TRƯỜNG ĐẠI HỌC CÔNG NGHỆ THÔNG TIN &

TRUYỀN THÔNG VIỆT HÀN

**Khoa Khoa Học Máy Tính**



ĐỒ ÁN CƠ SỞ 2

**WEBSITE QUẢN LÝ LỊCH TRÌNH CÁ NHÂN**

Sinh viên thực hiện: **Thái Hoài Vũ - 24IT314**

**Đinh Công Tiến - 24IT273**

Lớp: **24JIT**

Giảng viên hướng dẫn: **TS.Nguyễn Đức Hiển**

Đà Nẵng, tháng 12 năm 2025

TRƯỜNG ĐẠI HỌC CÔNG NGHỆ THÔNG TIN &

TRUYỀN THÔNG VIỆT HÀN

**Khoa Khoa Học Máy Tính**



ĐỒ ÁN CƠ SỞ 2

**WEBSITE QUẢN LÝ LỊCH TRÌNH CÁ NHÂN**

Sinh viên thực hiện: **Thái Hoài Vũ - 24IT314**

**Đinh Công Tiến - 24IT273**

Lớp: **24JIT**

Giảng viên hướng dẫn: **TS.Nguyễn Đức Hiển**

Đà Nẵng, tháng 12 năm 2025

# NHẬN XÉT CỦA GIẢNG VIÊN HƯỚNG DẪN

Đà Nẵng, ngày … tháng … năm 2025

Giảng viên hướng dẫn

*(Ký rõ họ và tên)*

# LỜI CẢM ƠN

Em xin trân trọng cảm ơn

Đà Nẵng, ngày … tháng … năm 2025

Sinh viên

*Thái Hoài Vũ*

*Đinh Công Tiến*

# MỤC LỤC

[NHẬN XÉT CỦA GIẢNG VIÊN HƯỚNG DẪN iii](#_Toc216117335)

[LỜI CẢM ƠN iv](#_Toc216117336)

[MỤC LỤC v](#_Toc216117337)

[DANH MỤC CÁC TỪ VIẾT TẮT vii](#_Toc216117338)

[DANH MỤC HÌNH VẼ viii](#_Toc216117339)

[DANH MỤC BẢNG ix](#_Toc216117340)

[MỞ ĐẦU 1](#_Toc216117341)

[CHƯƠNG 1. GIỚI THIỆU 3](#_Toc216117342)

[1.1. Tổng quan 3](#_Toc216117343)

[1.2. Công cụ hỗ trợ 3](#_Toc216117344)

[1.3. Cấu trúc đồ án 4](#_Toc216117345)

[CHƯƠNG 2. PHÂN TÍCH VÀ THIẾT KẾ HỆ THỐNG 5](#_Toc216117346)

[2.1. Xác định yêu cầu 5](#_Toc216117347)

[2.1.1. Yêu cầu của người dùng 5](#_Toc216117348)

[2.1.2. Yêu cầu chức năng 5](#_Toc216117349)

[2.1.3. Yêu cầu phi chức năng 6](#_Toc216117350)

[2.1.4. Yêu cầu hệ thống 7](#_Toc216117351)

[2.2. Phân tích hệ thống 7](#_Toc216117352)

[2.2.1. Usecase Diagram 7](#_Toc216117353)

[2.2.1.1. Đặc tả Usecase 7](#_Toc216117354)

[2.2.1.2. Các biểu đồ Usecase 12](#_Toc216117355)

[2.2.2. Class Diagram 15](#_Toc216117356)

[2.3. Thiết kế hệ thống 16](#_Toc216117357)

[2.3.1. Thiết kế CSDL ERD 16](#_Toc216117358)

[2.3.1.1. Đặc tả chi tiết các bảng CSDL 17](#_Toc216117358)

[2.3.2. Phác thảo website 27](#_Toc216117359)

[CHƯƠNG 3. XÂY DỰNG WEBSITE VÀ KẾT QUẢ 28](#_Toc216117360)

[3.1. Giao diện Home 28](#_Toc216117361)

[3.2. Giao diện Tasks 29](#_Toc216117362)

[3.3. Giao diện Calendar 29](#_Toc216117363)

[3.4. Giao diện Kanban 30](#_Toc216117364)

[3.5. Giao diện Reports 30](#_Toc216117365)

[3.6. Giao diện Groups 31](#_Toc216117366)

[3.7. Giao diện Export/Import 31](#_Toc216117367)

[3.8. Giao diện Cài đặt 32](#_Toc216117368)

[3.9. Giao diện Profile 32](#_Toc216117369)

[3.10. Giao diện Đăng nhập 33](#_Toc216117370)

[3.11. Giao diện Đăng ký 33](#_Toc216117371)

[3.12. Giao diện Quên mật khẩu 34](#_Toc216117372)

[3.13. Giao diện Xác thực OTP 34](#_Toc216117373)

[CHƯƠNG 4. KẾT LUẬN 35](#_Toc216117374)

[4.1 Kết luận 35](#_Toc216117375)

[4.2 Hướng phát triển 35](#_Toc216117376)

[TÀI LIỆU THAM KHẢO 36](#_Toc216117377)

# DANH MỤC CÁC TỪ VIẾT TẮT

|  |  |
| --- | --- |
| **VIẾT TẮT** | **NỘI DUNG** |
| 5E | Engagement, Exploration, Explanation, Elaboration, Evaluation |
| STEM | Science Technology Engineering Mathematics |

# DANH MỤC HÌNH VẼ

# DANH MỤC BẢNG

# MỞ ĐẦU

**1. Giới thiệu đề tài (Lý do chọn đề tài)**

Trong bối cảnh chuyển đổi số diễn ra mạnh mẽ hiện nay, việc quản lý thời gian và công việc cá nhân không còn đơn thuần là những ghi chép thủ công trên giấy tờ hay sổ tay. Với nhịp sống hối hả và khối lượng thông tin khổng lồ mà mỗi cá nhân phải xử lý hàng ngày, nhu cầu về một công cụ hỗ trợ quản lý khoa học, trực quan và đồng bộ trở nên cấp thiết hơn bao giờ hết. Sự trì hoãn, quên lịch trình hay việc phân bổ thời gian không hợp lý thường dẫn đến căng thẳng và giảm sút hiệu suất làm việc. Mặc dù trên thị trường đã tồn tại nhiều giải pháp quản lý như Trello, Google Calendar hay các ứng dụng ghi chú, nhưng người dùng thường gặp phải vấn đề phân mảnh dữ liệu khi phải sử dụng đồng thời quá nhiều ứng dụng rời rạc để phục vụ cho các mục đích khác nhau.

Xuất phát từ thực tiễn đó, ý tưởng xây dựng ứng dụng "Quản lý lịch trình cá nhân" được hình thành nhằm cung cấp một giải pháp "All-in-one" (Tất cả trong một). Ứng dụng này không chỉ đơn thuần là nơi lưu trữ danh sách công việc mà còn là một trung tâm điều phối hoạt động cá nhân, kết hợp chặt chẽ giữa việc lập kế hoạch dài hạn trên lịch biểu (Calendar) và quản lý tiến độ chi tiết qua bảng Kanban. Việc tích hợp các công cụ này vào một nền tảng duy nhất giúp người dùng có cái nhìn tổng quan, giảm thiểu thao tác chuyển đổi giữa các ứng dụng và tập trung tối đa vào việc hoàn thành mục tiêu.

Bên cạnh ý nghĩa thực tiễn, việc lựa chọn đề tài này còn mang ý nghĩa quan trọng trong quá trình học tập và nghiên cứu. Đây là cơ hội để vận dụng tổng hợp các kiến thức nền tảng về lập trình web, từ việc xây dựng Back-end với Node.js và Express, thiết kế cơ sở dữ liệu quan hệ với PostgreSQL, cho đến việc xử lý giao diện Front-end tương tác cao. Đặc biệt, đề tài còn tạo điều kiện để tiếp cận các kỹ thuật nâng cao như bảo mật xác thực hai lớp (2FA), tích hợp đăng nhập qua bên thứ ba (OAuth) và xử lý giao tiếp thời gian thực, qua đó hoàn thiện kỹ năng của một lập trình viên Full-stack tương lai.

