

TRƯỜNG ĐẠI HỌC SÀI GÒN
KHOA CÔNG NGHỆ THÔNG TIN



HỌC PHẦN: XÂY DỰNG PHẦN MỀM THEO MÔ HÌNH
PHÂN LỚP
ĐỀ TÀI: XÂY DỰNG ỨNG DỤNG FACEBOOK CLONE

Giảng viên hướng dẫn: **ThS. Từ Lăng Phiêu**

Sinh viên thực hiện: Phạm Trà Trường Giang - 3121410168

Email liên hệ: giangphamtratuong@gmail.com

Thành phố Hồ Chí Minh, tháng 05 năm 2025

This image shows a full page of white paper with horizontal dotted lines, typical of primary school writing paper. The lines are evenly spaced and run across the width of the page. There are no margins, text, or other markings on the paper.

LỜI CẢM ƠN

Trước tiên, em xin bày tỏ lòng biết ơn chân thành đến Thầy Từ Lãng Phiêu vì đã luôn tận tình hướng dẫn và chỉ bảo em trong suốt quá trình thực hiện báo cáo này. Những lời khuyên quý báu cùng sự hỗ trợ tận tâm từ Thầy không chỉ giúp em vượt qua những khó khăn mà còn khơi dậy trong em sự sáng tạo và khả năng tư duy phân tích sâu sắc hơn.

Dù đã cố gắng hoàn thiện nội dung một cách tốt nhất, em vẫn nhận thấy rằng báo cáo không tránh khỏi những thiếu sót. Em rất mong nhận được những góp ý từ Thầy để có thể cải thiện và nâng cao chất lượng sản phẩm trong tương lai.

Cuối cùng, em xin kính chúc Thầy luôn dồi dào sức khỏe, giữ vững ngọn lửa đam mê trong sự nghiệp giảng dạy và nghiên cứu. Em hy vọng sẽ tiếp tục được học hỏi thêm nhiều kiến thức quý báu từ Thầy trong tương lai.

Trân trọng,

Nhóm 11

MỤC LỤC

| | |
|--|-----------|
| LỜI CẢM ƠN | 3 |
| DANH MỤC HÌNH | 5 |
| CHƯƠNG I: TỔNG QUAN ĐỀ TÀI | 6 |
| 1. Giới thiệu đề tài | 6 |
| 2. Mục tiêu đề tài | 6 |
| 3. Phạm vi đề tài | 7 |
| 4. Các chức năng của dự án | 8 |
| 4.1. Chức năng đăng ký/đăng nhập người dùng | 8 |
| 4.2. Chức năng đăng bài viết (văn bản, hình ảnh, video) | 8 |
| 4.3. Chức năng tương tác (thích, bình luận, chia sẻ bài viết) | 9 |
| 4.4. Chức năng quản lý bạn bè (kết bạn, hủy kết bạn, danh sách bạn bè) | 9 |
| 4.5. Chức năng quản lý hồ sơ cá nhân | 9 |
| 4.6. Trang quản trị (Admin) | 9 |
| CHƯƠNG II: CƠ SỞ LÝ THUYẾT | 11 |
| 1. Lý thuyết chung | 11 |
| 2. Mô hình ứng dụng | 13 |
| CHƯƠNG III: THIẾT KẾ HỆ THỐNG | 14 |
| 1. Cơ sở dữ liệu | 14 |
| 2. Giao diện ứng dụng | 16 |
| CHƯƠNG IV: KẾT LUẬN | 24 |

DANH MỤC HÌNH

| | |
|--|----|
| Hình 1: Cơ sở dữ liệu | 14 |
| Hình 2: Giao diện Admin dashboard | 17 |
| Hình 3: Giao diện quản lý tài khoản người dùng | 17 |
| Hình 4: Giao diện quản lý bài đăng của người dùng | 18 |
| Hình 5: Giao diện trang chủ | 18 |
| Hình 6: Giao diện bài post | 19 |
| Hình 7: Giao diện trang profile | 19 |
| Hình 8: Giao diện chỉnh sửa thông tin người dùng | 20 |
| Hình 9: Giao diện tìm kiếm người dùng | 21 |
| Hình 10: Giao diện trang bạn bè | 21 |
| Hình 11: Giao diện trang profile của người dùng khác | 22 |
| Hình 12: Giao diện gửi lời mời kết bạn | 22 |
| Hình 13: Giao diện trang đăng nhập | 23 |
| Hình 14: Giao diện trang đăng ký | 24 |

CHƯƠNG I: TỔNG QUAN ĐỀ TÀI

1. Giới thiệu đề tài

Trong thời đại công nghệ số phát triển mạnh mẽ, mạng xã hội đã trở thành một phần không thể thiếu trong đời sống hằng ngày của con người. Việc giao tiếp, chia sẻ thông tin, hình ảnh, video hay kết nối bạn bè không còn bị giới hạn bởi khoảng cách địa lý. Facebook là một trong những nền tảng mạng xã hội phổ biến nhất hiện nay, với hàng tỷ người dùng trên toàn thế giới. Tuy nhiên, việc nghiên cứu, phân tích và mô phỏng lại một phần hệ thống như Facebook mang lại nhiều giá trị trong quá trình học tập và phát triển kỹ năng lập trình, thiết kế hệ thống và tư duy giải quyết vấn đề.

Dự án "Facebook Clone" được thực hiện với mục tiêu xây dựng một ứng dụng mạng xã hội mô phỏng các chức năng cơ bản của Facebook như: đăng ký/đăng nhập tài khoản, đăng bài viết (văn bản, hình ảnh, video), tương tác với bài viết (thích, bình luận, chia sẻ), quản lý bạn bè, cập nhật hồ sơ cá nhân và quản trị hệ thống. Ngoài ra, dự án còn hướng đến việc rèn luyện kỹ năng phát triển phần mềm theo mô hình client-server, sử dụng các công nghệ hiện đại như React cho giao diện người dùng (frontend), Spring Boot cho phía máy chủ (backend), và MySQL làm hệ quản trị cơ sở dữ liệu.

Thông qua quá trình xây dựng Facebook Clone, em không chỉ có cơ hội áp dụng kiến thức lý thuyết đã học vào thực tiễn, mà còn hiểu rõ hơn về cách hoạt động của các hệ thống mạng xã hội hiện đại, từ đó nâng cao tư duy hệ thống, khả năng thiết kế kiến trúc phần mềm, và làm quen với quy trình phát triển phần mềm thực tế.

