và tích hợp WebSocket. 
- MongoDB: Cơ sở dữ liệu NoSQL, phù hợp để lưu trữ dữ liệu phi cấu trúc như nội dung bảng vẽ. 
- Docker: Đóng gói các dịch vụ, giúp triển khai dễ dàng và nhất quán. 
- Giao tiếp thời gian thực: Sử dụng giao thức WebSocket để đồng bộ thao tác người dùng tức thời.
- Frontend (Vuejs): Giao diện bảng vẽ được xây dựng bằng Vuejs, sử dụng kết hợp với Fabricjs và HTML5. 

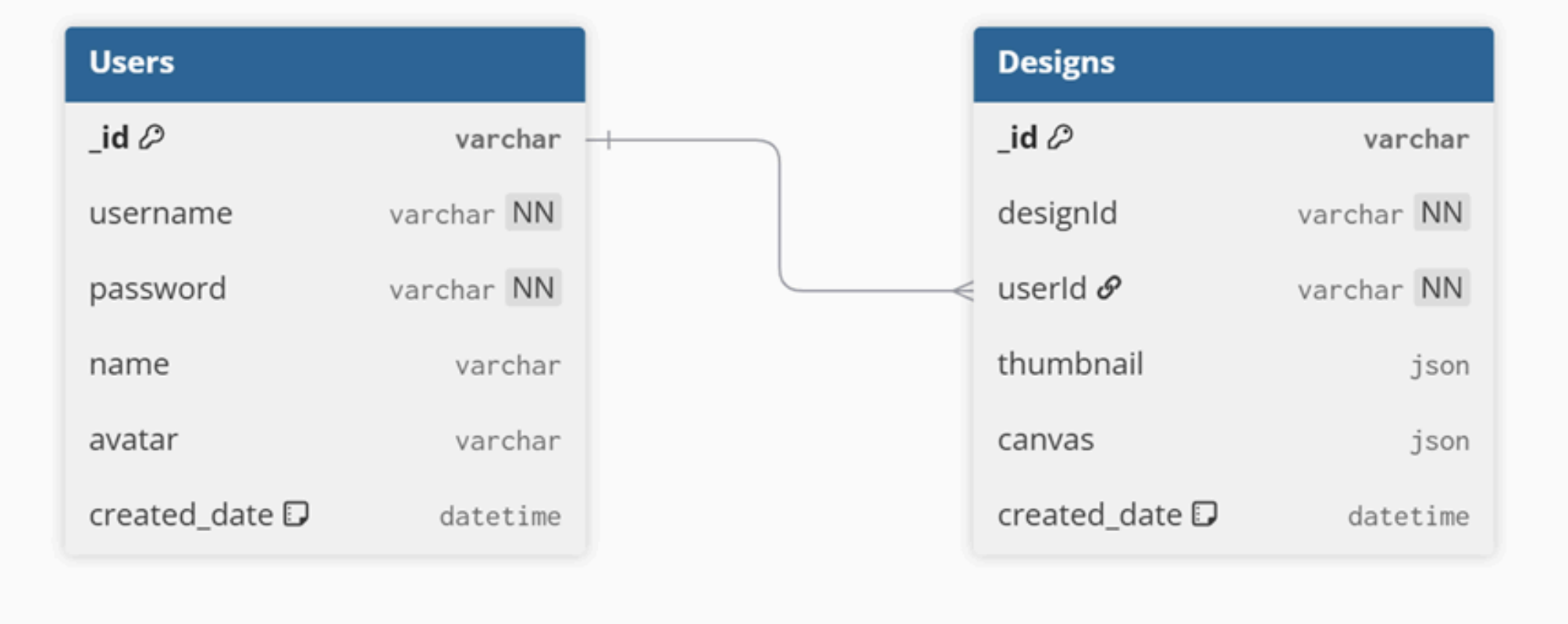