**2. Mục tiêu của đề tài**

Mục tiêu cốt lõi của đề tài là xây dựng hoàn thiện một hệ thống website quản lý công việc và lịch trình hoạt động ổn định, đáp ứng đầy đủ các tiêu chuẩn của một ứng dụng web hiện đại. Về mặt chức năng, hệ thống hướng tới việc cung cấp cho người dùng khả năng kiểm soát toàn diện đối với quỹ thời gian của mình. Cụ thể, ứng dụng cho phép người dùng khởi tạo, chỉnh sửa và theo dõi trạng thái của các nhiệm vụ một cách linh hoạt thông qua giao diện kéo thả trực quan trên bảng Kanban, đồng thời hỗ trợ lập kế hoạch chi tiết theo thời gian thực trên giao diện Lịch. Bên cạnh đó, hệ thống cũng hướng đến việc hỗ trợ làm việc nhóm thông qua các tính năng cộng tác như tạo nhóm chat, trao đổi tin nhắn và chia sẻ tài liệu, giúp xóa nhòa khoảng cách giữa làm việc cá nhân và làm việc tập thể.

Về mặt kỹ thuật, đề tài đặt mục tiêu xây dựng một kiến trúc phần mềm vững chắc, tuân thủ mô hình MVC (Model-View-Controller) kết hợp với lớp Service để đảm bảo tính bảo trì và mở rộng. Hệ thống phải đảm bảo an toàn thông tin người dùng thông qua các cơ chế mã hóa mật khẩu, xác thực qua Email OTP và bảo mật phiên làm việc (Session Management). Đồng thời, giao diện người dùng phải được thiết kế thân thiện, hiện đại, áp dụng phong cách Glassmorphism và có khả năng tương thích tốt trên nhiều thiết bị khác nhau (Responsive Design), mang lại trải nghiệm mượt mà và chuyên nghiệp cho người sử dụng.

**3. Đối tượng và phạm vi nghiên cứu**

Đối tượng nghiên cứu của đề tài tập trung vào các quy trình nghiệp vụ quản lý thời gian cá nhân và làm việc nhóm nhỏ. Hệ thống hướng đến phục vụ hai nhóm đối tượng chính. Nhóm thứ nhất là học sinh, sinh viên, những người cần quản lý lịch học, bài tập lớn và các hoạt động ngoại khóa. Nhóm thứ hai là nhân viên văn phòng hoặc những người làm việc tự do (freelancer), những người thường xuyên phải đối mặt với nhiều dự án cùng lúc và cần một công cụ để sắp xếp thứ tự ưu tiên công việc, đảm bảo đúng hạn (deadline) và phối hợp với đồng nghiệp.

Về phạm vi nghiên cứu, đề tài tập trung phát triển ứng dụng trên nền tảng Web (Web Application). Về mặt nghiệp vụ, hệ thống giới hạn trong việc quản lý Task (công việc), Event (sự kiện lịch), Group Chat (trò chuyện nhóm) và Báo cáo thống kê cơ bản. Các tính năng nâng cao như tích hợp AI để gợi ý lịch trình hay ứng dụng di động (Native App) chưa nằm trong phạm vi hiện thực của đồ án này mà được xem xét là hướng phát triển trong tương lai. Về mặt công nghệ, phạm vi nghiên cứu bao gồm ngôn ngữ JavaScript (chạy trên môi trường Node.js), hệ quản trị cơ sở dữ liệu PostgreSQL và các thư viện hỗ trợ liên quan trong hệ sinh thái NPM.

**4. Cấu trúc đồ án**

Sau phần *Mở đầu*, báo cáo được trình bày trong bốn chương, cụ thể như sau:

Chương 1. *Giới thiệu*

Chương 2. *Phân tích và thiết kế hệ thống*

Chương 3. *Xây dựng Website và kết quả*

Chương 4. *Kết luận*

Cuối cùng là *Tài liệu tham khảo* liên quan đến đề tài.

# CHƯƠNG 1. GIỚI THIỆU

## 1.1. Tổng quan

Trong bối cảnh xã hội hiện đại, nhịp sống ngày càng trở nên hối hả, mỗi cá nhân đều phải đối mặt với một khối lượng công việc và thông tin khổng lồ từ nhiều nguồn khác nhau. Việc quản lý thời gian, sắp xếp lịch trình và theo dõi tiến độ công việc không còn đơn thuần là một kỹ năng bổ trợ mà đã trở thành yếu tố sống còn để đảm bảo hiệu suất làm việc và cân bằng cuộc sống. Sự bùng nổ của công nghệ thông tin đã mang đến nhiều công cụ hỗ trợ, tuy nhiên, người dùng thường gặp phải tình trạng phân mảnh dữ liệu khi phải sử dụng quá nhiều ứng dụng rời rạc cho từng mục đích riêng biệt như ghi chú, lịch biểu hay quản lý dự án. Điều này không những gây bất tiện trong thao tác mà còn làm giảm khả năng tổng hợp và nhìn nhận bức tranh toàn cảnh về công việc của bản thân.

Xuất phát từ thực tiễn đó, đề tài "Xây dựng ứng dụng quản lý lịch trình và công việc cá nhân" được lựa chọn với mong muốn tạo ra một giải pháp tích hợp toàn diện. Ứng dụng này không chỉ đơn thuần là nơi lưu trữ danh sách các việc cần làm mà còn là một trợ lý ảo đắc lực, giúp người dùng lập kế hoạch, theo dõi tiến độ và tối ưu hóa quy trình làm việc cá nhân cũng như cộng tác nhóm. Việc nghiên cứu và phát triển đề tài này cũng là cơ hội để vận dụng những kiến thức đã học về phát triển ứng dụng web, cơ sở dữ liệu và các công nghệ hiện đại vào một sản phẩm thực tế có tính ứng dụng cao.

Hiện nay, trên thị trường đã có nhiều sản phẩm nổi tiếng trong lĩnh vực này như Trello, Asana hay Google Calendar. Trello nổi bật với giao diện Kanban trực quan, Asana mạnh mẽ trong quản lý dự án phức tạp, còn Google Calendar lại là tiêu chuẩn cho việc quản lý lịch trình. Tuy nhiên, mỗi ứng dụng thường chỉ tập trung giải quyết tốt một khía cạnh cụ thể. Ứng dụng trong đề tài này hướng đến việc kết hợp những ưu điểm cốt lõi của các công cụ trên vào một nền tảng duy nhất, đồng thời bổ sung các tính năng cá nhân hóa phù hợp với thói quen của người dùng Việt Nam, tạo nên sự liền mạch trong trải nghiệm quản lý công việc và thời gian.

## 1.2. Công cụ hỗ trợ

Để hiện thực hóa đề tài này, tôi sử dụng các ngôn ngữ và công cụ lập trình phổ biến, mạnh mẽ trong việc phát triển ứng dụng web hiện đại:

* **Ngôn ngữ lập trình & Nền tảng:** Sử dụng JavaScript làm ngôn ngữ chủ đạo cho cả hai phía (Full-stack). Nền tảng Node.js được dùng để xây dựng Server (Backend), tận dụng khả năng xử lý bất đồng bộ hiệu quả cho các tác vụ thời gian thực.
* **Framework & Thư viện Backend:** Sử dụng Express Framework để xây dựng các API và xử lý routing. Các thư viện hỗ trợ gồm Nodemailer (gửi email OTP), Socket.io (xử lý chat/thông báo thời gian thực) và Multer (quản lý upload file).
* **Cơ sở dữ liệu:** Hệ quản trị PostgreSQL được lựa chọn để lưu trữ dữ liệu với độ tin cậy cao, hỗ trợ tốt các ràng buộc quan hệ và kiểu dữ liệu JSONB linh hoạt.
* **Giao diện Frontend:** Sử dụng HTML5, CSS3 (với phong cách Glassmorphism) và EJS (Embedded JavaScript) làm template engine để render giao diện. Logic phía client được xử lý bằng Vanilla JavaScript (JS thuần) giúp tối ưu hiệu năng mà không phụ thuộc framework nặng.