2. Mục tiêu đề tài

Mục tiêu chính của dự án **Facebook Clone** là xây dựng một ứng dụng web mô phỏng các chức năng cơ bản của một nền tảng mạng xã hội hiện đại, bao gồm:

- **Xây dựng hệ thống đăng nhập và đăng ký người dùng** với cơ chế xác thực và phân quyền an toàn, sử dụng Spring Security và JWT.
- **Tạo, hiển thị và tương tác với bài viết**: cho phép người dùng đăng bài viết dưới dạng văn bản, hình ảnh hoặc video, cùng với các tính năng tương tác như thích (like), bình luận (comment), và chia sẻ (share).

- **Quản lý quan hệ bạn bè:** gửi lời mời kết bạn, chấp nhận hoặc từ chối, và hiển thị danh sách bạn bè.
- Xây dựng giao diện người dùng thân thiện, đáp ứng tốt trên nhiều thiết bị, sử dụng React kết hợp với Tailwind CSS và Shadcn UI.
- Phát triển kiến trúc backend rõ ràng, tách biệt, áp dụng mô hình Spring MVC cùng cơ sở dữ liệu MySQL để lưu trữ và truy vấn dữ liệu hiệu quả.

Ngoài các chức năng kỹ thuật, dự án còn hướng tới:

- Tạo nền tảng mở rộng, có thể bổ sung các tính năng nâng cao như trò chuyện thời gian thực, thông báo, hoặc tích hợp trí tuệ nhân tạo trong các phiên bản tiếp theo.

3. Phạm vi đề tài

Phạm vi của dự án tập trung vào việc phát triển các tính năng cốt lõi của một nền tảng mạng xã hội cơ bản, bao gồm:

- **Quản lý người dùng:** Đăng ký tài khoản mới, đăng nhập, cập nhật thông tin cá nhân.
- **Quản lý bài viết và tương tác:**

Tạo bài viết mới với nội dung văn bản hoặc đính kèm hình ảnh/video.

Hiển thị bài viết trên bảng tin (news feed).

Thực hiện hành động thích, bình luận và chia sẻ bài viết.

- **Quản lý bạn bè:** Gửi lời mời kết bạn, đồng ý hoặc từ chối, và xem danh sách bạn bè hiện có.
- **Giao diện người dùng:**
Thiết kế các trang chính như Đăng nhập/Đăng ký, Bảng tin, Trang cá nhân và Danh sách bạn bè.
Sử dụng **React** cho hiệu năng cao và trải nghiệm mượt mà.

Dùng **Tailwind CSS** + **Shaden UI** để xây dựng giao diện đẹp, hiện đại và responsive.

Dự án không bao gồm các tính năng phức tạp như:

- Trò chuyện thời gian thực (real-time chat)
- Phân tích dữ liệu người dùng
- Gợi ý bạn bè dựa trên thuật toán học máy
- Hoặc lưu trữ và xử lý tệp đa phương tiện trên nền tảng đám mây

Những chức năng này sẽ được cân nhắc để phát triển trong các phiên bản mở rộng nếu thời gian và nguồn lực cho phép

4. Các chức năng của dự án

4.1. Chức năng đăng ký/đăng nhập người dùng

- Cho phép người dùng tạo tài khoản mới với các thông tin cơ bản như tên, email, mật khẩu.
- Mã hóa mật khẩu trước khi lưu trữ vào cơ sở dữ liệu nhằm tăng tính bảo mật.
- Cung cấp chức năng đăng nhập bằng thông tin tài khoản đã đăng ký.
- Sử dụng **JWT (JSON Web Token)** để xác thực và duy trì phiên đăng nhập của người dùng.
- Hỗ trợ thông báo lỗi khi đăng nhập sai thông tin hoặc đăng ký trùng email.

4.2. Chức năng đăng bài viết (văn bản, hình ảnh, video)

- Cho phép người dùng đăng bài viết với nội dung văn bản ngắn.
- Hỗ trợ đính kèm hình ảnh hoặc video vào bài viết.
- Giao diện hiển thị bài viết theo thời gian đăng (mới nhất lên đầu), bao gồm tên người đăng, ảnh đại diện, nội dung và thời gian.

- Lưu trữ thông tin bài viết vào cơ sở dữ liệu và xử lý upload tệp đa phương tiện lên thư mục hoặc dịch vụ lưu trữ.

4.3. Chức năng tương tác (thích, bình luận, chia sẻ bài viết)

- Thích (Like): Cho phép người dùng nhấn thích bài viết. Mỗi người chỉ được thích một lần.
- Bình luận (Comment): Người dùng có thể để lại bình luận dưới mỗi bài viết. Bình luận được hiển thị theo thứ tự thời gian.
- Chia sẻ (Share): Cho phép người dùng chia sẻ bài viết của người khác lên dòng thời gian của mình.

4.4. Chức năng quản lý bạn bè (kết bạn, hủy kết bạn, danh sách bạn bè)

- Cho phép người dùng gửi lời mời kết bạn đến người dùng khác.
- Cho phép người dùng chấp nhận hoặc từ chối lời mời kết bạn.
- Hiển thị danh sách bạn bè hiện có, hỗ trợ tìm kiếm bạn bè theo tên.
- Hỗ trợ chức năng hủy kết bạn với người dùng bất kỳ trong danh sách.

4.5. Chức năng quản lý hồ sơ cá nhân

- Cho phép người dùng xem và chỉnh sửa thông tin cá nhân
- Hỗ trợ cập nhật ảnh đại diện (avatar) bằng cách upload từ thiết bị.
- Hiển thị trang hồ sơ cá nhân gồm các bài viết đã đăng, số lượng bạn bè, ảnh đại diện và thông tin cơ bản.

4.6. Trang quản trị (Admin)

- Dành riêng cho người dùng có vai trò quản trị viên (admin).
- Quản lý người dùng: Xem danh sách tài khoản người dùng, khóa/mở tài khoản, xóa tài khoản vi phạm.

- Quản lý bài viết: Xem và xóa các bài viết vi phạm nội dung hoặc bị người dùng báo cáo.
- Giao diện quản lý đơn giản, dễ sử dụng, hiển thị các thống kê cơ bản về số lượng người dùng và bài viết.

CHƯƠNG II: CƠ SỞ LÝ THUYẾT

1. Lý thuyết chung

Dự án Facebook Clone được xây dựng dựa trên sự kết hợp của các công nghệ và thư viện hiện đại, đảm bảo hiệu suất và khả năng mở rộng. Dưới đây là các công nghệ chính được sử dụng trong dự án:

Kiến trúc Client - Server

Mô hình Client - Server là kiến trúc phổ biến trong phát triển web, trong đó:

- **Client (Frontend)** là phía giao diện người dùng, gửi yêu cầu và nhận phản hồi từ server.
- **Server (Backend)** xử lý logic nghiệp vụ, quản lý dữ liệu, xác thực người dùng và phản hồi lại client.