# Kiến trúc tổng thể



Hình 1: Services trong Microservices

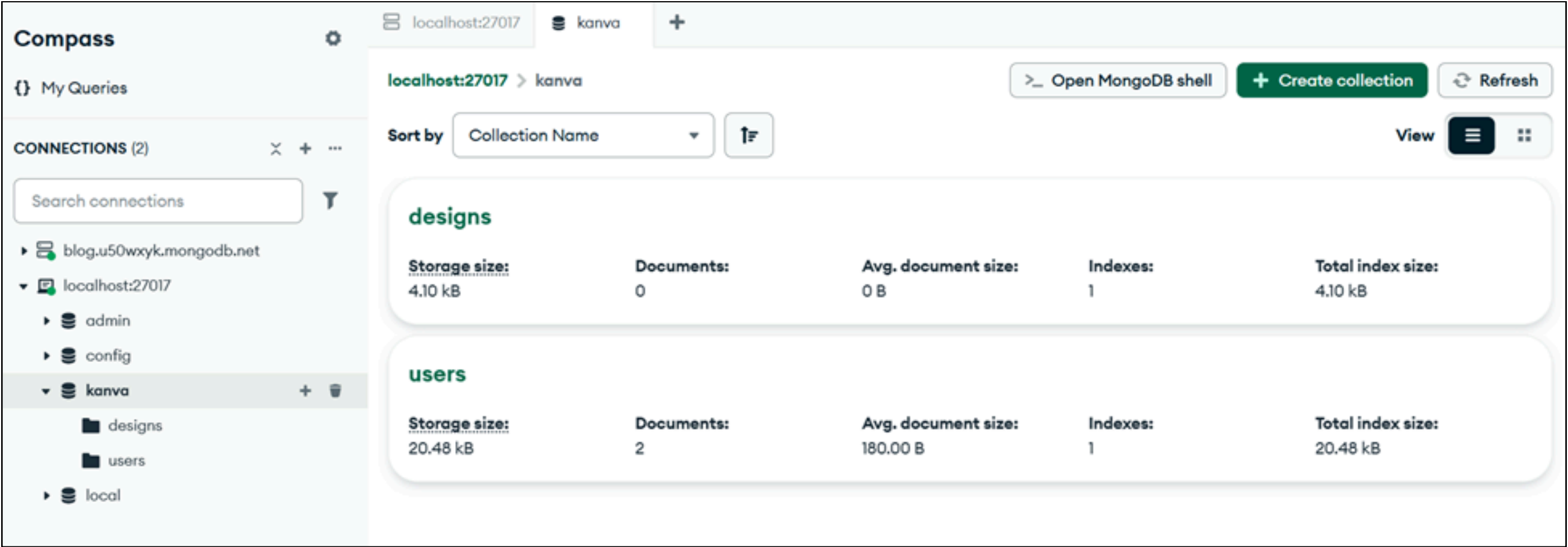


# Cơ sở dữ liệu



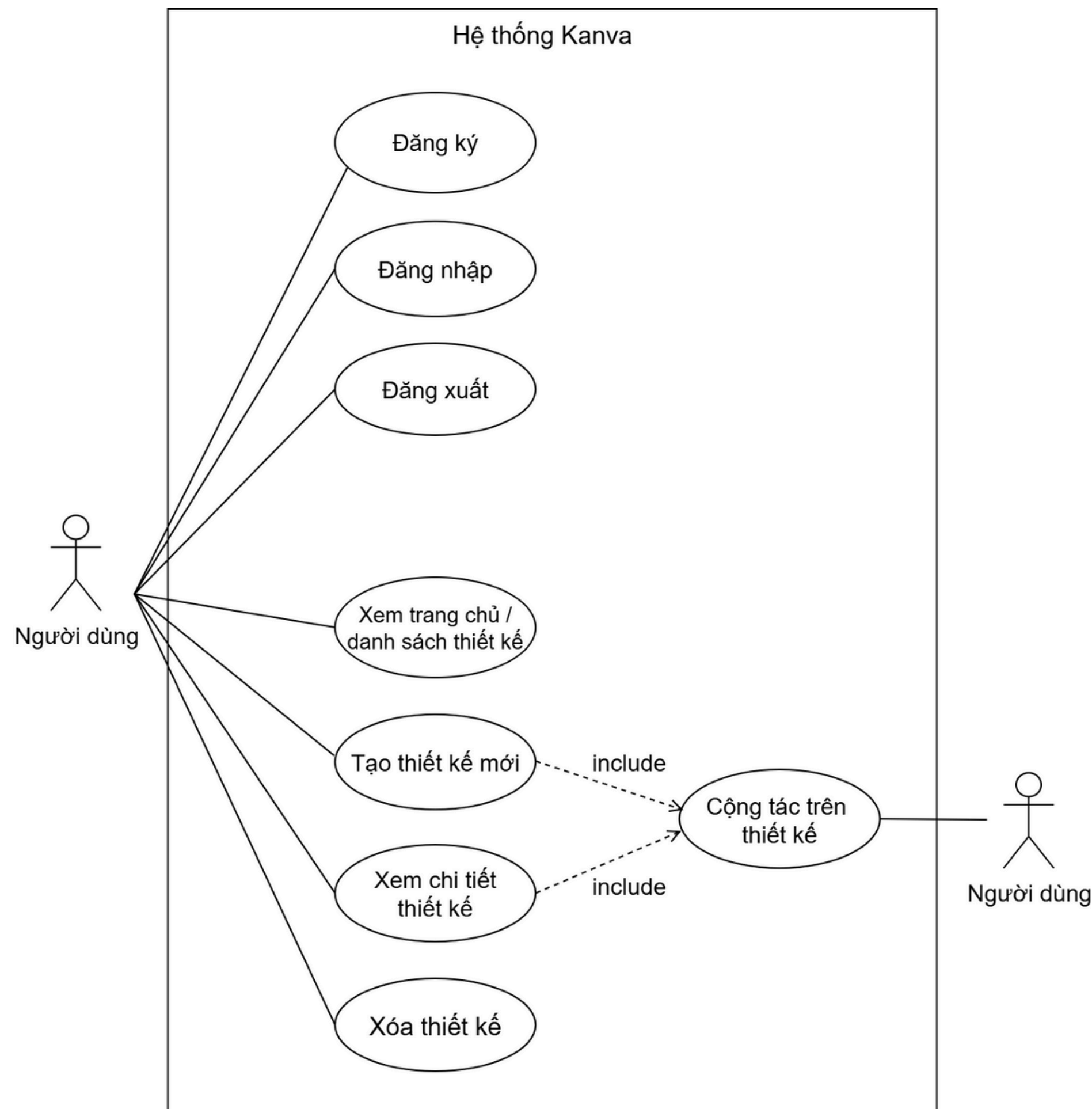
Hình 2: ER Diagram

Hình 3: MongoDB database

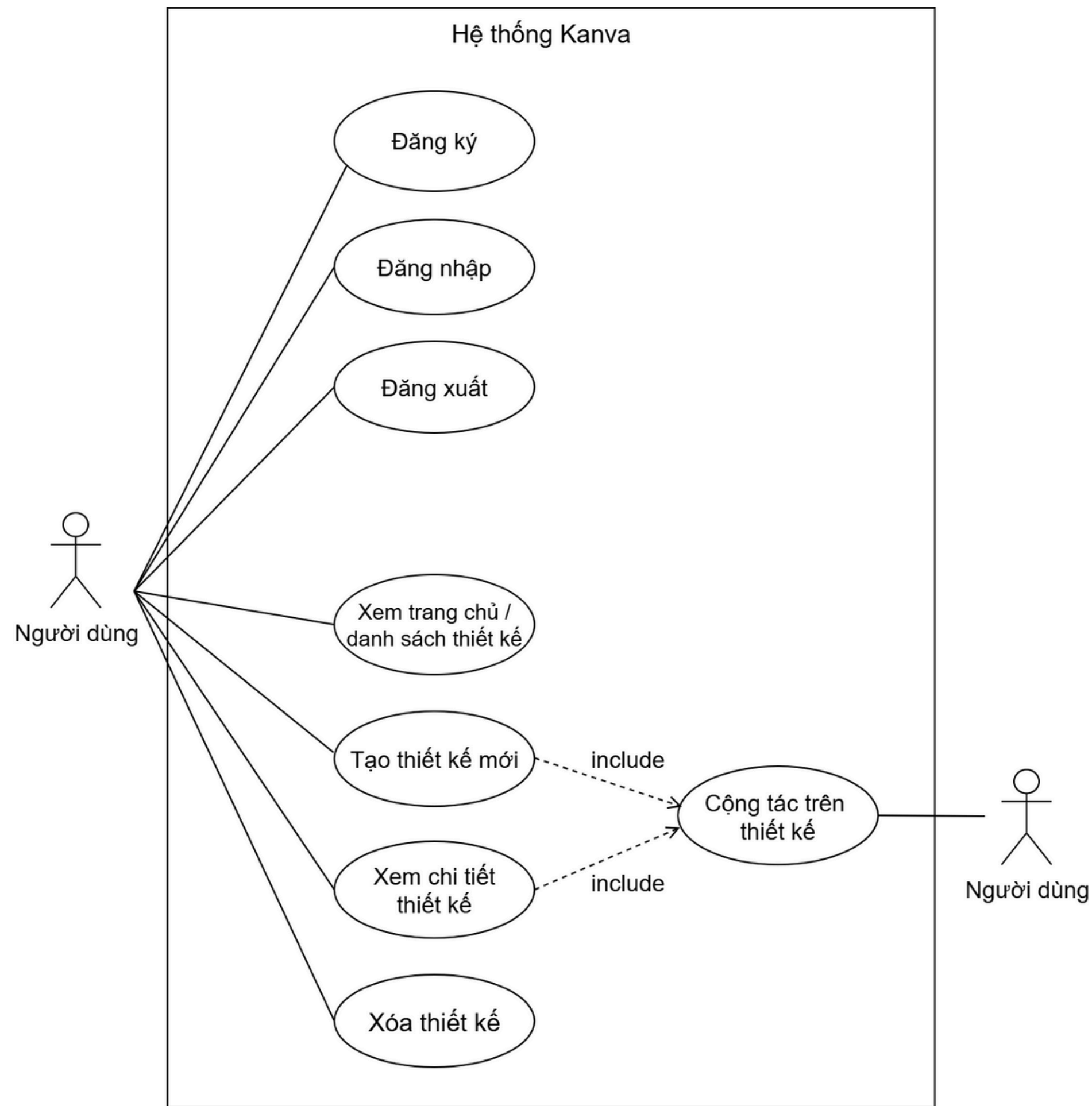


# Sơ đồ Use case

Sơ đồ Use Case để hình dung rõ hơn về các chức năng và tương tác giữa người dùng với hệ thống Kanva.









## Kết quả và đánh giá

- Kết quả: Hệ thống đã triển khai thành công các chức năng quản lý người dùng, quản lý thiết kế và cộng tác thời gian thực.
- Ưu điểm: Độ trễ thấp nhờ WebSocket. Kiến trúc microservices giúp hệ thống có khả năng mở rộng và chịu lỗi tốt.
- Hạn chế: Các tính năng còn cơ bản, chưa có chức năng undo/redo và cơ chế phân quyền còn đơn giản.



**Học viện**  
**Công nghệ Bưu chính Viễn thông**

**THANK YOU!**