## 1.3. Cấu trúc đồ án

Báo cáo được trình bày một cách hệ thống và logic qua 4 chương chính nhằm trình bày một cách có hệ thống toàn bộ quá trình nghiên cứu và xây dựng ứng dụng:

* **Chương 1: Giới thiệu**

Trình bày tổng quan về lý do chọn đề tài, mục tiêu và phạm vi nghiên cứu. Chương này cũng liệt kê các công cụ, công nghệ hỗ trợ và tóm tắt bố cục của đồ án.

* **Chương 2: Phân tích và thiết kế hệ thống**

Mô tả các yêu cầu chức năng và phi chức năng của hệ thống. Trình bày chi tiết phần thiết kế bao gồm sơ đồ Use Case, biểu đồ Class và thiết kế cơ sở dữ liệu (ERD).

* **Chương 3: Xây dựng Website**

Trình bày quá trình hiện thực hóa các chức năng chính như xác thực, quản lý task, lịch trình và chat. Đánh giá kết quả đạt được thông qua hình ảnh giao diện thực tế của ứng dụng.

* **Chương 4: Kết luận và hướng phát triển**

Tổng kết lại những kết quả đã đạt được của đề tài, đánh giá mức độ hoàn thành so với mục tiêu ban đầu. Đồng thời nêu ra các hạn chế và đề xuất hướng phát triển trong tương lai.

# CHƯƠNG 2. PHÂN TÍCH VÀ THIẾT KẾ HỆ THỐNG

## 2.1. Xác định yêu cầu

### 2.1.1. Yêu cầu của người dùng

Yêu cầu của người dùng được mô tả thông qua các kịch bản sử dụng (User Stories), phản ánh mong muốn và mục đích của từng đối tượng khi tương tác với hệ thống. Trong phạm vi đồ án này, có hai nhóm người dùng chính là Người dùng phổ thông (User) và Quản trị viên (Admin).

Đối với Người dùng phổ thông, nhu cầu cốt lõi xoay quanh việc quản lý hiệu quả các hoạt động cá nhân. Họ mong muốn một quy trình đăng ký và đăng nhập nhanh chóng nhưng phải đảm bảo an toàn. Cụ thể:

* Đăng ký tài khoản và xác thực qua Email để đảm bảo tài khoản là của chính chủ.
* Đăng nhập nhanh bằng tài khoản Google.
* Xác thực 2 lớp (2FA) để bảo vệ thông tin cá nhân tốt hơn.
* Tạo các nhiệm vụ (Task) với đầy đủ thông tin như tiêu đề, mô tả, hạn chót và mức độ ưu tiên để dễ dàng theo dõi.
* Xem và kéo thả công việc trên bảng Kanban để cập nhật trạng thái tiến độ nhanh chóng.
* Xem lịch trình theo tháng hoặc tuần và tạo các sự kiện (Event) để không bỏ lỡ các cuộc họp quan trọng.
* Tạo nhóm chat và gửi tin nhắn để trao đổi công việc với đồng nghiệp.
* Xem biểu đồ thống kê để biết mình đã hoàn thành bao nhiêu phần trăm công việc trong tuần.

Đối với Quản trị viên (Admin), nhu cầu tập trung vào việc vận hành và kiểm soát hệ thống:

* Xem danh sách người dùng để nắm bắt số lượng thành viên đang hoạt động.
* Quản lý các cấu hình hệ thống để đảm bảo ứng dụng vận hành ổn định.

### 2.1.2. Yêu cầu chức năng

Từ việc phân tích các kịch bản người dùng ở trên, hệ thống được xác định bao gồm các nhóm chức năng chính sau đây nhằm đáp ứng đầy đủ các nhu cầu nghiệp vụ.

Nhóm chức năng Xác thực và Quản lý tài khoản: Đây là cổng vào của hệ thống, đảm bảo tính bảo mật và định danh người dùng. Hệ thống cung cấp chức năng Đăng ký với quy trình xác thực mã OTP gửi về Email nhằm ngăn chặn tài khoản ảo. Chức năng Đăng nhập hỗ trợ hai phương thức: tài khoản cục bộ và xác thực qua Google OAuth. Ngoài ra, người dùng có thể Quên mật khẩu để khôi phục quyền truy cập, Đổi mật khẩu và Cập nhật thông tin cá nhân. Đặc biệt, hệ thống hỗ trợ tính năng Bảo mật 2 lớp (2FA) trong phần cài đặt để tăng cường an ninh.

Nhóm chức năng Quản lý Nhiệm vụ (Task Management): Module này cho phép người dùng thực hiện đầy đủ các thao tác CRUD (Thêm, Xem, Sửa, Xóa) đối với công việc. Các nhiệm vụ được phân loại theo trạng thái (Đang làm, Đang tiến hành, Hoàn thành) và mức độ ưu tiên (Thấp, Trung bình, Cao). Điểm nổi bật là chức năng Kanban Board, cho phép người dùng thay đổi trạng thái công việc bằng thao tác kéo thả trực quan. Hệ thống cũng cung cấp công cụ Tìm kiếm và Lọc giúp người dùng nhanh chóng truy xuất công việc theo tiêu chí mong muốn.

Nhóm chức năng Quản lý Lịch trình (Calendar & Event): Chức năng này giúp người dùng quản lý thời gian biểu một cách khoa học. Hệ thống hiển thị Lịch theo các chế độ xem Tháng, Tuần, Ngày. Người dùng có thể Tạo sự kiện mới với các thông tin chi tiết về thời gian bắt đầu, kết thúc, địa điểm và màu sắc nhận diện. Hệ thống tự động kiểm tra logic thời gian để đảm bảo thời gian kết thúc không được nhỏ hơn thời gian bắt đầu.

Nhóm chức năng Cộng tác và Giao tiếp (Chat & Groups): Để hỗ trợ làm việc nhóm, hệ thống cung cấp chức năng Tạo nhóm chat, cho phép người dùng thêm thành viên và trò chuyện trong không gian chung. Người dùng có thể Gửi tin nhắn văn bản và Đính kèm tệp tin (ảnh, tài liệu) để chia sẻ tài nguyên. Danh sách bạn bè và các nhóm chat được quản lý tập trung ở thanh bên (Sidebar) giúp dễ dàng truy cập.

Nhóm chức năng Báo cáo và Thống kê: Hệ thống tự động tổng hợp dữ liệu để cung cấp cái nhìn toàn cảnh về hiệu suất làm việc. Chức năng Dashboard hiển thị các chỉ số quan trọng như số lượng công việc hoàn thành, công việc trễ hạn và các sự kiện sắp tới. Biểu đồ thống kê trực quan giúp người dùng đánh giá được năng suất làm việc của bản thân theo thời gian thực.

### 2.1.3. Yêu cầu phi chức năng

Bên cạnh các chức năng nghiệp vụ, hệ thống cần đáp ứng các tiêu chuẩn khắt khe về chất lượng để đảm bảo trải nghiệm người dùng tốt nhất.

Về Giao diện người dùng (UI/UX), ứng dụng phải có thiết kế hiện đại, áp dụng phong cách Glassmorphism để tạo cảm giác sang trọng và chuyên nghiệp. Giao diện cần thân thiện, dễ sử dụng, bố cục rõ ràng và đặc biệt là phải hỗ trợ Responsive Design, hiển thị tốt trên cả máy tính cá nhân, máy tính bảng và điện thoại di động.

Về Hiệu năng và Độ ổn định, hệ thống cần đảm bảo tốc độ phản hồi nhanh, các thao tác chuyển trang hoặc gọi API không được có độ trễ quá lớn (dưới 2 giây trong điều kiện mạng ổn định). Các tính năng tương tác thời gian thực như cập nhật trạng thái Kanban hay nhận tin nhắn Chat phải diễn ra gần như tức thì để không làm gián đoạn luồng công việc của người dùng.