Trong dự án này, frontend sử dụng **React** và backend sử dụng **Spring Boot** để triển khai mô hình client-server, đảm bảo khả năng mở rộng và bảo trì dễ dàng.

RESTful API

REST (Representational State Transfer) là kiến trúc thiết kế API phổ biến. RESTful API cho phép frontend và backend giao tiếp thông qua HTTP với các phương thức như GET, POST, PUT, DELETE.

Dự án sử dụng RESTful API để quản lý các đối tượng như người dùng, bài viết, bình luận, tương tác, v.v., đảm bảo tính tách biệt giữa frontend và backend.

Spring Boot và Spring Security

- **Spring Boot** là một framework Java giúp xây dựng các ứng dụng web và RESTful API một cách nhanh chóng, với cấu hình mặc định và khả năng mở rộng cao.

- **Spring Security** là module hỗ trợ xác thực và phân quyền người dùng. Trong dự án, Spring Security được kết hợp với **JWT (JSON Web Token)** để đảm bảo bảo mật cho các endpoint.

React

React là thư viện JavaScript do Facebook phát triển, cho phép xây dựng giao diện người dùng bằng cách chia nhỏ thành các component tái sử dụng. Với React:

- Trạng thái được quản lý bằng `useState`, `useEffect`, hoặc thư viện như `Redux`.
- Giao diện được cập nhật hiệu quả nhờ cơ chế **Virtual DOM**.
- Dự án tận dụng React để xây dựng các trang như: đăng nhập, đăng ký, dòng thời gian, hồ sơ cá nhân, quản lý bạn bè...

Cơ sở dữ liệu quan hệ - MySQL

MySQL là hệ quản trị cơ sở dữ liệu quan hệ mã nguồn mở, phổ biến trong các ứng dụng web. Các bảng dữ liệu chính trong dự án bao gồm:

- `users`: lưu thông tin người dùng.
- `posts`: lưu bài viết (văn bản, hình ảnh, video).
- `comments`, `likes`, `friends`, `notifications`, v.v.: lưu thông tin liên quan đến tương tác giữa người dùng.

Dữ liệu được thao tác thông qua **Spring Data JPA**, giúp trừu tượng hóa các truy vấn SQL và giảm thiểu mã lặp.

JSON Web Token (JWT)

JWT là phương thức phổ biến để xác thực người dùng trong các hệ thống phân tán. Mỗi token chứa thông tin người dùng và thời gian hết hạn, được mã hóa và gửi kèm trong các request sau khi đăng nhập thành công.

Trong dự án, JWT giúp bảo mật các route quan trọng như đăng bài, cập nhật hồ sơ, truy xuất danh sách bạn bè,...

2. Mô hình ứng dụng

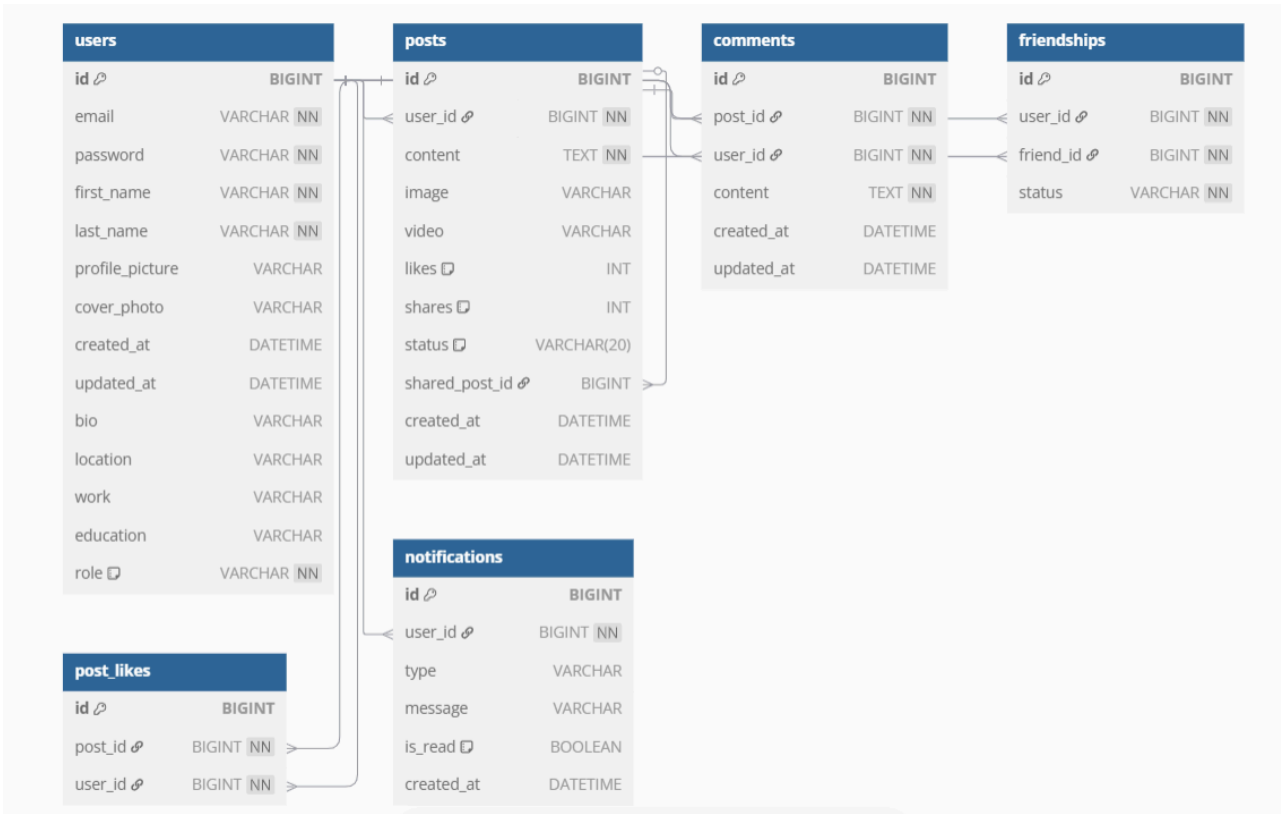
Mô hình kiến trúc **MVC** là viết tắt của Model - View - Controller, một mô hình kiến trúc phần mềm phổ biến giúp phân tách các thành phần logic nghiệp vụ, giao diện và điều phối dữ liệu. Mô hình này giúp tăng khả năng mở rộng, bảo trì và tái sử dụng mã nguồn.