Về Bảo mật và An toàn dữ liệu, mật khẩu người dùng bắt buộc phải được mã hóa (sử dụng thuật toán như Bcrypt) trước khi lưu xuống cơ sở dữ liệu. Hệ thống cần có cơ chế kiểm soát phiên làm việc (Session Management) chặt chẽ, tự động đăng xuất khi hết phiên. Các dữ liệu nhạy cảm và quyền truy cập vào tài nguyên phải được phân quyền rõ ràng, đảm bảo người dùng chỉ có thể thao tác trên dữ liệu của chính mình.

### 2.1.4. Yêu cầu hệ thống

Để triển khai và vận hành ứng dụng "Quản Lý Lịch Trình", môi trường phần cứng và phần mềm cần đáp ứng các yêu cầu kỹ thuật sau:

Môi trường Server (Máy chủ):

* Hệ điều hành: Có thể chạy trên đa nền tảng như Windows Server, Linux (Ubuntu, CentOS) hoặc macOS.
* Nền tảng thực thi: Yêu cầu cài đặt Node.js (phiên bản 14.x trở lên) để chạy mã nguồn Backend.
* Cơ sở dữ liệu: Hệ quản trị cơ sở dữ liệu PostgreSQL (phiên bản 12.x trở lên) để lưu trữ dữ liệu bền vững.

Môi trường Client (Máy trạm):

* Thiết bị: Máy tính để bàn, laptop hoặc các thiết bị di động thông minh (Smartphone, Tablet).
* Trình duyệt web: Ứng dụng hoạt động tốt nhất trên các trình duyệt hiện đại hỗ trợ HTML5 và CSS3 như Google Chrome, Mozilla Firefox, Microsoft Edge, hoặc Safari. Không yêu cầu cài đặt thêm phần mềm hỗ trợ nào khác.
* Kết nối mạng: Yêu cầu kết nối Internet ổn định để thực hiện các tác vụ đồng bộ dữ liệu và giao tiếp thời gian thực.

## 2.2. Phân tích hệ thống

### 2.2.1. Usecase Diagram

#### 2.2.1.1. Đặc tả Usecase

Bảng 1. Đặc tả cho Usecase Đăng ký/Đăng nhập

|  |  |
| --- | --- |
| **Tên Usecase** | **Đăng ký / Đăng nhập** |
| **Tác nhân** | Người dùng, Admin |
| **Mục đích** | Xác thực danh tính để truy cập vào hệ thống. |
| **Điều kiện trước** | Người dùng truy cập vào địa chỉ trang web. |
| **Luồng sự kiện chính** | 1. Người dùng chọn chức năng "Đăng nhập" hoặc "Đăng ký".  2. Hệ thống hiển thị form nhập liệu.  3. Người dùng nhập thông tin (Username, Password, Email...).  4. Người dùng nhấn nút xác nhận.  5. Hệ thống kiểm tra tính hợp lệ của dữ liệu (định dạng email, độ mạnh mật khẩu).  6. Hệ thống xác thực với cơ sở dữ liệu.  7. Hệ thống thông báo thành công và chuyển hướng vào Trang chủ. |
| **Luồng ngoại lệ** | - Nếu tên đăng nhập/mật khẩu sai: Hệ thống báo lỗi và yêu cầu nhập lại.  - Nếu tài khoản bị khóa: Hệ thống từ chối truy cập và thông báo liên hệ Admin. |
| **Điều kiện sau** | Người dùng được cấp quyền truy cập (Session/Token) để sử dụng các chức năng khác. |

Bảng 2. Đặc tả cho Usecase Quản lý tài khoản & Cài đặt

|  |  |
| --- | --- |
| **Tên Usecase** | **Quản lý tài khoản & Cài đặt** |
| **Tác nhân** | Người dùng, Admin |
| **Mục đích** | Cập nhật thông tin cá nhân, bảo mật và tùy chỉnh trải nghiệm hệ thống. |
| **Điều kiện trước** | Người dùng đã đăng nhập thành công vào hệ thống. |
| **Luồng sự kiện chính** | 1. Người dùng chọn menu "Cài đặt" hoặc nhấp vào Avatar.  2. Hệ thống hiển thị trang quản lý hồ sơ.  3. **Cập nhật thông tin:** Người dùng sửa tên hiển thị, cập nhật ảnh đại diện (Avatar) và nhấn "Lưu".  4. **Cài đặt hệ thống:** Người dùng chuyển sang tab Cài đặt để chọn Ngôn ngữ (Tiếng Việt/Anh) hoặc Giao diện (Sáng/Tối).  5. Hệ thống lưu thay đổi và áp dụng ngay lập tức.  6. Hệ thống thông báo "Cập nhật thành công" |
| **Luồng phụ** | **Đổi mật khẩu:**  - Người dùng chọn chức năng "Đổi mật khẩu".  - Hệ thống yêu cầu nhập mật khẩu cũ, mật khẩu mới và xác nhận mật khẩu mới.  - Hệ thống kiểm tra mật khẩu cũ. Nếu đúng, hệ thống cập nhật mật khẩu mới và thông báo thành công. |
| **Điều kiện sau** | Thông tin hồ sơ, cấu hình hiển thị hoặc mật khẩu mới được cập nhật vào cơ sở dữ liệu. |

Bảng 3. Đặc tả cho Usecase Quản lý nhiệm vụ

|  |  |
| --- | --- |
| **Tên Usecase** | **Quản lý nhiệm vụ** |
| **Tác nhân** | Người dùng |
| **Mục đích** | Tạo lập và theo dõi tiến độ các công việc cá nhân. |
| **Điều kiện trước** | Người dùng đã đăng nhập thành công. |
| **Luồng sự kiện chính** | 1. Người dùng chọn menu "Quản lý nhiệm vụ".  2. Hệ thống hiển thị danh sách nhiệm vụ (dạng List hoặc Kanban).  3. **Thêm mới:** Người dùng nhập tiêu đề, mô tả, hạn chót, độ ưu tiên và nhấn Lưu.  4. **Cập nhật trạng thái (Kanban):** Người dùng kéo thả thẻ nhiệm vụ từ cột này sang cột khác (VD: Từ "Đang làm" sang "Hoàn thành").  5. Hệ thống cập nhật dữ liệu vào CSDL.  6. Hệ thống thông báo cập nhật thành công. |
| **Luồng phụ** | - Tìm kiếm nhiệm vụ: Người dùng nhập từ khóa, hệ thống lọc và trả về kết quả tương ứng.  - Xóa nhiệm vụ: Người dùng chọn xóa, hệ thống yêu cầu xác nhận trước khi xóa vĩnh viễn. |
| **Điều kiện sau** | Dữ liệu nhiệm vụ được cập nhật chính xác trong hệ thống. |

Bảng 4. Đặc tả cho Usecsae Quản lý lịch trình

|  |  |
| --- | --- |
| **Tên Usecase** | **Quản lý lịch trình** |
| **Tác nhân** | Người dùng |
| **Mục đích** | Sắp xếp thời gian biểu và thiết lập nhắc nhở tự động. |
| **Điều kiện trước** | Người dùng đã đăng nhập thành công. |
| **Luồng sự kiện chính** | 1. Người dùng chọn menu "Lịch trình".  2. Hệ thống hiển thị giao diện lịch (Calendar View).  3. Người dùng chọn một ngày hoặc khung giờ cụ thể.  4. Người dùng nhập thông tin sự kiện (Tên, Thời gian bắt đầu/kết thúc, Địa điểm).  5. **[Mở rộng] Cài đặt nhắc nhở:** Người dùng chọn mốc thời gian nhắc trước (ví dụ: 15 phút) và loại chuông báo.  6. Người dùng nhấn "Lưu".  7. Hệ thống lưu sự kiện và thiết lập trigger nhắc nhở. |
| **Điều kiện sau** | Sự kiện hiển thị trên lịch. Hệ thống sẽ gửi thông báo khi đến giờ hẹn. |

Bảng 5. Đặc tả cho Usecase Tin nhắn & Chia sẻ

|  |  |
| --- | --- |
| **Tên Usecase** | **Tin nhắn & Chia sẻ** |
| **Tác nhân** | Người dùng |
| **Mục đích** | Trao đổi thông tin và cộng tác làm việc với người dùng khác. |
| **Điều kiện trước** | Người dùng đã đăng nhập và có thông tin của người cần liên hệ. |
| **Luồng sự kiện chính** | **A. Nhắn tin:**  1. Người dùng chọn đối tượng chat (1v1 hoặc Nhóm).  2. Nhập nội dung hoặc đính kèm tệp tin và nhấn Gửi.  3. Hệ thống chuyển tin nhắn đến người nhận.  **B. Chia sẻ lịch trình:**  1. Người dùng chọn một sự kiện lịch trình.  2. Chọn chức năng "Chia sẻ".  3. Chọn phương thức chia sẻ (Sao chép liên kết hoặc Đồng bộ).  4. Thiết lập phân quyền (Chỉ xem hoặc Được chỉnh sửa).  5. Hệ thống cấp quyền truy cập cho người được chia sẻ. |
| **Điều kiện sau** | Tin nhắn được lưu và hiển thị. Quyền truy cập lịch trình được cập nhật cho người nhận. |

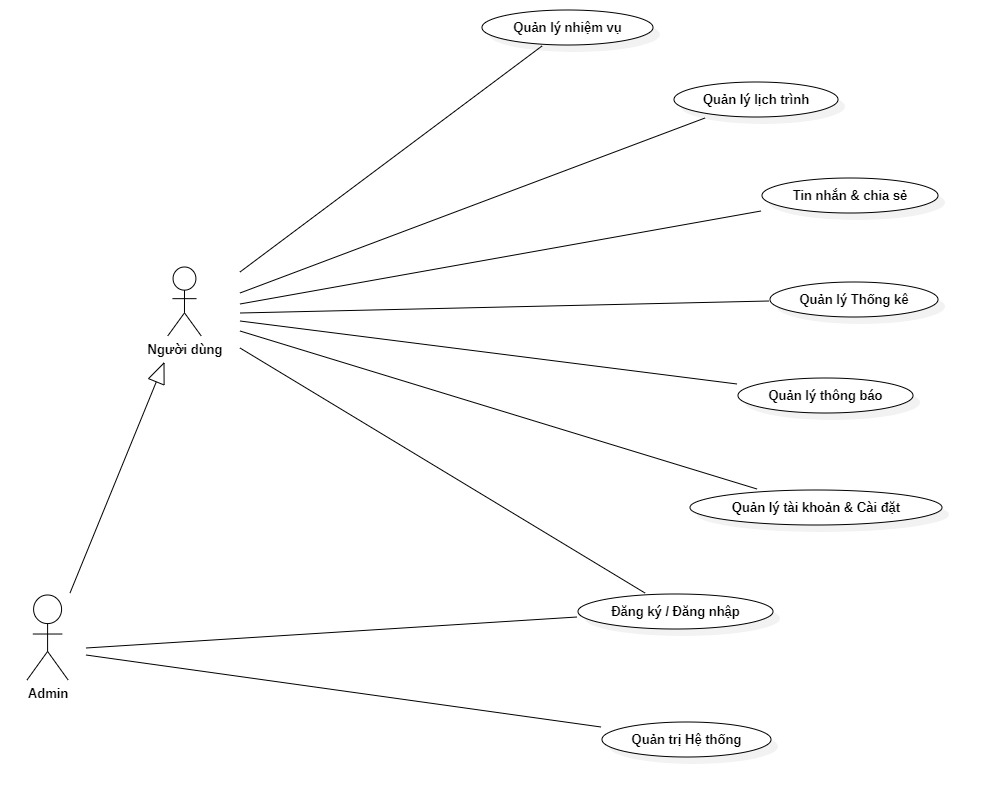
Bảng 6. Đặc tả cho Usecase Quản lý Thống kê

|  |  |
| --- | --- |
| **Tên Usecase** | **Quản lý Thống kê** |
| **Tác nhân** | Người dùng |
| **Mục đích** | Theo dõi hiệu suất làm việc cá nhân. |
| **Điều kiện trước** | Người dùng đã có dữ liệu hoạt động (Nhiệm vụ/Lịch trình) trong quá khứ. |
| **Luồng sự kiện chính** | 1. Người dùng chọn menu "Thống kê".  2. Hệ thống tổng hợp dữ liệu từ database.  3. Hệ thống hiển thị các biểu đồ: Tỷ lệ hoàn thành nhiệm vụ, Phân bố thời gian lịch trình.  4. Người dùng chọn chức năng "Xuất báo cáo".  5. Hệ thống xuất file (Word/Excel) chứa thông tin thống kê chi tiết. |

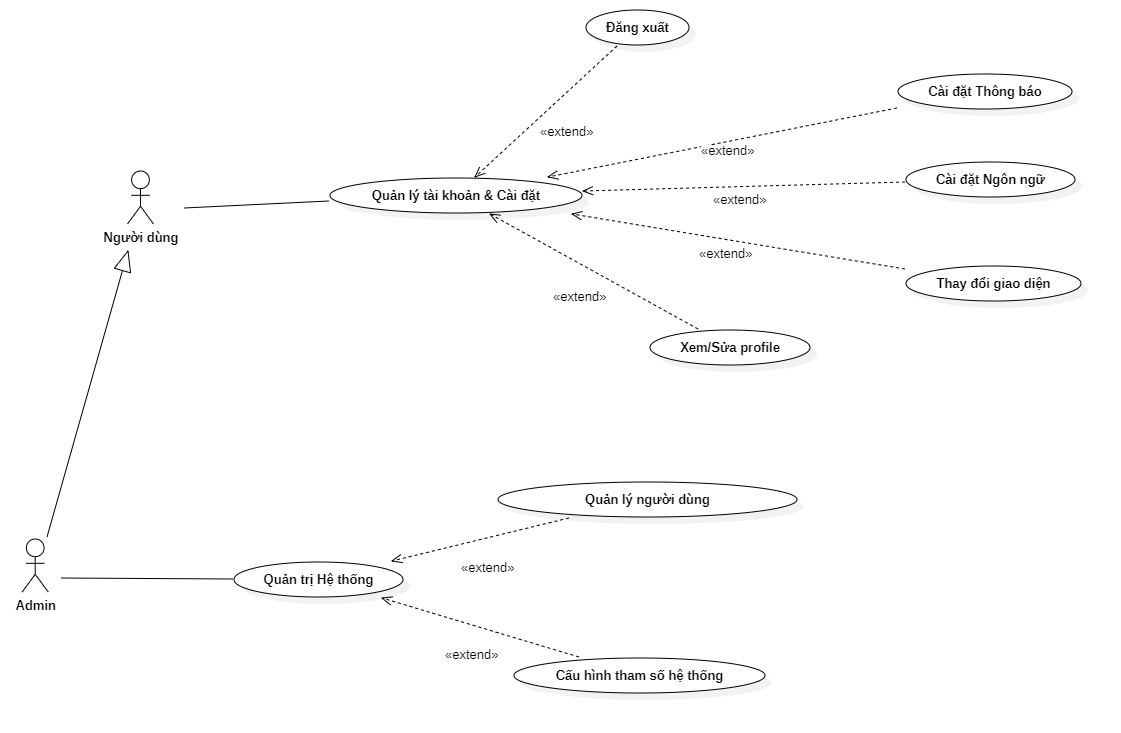
Bảng 7. Đặc tả cho Usecase Quản trị Hệ thống

|  |  |
| --- | --- |
| **Tên Use Case** | **Quản trị Hệ thống** |
| **Tác nhân** | Quản trị viên (Admin) |
| **Mục đích** | Duy trì hoạt động và kiểm soát người dùng hệ thống. |
| **Điều kiện trước** | Đăng nhập với quyền Admin. |
| **Luồng sự kiện chính** | 1. Admin truy cập trang Dashboard quản trị.  2. **Quản lý người dùng:** Admin xem danh sách, tìm kiếm người dùng. Admin chọn khóa/mở khóa hoặc reset mật khẩu cho tài khoản vi phạm.  3. **Cấu hình hệ thống:** Admin thay đổi các tham số kỹ thuật chung.  4. Hệ thống cập nhật trạng thái mới vào cơ sở dữ liệu. |
| **Điều kiện sau** | Trạng thái tài khoản người dùng hoặc cấu hình hệ thống được thay đổi. |