Trong dự án Facebook Clone:

- **Model:** Đại diện cho các lớp dữ liệu (entity) và logic liên quan đến dữ liệu (truy vấn, lưu trữ vào cơ sở dữ liệu).
- **View:** Là giao diện người dùng (React), nơi hiển thị dữ liệu và nhận tương tác từ người dùng.
- **Controller:** Là lớp điều khiển trong Spring Boot, tiếp nhận request từ người dùng, xử lý logic, tương tác với Model, và trả về dữ liệu.

CHƯƠNG III: THIẾT KẾ HỆ THỐNG

1. Cơ sở dữ liệu



Hình 1: Cơ sở dữ liệu

Bảng "Users" (Người dùng)

Đây là nơi lưu trữ thông tin cơ bản của mỗi thành viên trong mạng xã hội. Mỗi người dùng được định danh bằng một mã số duy nhất (id), cùng với email (làm địa chỉ đăng nhập, không được trùng lặp), mật khẩu đã mã hóa, tên và họ, ảnh đại diện, ảnh bìa, thời gian tạo tài khoản, tiểu sử, địa chỉ, công việc, học vấn, và vai trò (mặc định là "user", có thể là "admin"). Bảng này đóng vai trò như "trái tim" của hệ thống, là điểm xuất phát để liên kết với các hoạt động khác như đăng bài, bình luận hay kết bạn.

Bảng "Posts" (Bài viết)

Bảng này quản lý tất cả các bài viết mà người dùng đăng lên. Mỗi bài có một mã định danh (id), được liên kết với người đăng qua `user_id`, chứa nội dung (văn bản), có thể kèm ảnh hoặc video, và ghi nhận số lượt thích, lượt chia sẻ (mặc định là 0). Ngoài ra, nó còn hỗ trợ tính năng chia sẻ bài viết khác thông qua `shared_post_id`, cùng với trạng thái bài viết (mặc định là "active") và thời gian tạo/sửa. Đây là nơi lưu giữ mọi "câu chuyện" mà người dùng muốn chia sẻ.

Bảng "Comments" (Bình luận)

Bảng này lưu trữ các bình luận mà người dùng để lại trên bài viết. Mỗi bình luận có mã riêng (id), liên kết với bài viết (`post_id`) và người bình luận (`user_id`), cùng nội dung và thời gian tạo/sửa. Đây là cầu nối giúp người dùng tương tác sâu hơn với các bài đăng, tạo nên sự sôi nổi trong cộng đồng.

Bảng "Friendships" (Quan hệ bạn bè)

Bảng này giúp quản lý mối quan hệ bạn bè giữa các thành viên. Mỗi quan hệ có mã định danh (id), với `user_id` là người gửi lời mời và `friend_id` là người nhận, cùng trạng thái (có thể là "PENDING" hoặc "ACCEPTED"). Đây là nền tảng để xây dựng mạng lưới kết nối xã hội trong hệ thống.

Bảng "Post_likes" (Lượt thích bài viết)

Bảng này theo dõi từng lượt thích mà người dùng dành cho bài viết. Mỗi lượt thích có mã (id), liên kết với bài viết (`post_id`) và người thích (`user_id`). Điều này giúp hệ thống ghi nhận chính xác ai đã thích bài nào, thay vì chỉ dựa vào số liệu tổng quát.

Bảng "Notifications" (Thông báo)

Cuối cùng, bảng này quản lý các thông báo gửi đến người dùng. Mỗi thông báo có mã (id), liên kết với người nhận (`user_id`), loại thông báo (như "LIKE" hoặc "COMMENT"),

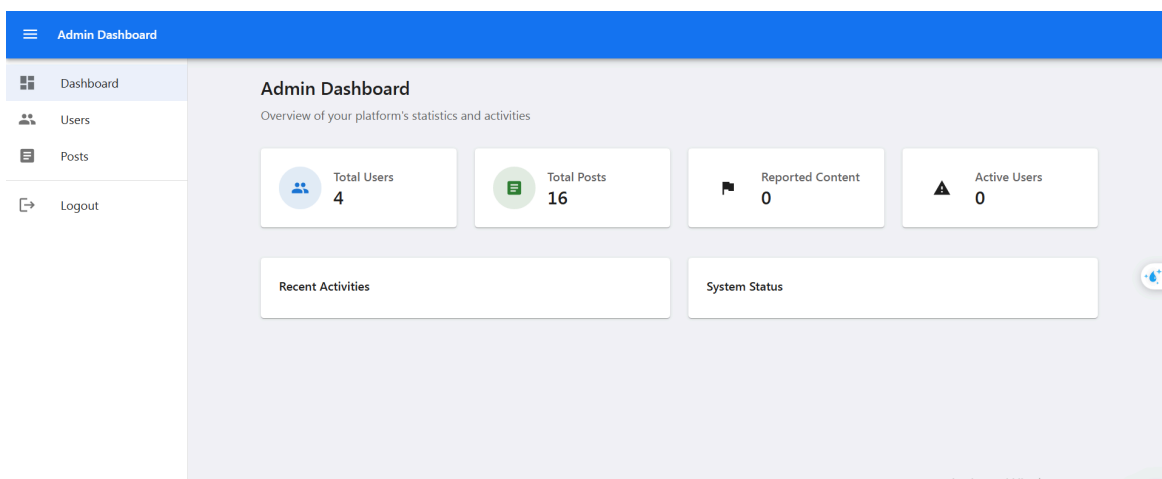
nội dung, trạng thái đã đọc (mặc định là chưa đọc), và thời gian tạo. Bảng này đảm bảo người dùng không bỏ lỡ bất kỳ hoạt động quan trọng nào.

Mối quan hệ giữa các bảng:

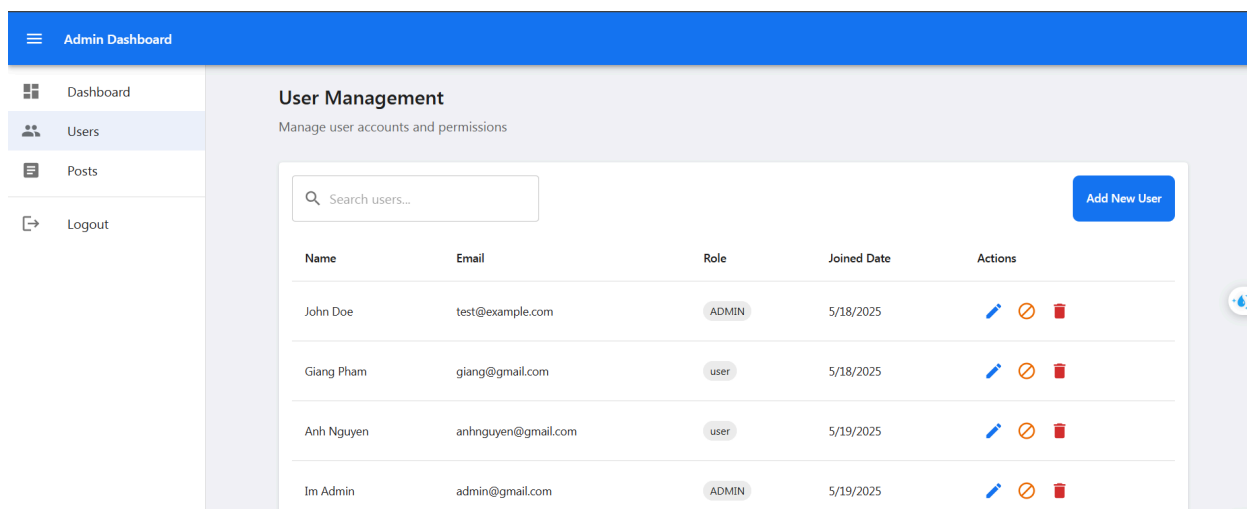
Hệ thống này được xây dựng với các mối quan hệ chặt chẽ:

- Một người dùng có thể đăng nhiều bài viết, nhưng mỗi bài chỉ thuộc về một người dùng.
- Một bài viết có thể được chia sẻ từ bài khác, tạo thành chuỗi liên kết nội dung.
- Một bài viết có thể nhận nhiều bình luận, và mỗi bình luận thuộc về một bài viết cũng như một người dùng.
- Một người dùng có thể gửi/nhận nhiều lời mời kết bạn, với trạng thái được theo dõi rõ ràng.
- Một bài viết có thể nhận nhiều lượt thích, và mỗi lượt thích liên kết với một người dùng.
- Một người dùng có thể nhận nhiều thông báo về các hoạt động liên quan.