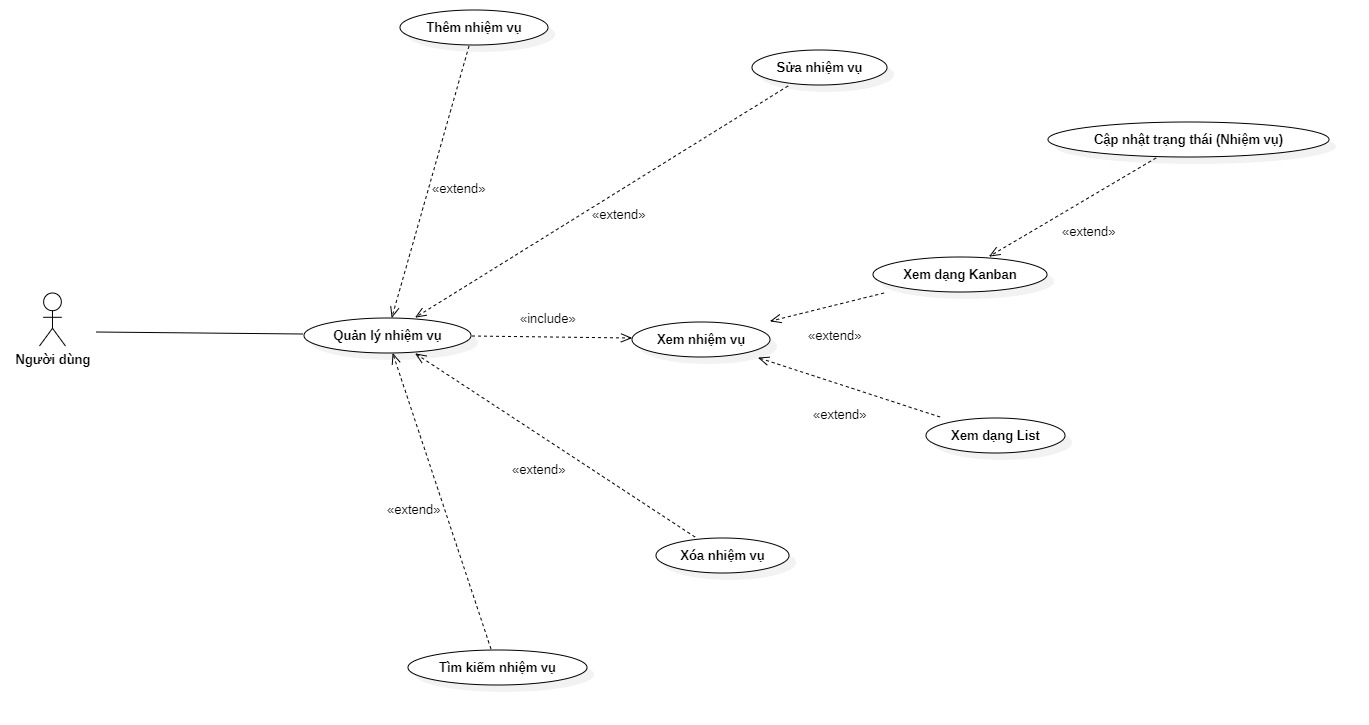
#### 2.2.1.2. Các biểu đồ Usecase



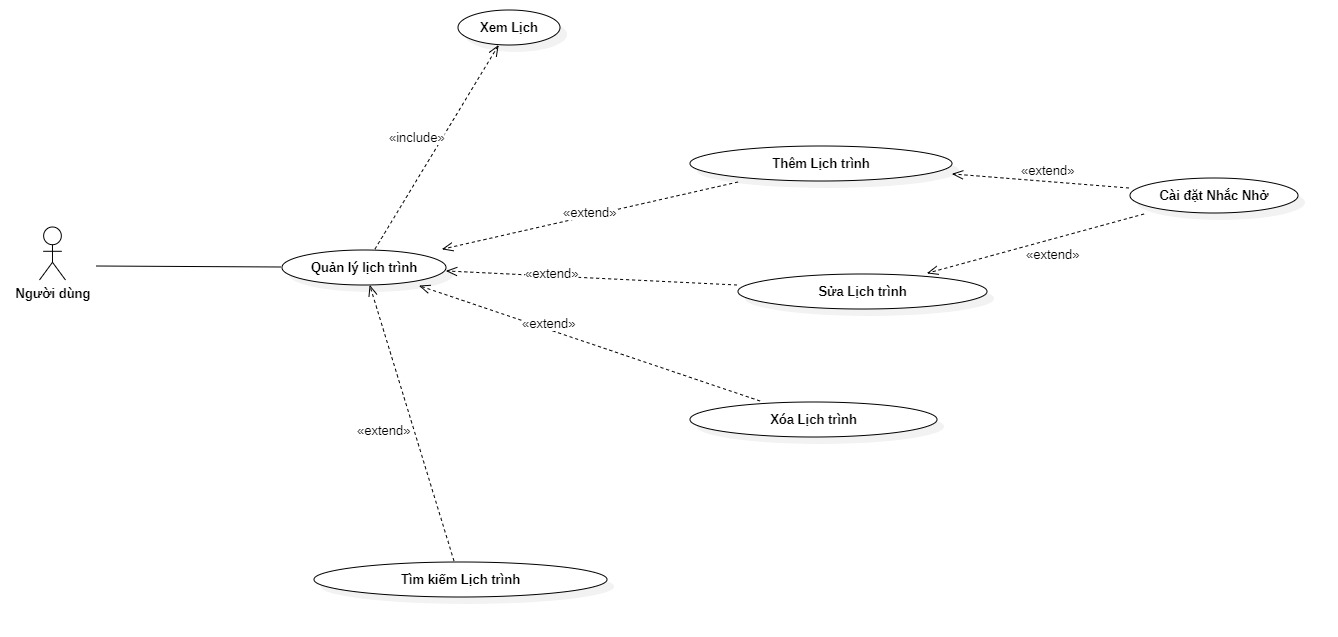
Hình 1. Biểu đồ Usecase Tổng quát



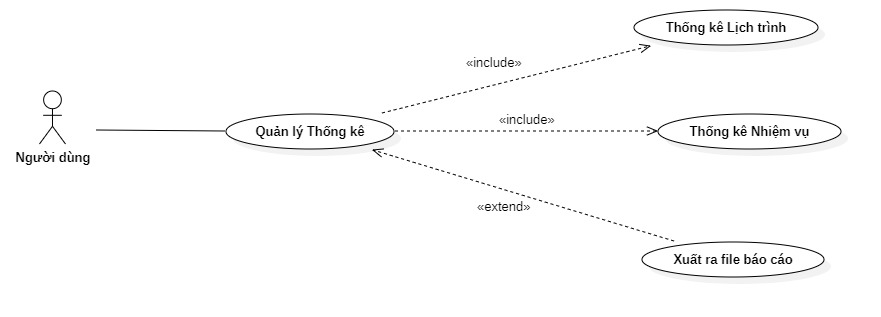
Hình 2. Biểu đồ Usecase Quản lý tài khoản & Quản trị Hệ thống