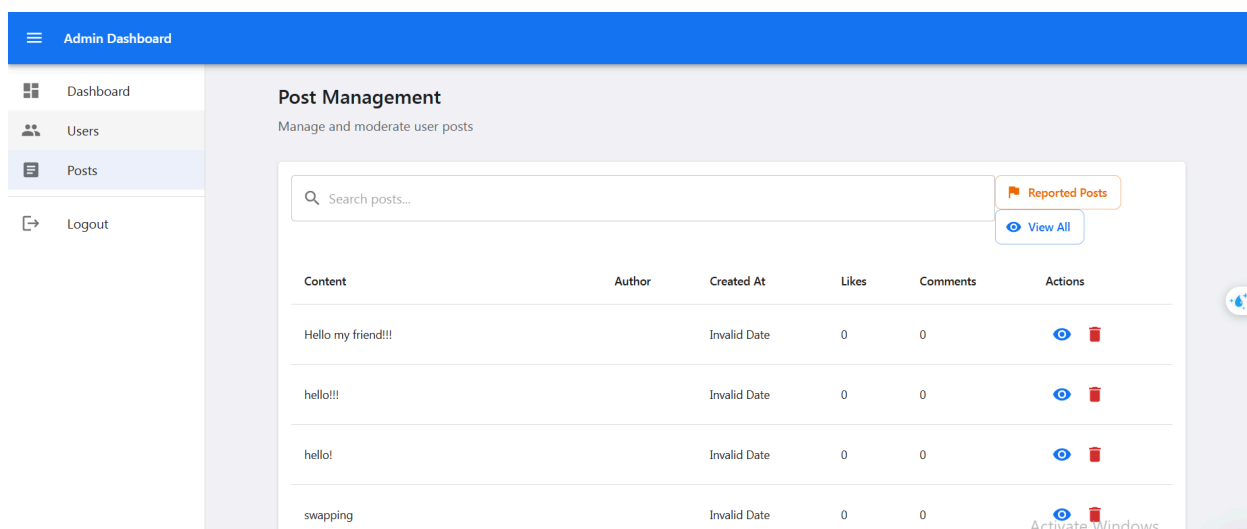
2. Giao diện ứng dụng



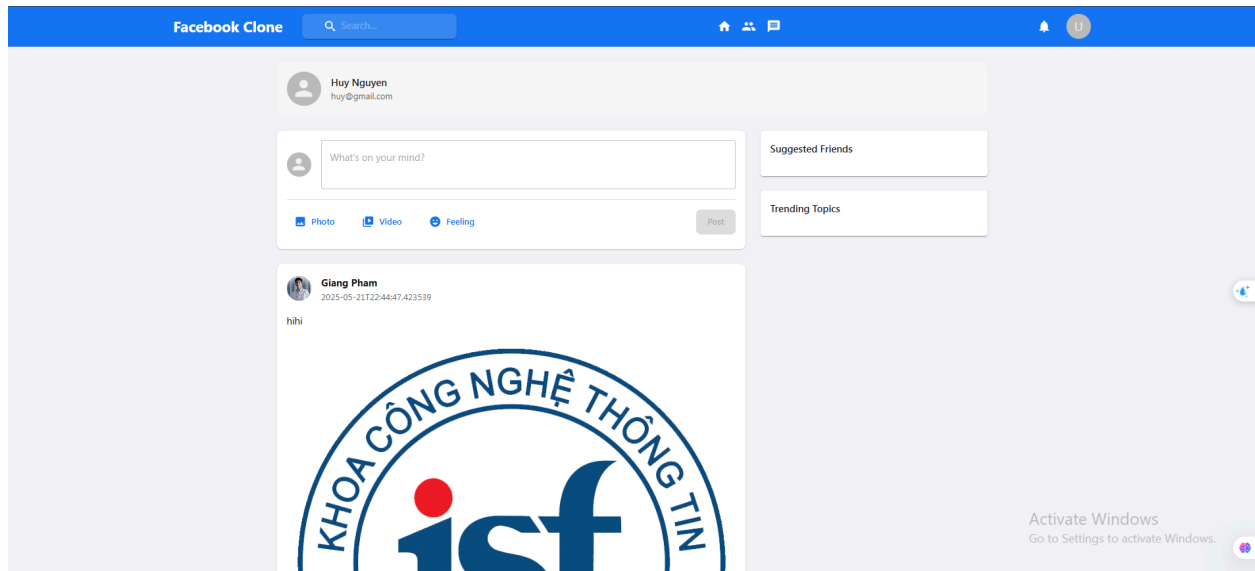
Hình 2: Giao diện Admin dashboard



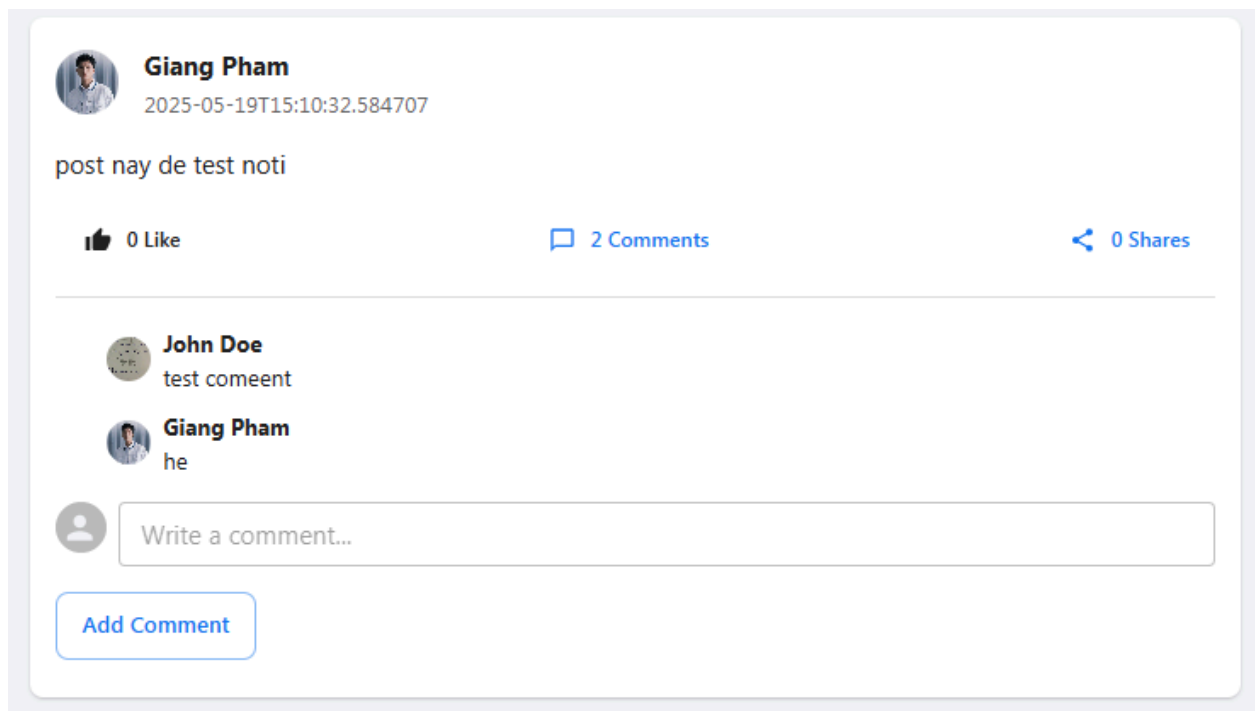
Hình 3: Giao diện quản lý tài khoản người dùng



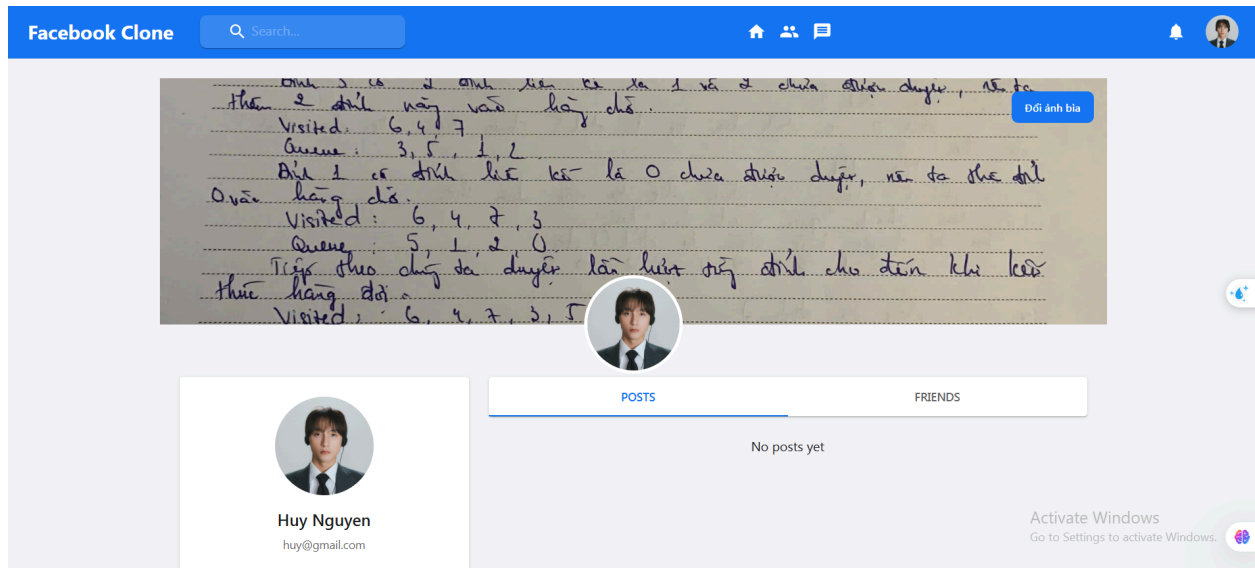
Hình 4: Giao diện quản lý bài đăng của người dùng