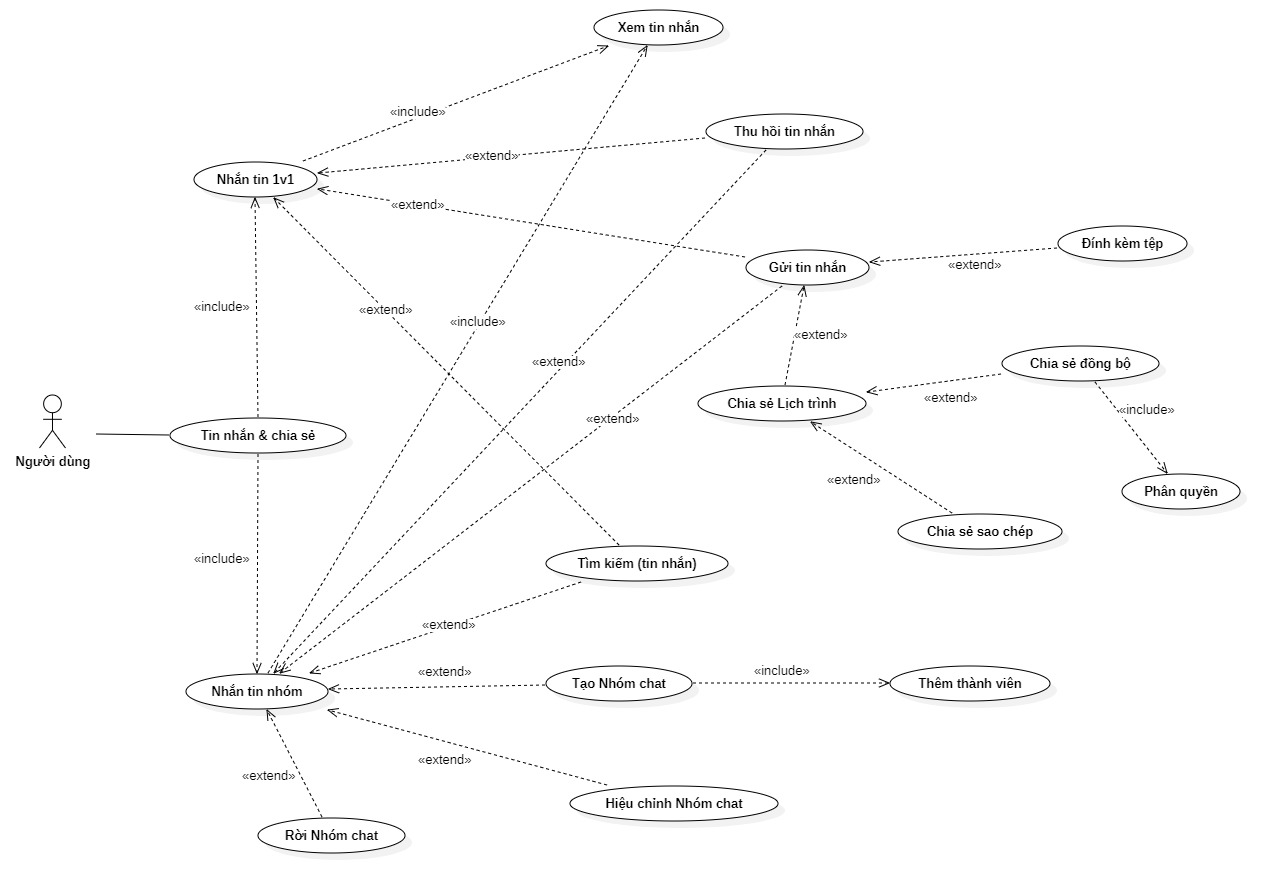
Hình 3. Biểu đồ Usecase Quản lý nhiệm vụ