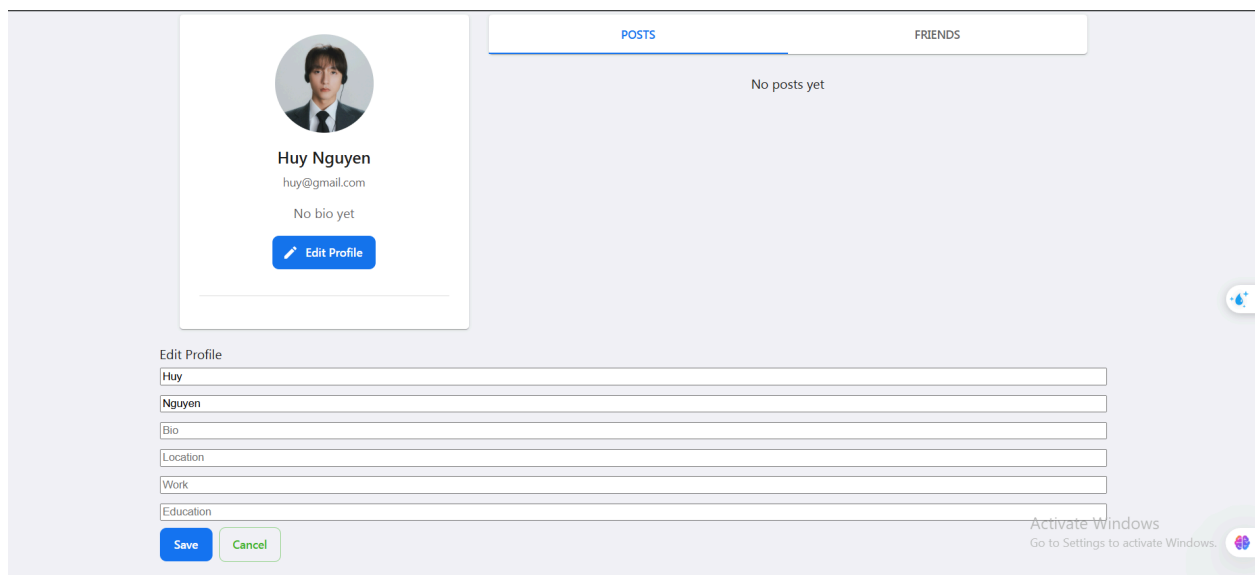
Hình 5: Giao diện trang chủ



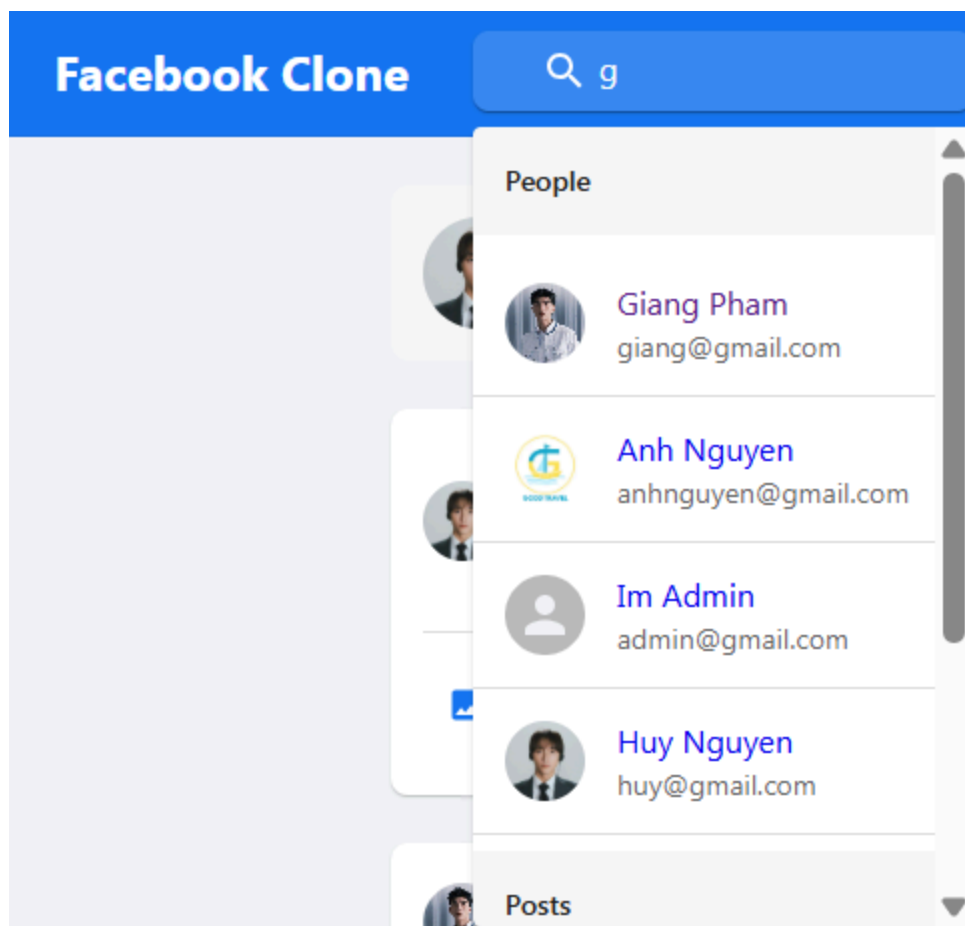
Hình 6: Giao diện bài post



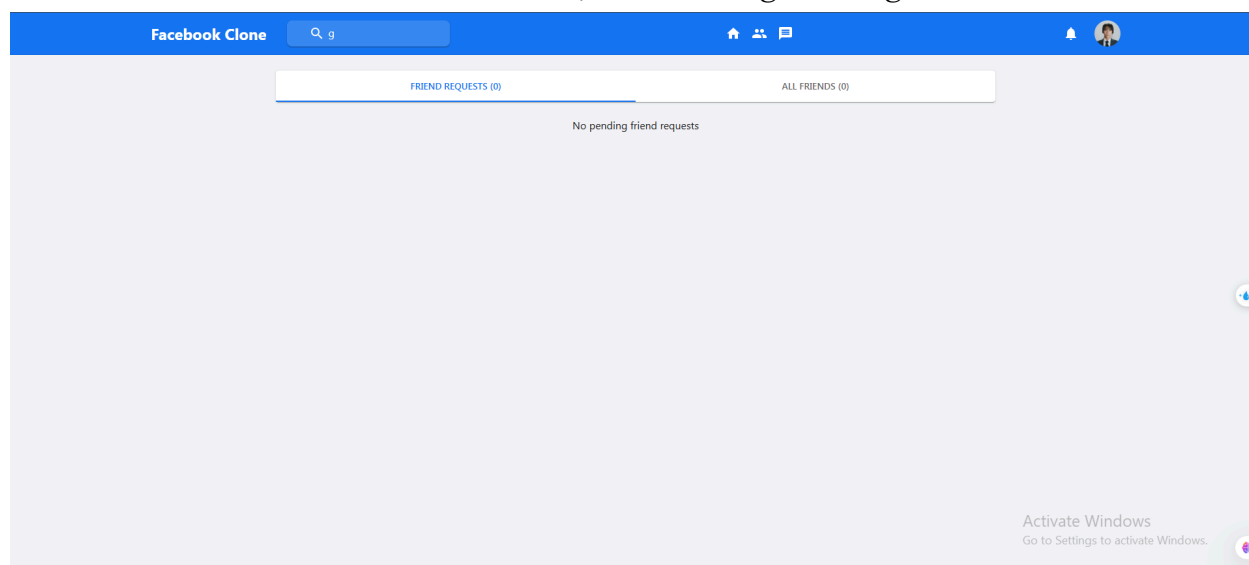
Hình 7: Giao diện trang profile



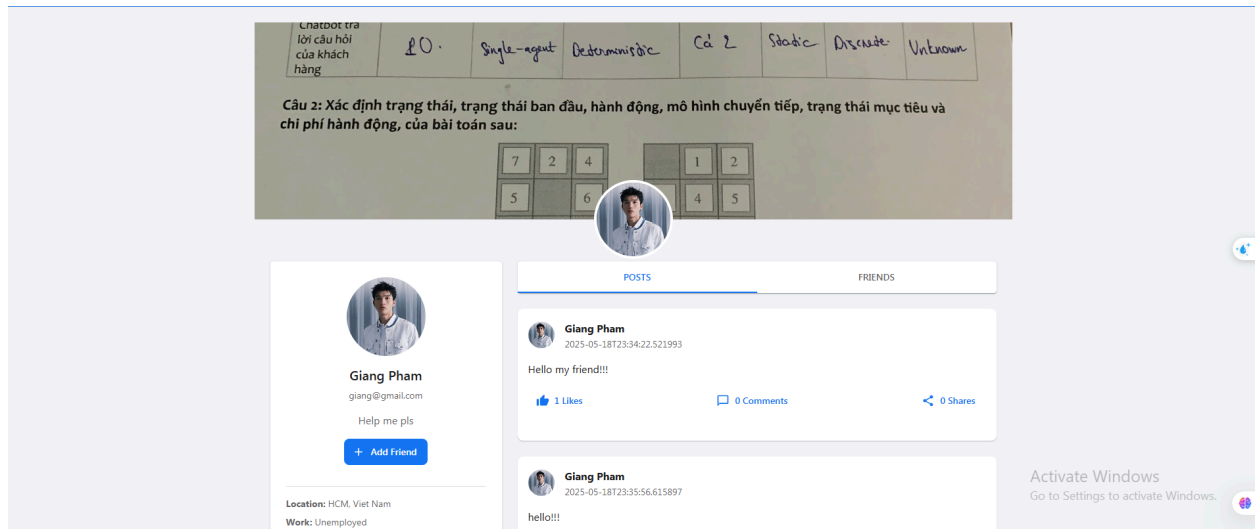
Hình 8: Giao diện chỉnh sửa thông tin người dùng



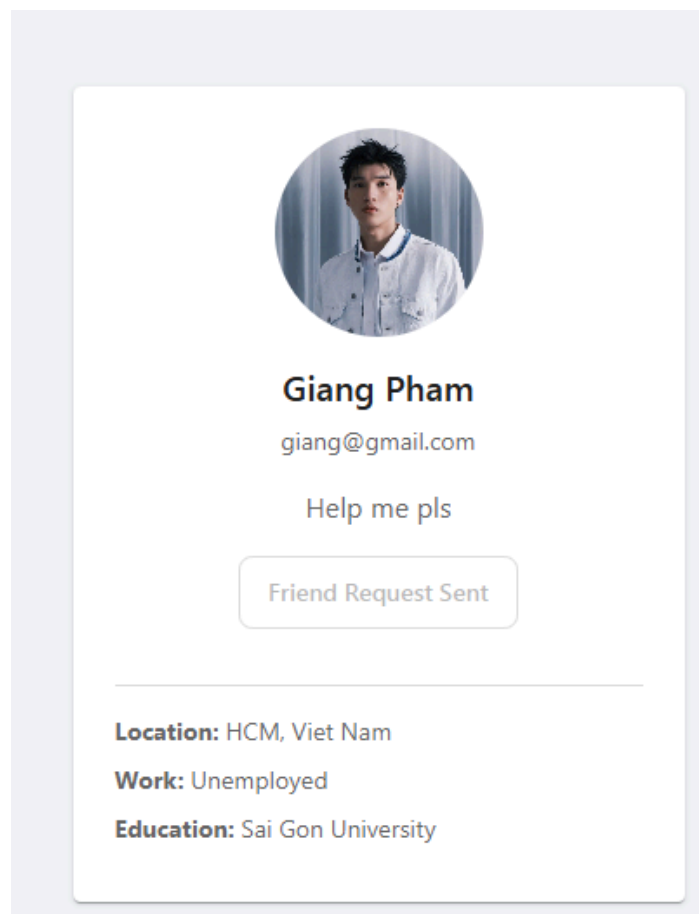
Hình 9: Giao diện tìm kiếm người dùng



Hình 10: Giao diện trang bạn bè




Hình 11: Giao diện trang profile của người dùng khác



Hình 12: Giao diện gửi lời mời kết bạn

Facebook Clone



Log In


[Forgot password?](#)


Don't have an account?

[Sign up](#)

Hình 13: Giao diện trang đăng nhập

Facebook Clone





Already have an account? [Sign in](#)

Hình 14: Giao diện trang đăng ký

CHƯƠNG IV: KẾT LUẬN

Dự án Facebook Clone đã hoàn thiện các chức năng cơ bản của một mạng xã hội như đăng ký, đăng nhập, đăng bài viết, tương tác với bài viết, quản lý bạn bè và hồ sơ cá nhân, cũng như trang quản trị dành cho admin. Việc sử dụng các công nghệ hiện đại như React cho frontend, Spring Boot MVC cho backend và MySQL cho cơ sở dữ liệu đã giúp hệ thống đạt được hiệu năng ổn định, khả năng mở rộng tốt và giao diện thân thiện với người dùng.

Tuy nhiên, dự án vẫn còn một số hạn chế cần khắc phục, bao gồm:

- Chưa tích hợp tính năng phát thông báo theo thời gian thực (real-time) như chat, thông báo mới về bài viết hay lời mời kết bạn.
- Giao diện người dùng còn đơn giản, chưa có nhiều hiệu ứng hoặc trải nghiệm mượt mà.
- Hệ thống bảo mật cơ bản, chưa áp dụng các biện pháp nâng cao như kiểm soát truy cập chi tiết hay bảo vệ chống tấn công mạng nâng cao.
- Chưa tối ưu hóa cho việc mở rộng quy mô lớn với lượng người dùng và dữ liệu khổng lồ.

Trong tương lai, dự án có thể được cải tiến và mở rộng với các hướng đi như:

- Thêm tính năng chat và thông báo theo thời gian thực bằng cách tích hợp WebSocket hoặc các công nghệ tương tự.
- Nâng cao trải nghiệm người dùng bằng việc cải tiến giao diện, thiết kế responsive và hiệu ứng động.
- Tăng cường bảo mật với các cơ chế xác thực đa yếu tố, kiểm soát truy cập theo vai trò chi tiết hơn.
- Ứng dụng các công nghệ xử lý dữ liệu lớn và tối ưu hiệu suất backend để phục vụ lượng người dùng lớn hơn.

- Mở rộng các tính năng mạng xã hội như nhóm, sự kiện, livestream hoặc tích hợp trí tuệ nhân tạo để gợi ý nội dung phù hợp.