Hình 4. Biểu đồ Usecase Quản lý lịch trình

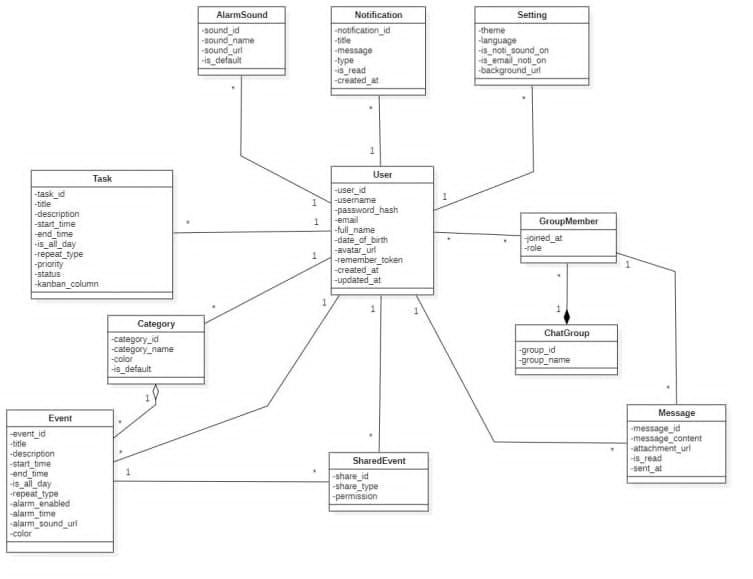


Hình 5. Biểu đồ Usecase Quản lý Thống kê



Hình 6. Biểu đồ Usecase Nhắn tin & Chia sẻ

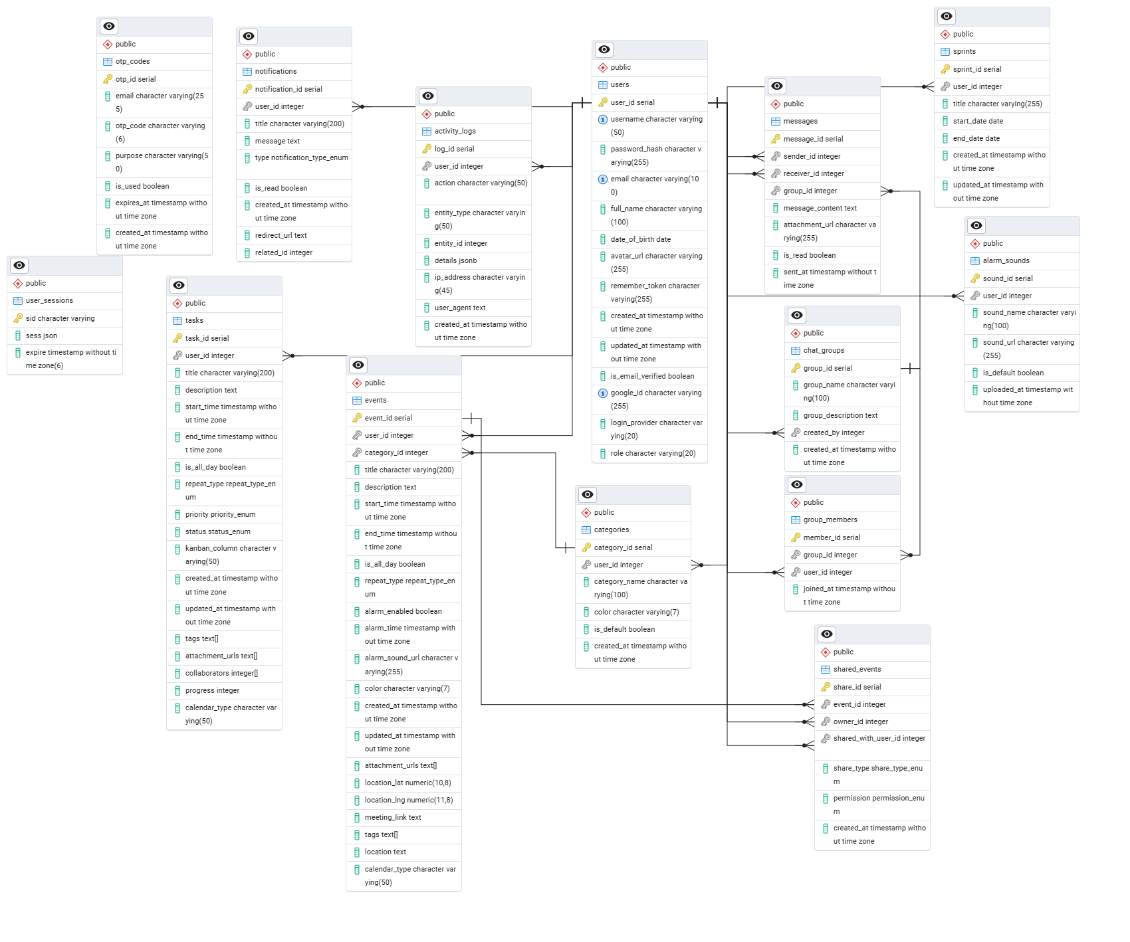
### 2.2.2. Class Diagram



Hình 7. Biểu đồ Class

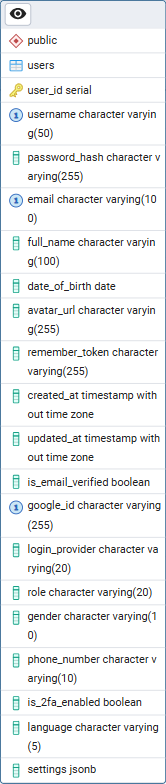
## 2.3. Thiết kế hệ thống

### 2.3.1. Thiết kế CSDL ERD



Hình 8. Biểu đồ ERD

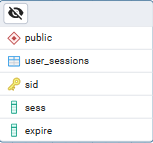
#### 2.3.1.1. Đặc tả chi tiết các bảng CSDL



Hình 9. Bảng users

Bảng users được thiết kế để lưu trữ toàn bộ thông tin tài khoản và hồ sơ cá nhân của người dùng trong hệ thống. Đây là bảng trung tâm kết nối với hầu hết các nghiệp vụ khác.

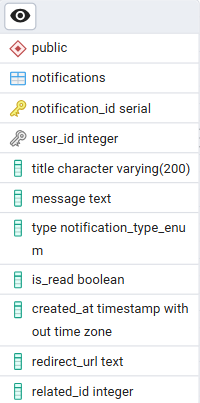
* **user\_id**: Mã định danh duy nhất của người dùng. Đây là Khóa chính (Primary Key).
* **username**: Tên đăng nhập.
* **password\_hash**: Mật khẩu đã được mã hóa (dùng Bcrypt).
* **email**: Địa chỉ email dùng để xác thực và liên lạc.
* **full\_name**: Tên đầy đủ của người dùng.
* **date\_of\_birth**: Ngày sinh.
* **avatar\_url**: Đường dẫn đến ảnh đại diện của người dùng.
* **remember\_token**: Token dùng cho tính năng "Ghi nhớ đăng nhập".
* **created\_at**: Thời điểm tài khoản được tạo.
* **updated\_at**: Thời điểm cập nhật thông tin gần nhất.
* **is\_email\_verified**: Trạng thái xác thực email.
* **google\_id**: ID nếu người dùng đăng nhập qua Google OAuth.
* **login\_provider**: Phương thức đăng nhập (local, google).
* **role**: Vai trò người dùng trong hệ thống (User/Admin).
* **gender**: Giới tính.
* **phone\_number**: Số điện thoại.
* **is\_2fa\_enabled**: Trạng thái bật/tắt bảo mật 2 lớp (2FA).
* **language**: Ngôn ngữ giao diện người dùng.
* **settings**: Lưu trữ các cài đặt cá nhân khác dưới dạng jsonb.



Hình 10. Bảng user\_sessions

Bảng user\_sessions được sử dụng để quản lý phiên làm việc của người dùng. Bảng này rất quan trọng trong việc duy trì trạng thái đăng nhập và tăng cường bảo mật phiên làm việc (Session Management).

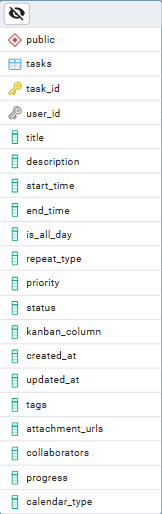
* **sid**: Session ID. Đây là Khóa chính (Primary Key).
* **sess**: Dữ liệu phiên làm việc được mã hóa.
* **expire**: Thời điểm phiên làm việc hết hạn.



Hình 11. Bảng notifications

Bảng notifications được dùng để lưu trữ và quản lý các thông báo được gửi đến người dùng, đảm bảo người dùng không bỏ lỡ thông tin quan trọng như nhắc nhở sự kiện, tin nhắn mới, hoặc cập nhật trạng thái công việc.

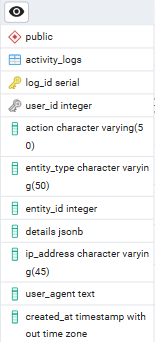
* **notification\_id**: Mã định danh duy nhất của thông báo. Đây là Khóa chính.
* **user\_id**: Khóa ngoại liên kết với người nhận thông báo.
* **title**: Tiêu đề ngắn của thông báo.
* **message**: Nội dung chi tiết của thông báo.
* **type**: Loại thông báo (ví dụ: task\_update, chat\_message, alarm).
* **is\_read**: Trạng thái đã đọc hay chưa.
* **created\_at**: Thời điểm thông báo được tạo.
* **redirect\_url**: Đường dẫn chuyển hướng khi người dùng nhấp vào thông báo.
* **related\_id**: ID của thực thể liên quan (ví dụ: task\_id nếu là thông báo về nhiệm vụ).



Hình 12. Bảng tasks

Bảng tasks lưu trữ dữ liệu chi tiết về các nhiệm vụ (Task) cá nhân , hỗ trợ quản lý công việc trên bảng Kanban và dưới dạng danh sách.

* **task\_id**: Mã định danh duy nhất của nhiệm vụ. Đây là Khóa chính.
* **user\_id**: Khóa ngoại liên kết với người tạo nhiệm vụ.
* **title**: Tiêu đề của nhiệm vụ.
* **description**: Mô tả chi tiết.
* **start\_time**: Thời gian bắt đầu.
* **end\_time**: Thời gian hạn chót.
* **is\_all\_day**: Xác định nhiệm vụ kéo dài cả ngày.
* **repeat\_type**: Loại lặp lại (nếu có).
* **priority**: Mức độ ưu tiên (Low, Medium, High).
* **status**: Trạng thái hiện tại của nhiệm vụ.
* **kanban\_column**: Cột Kanban mà nhiệm vụ đang nằm trong (To Do, In Progress, Done).
* **created\_at**: Thời điểm tạo nhiệm vụ.
* **updated\_at**: Thời điểm cập nhật cuối cùng.
* **tags**: Danh sách tag/nhãn dưới dạng mảng text.
* **attachment\_urls**: Mảng các đường dẫn file đính kèm.
* **collaborators**: Mảng ID người dùng cộng tác.
* **progress**: Phần trăm hoàn thành.
* **calendar\_type**: Loại lịch (ví dụ: Personal, Work).



Hình 13. Bảng activity\_logs

Bảng activity\_logs được sử dụng để ghi lại lịch sử hoạt động của người dùng trong hệ thống, phục vụ cho mục đích bảo mật, kiểm tra và hiển thị các hoạt động gần đây của người dùng.

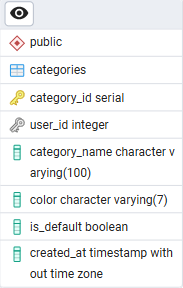
* **log\_id**: Mã định danh duy nhất của log. Đây là Khóa chính.
* **user\_id**: Khóa ngoại liên kết với người thực hiện hành động.
* **action**: Loại hành động được thực hiện (ví dụ: TASK\_CREATE, LOGIN\_SUCCESS).
* **entity\_type**: Loại thực thể bị ảnh hưởng (ví dụ: Task, Event, User).
* **entity\_id**: ID của thực thể bị ảnh hưởng.
* **details**: Thông tin chi tiết về sự thay đổi dưới dạng jsonb.
* **ip\_address**: Địa chỉ IP thực hiện hành động.
* **user\_agent**: Thông tin trình duyệt/thiết bị người dùng.
* **created\_at**: Thời điểm ghi log.



Hình 14. Bảng alarm sounds

Bảng alarm\_sounds lưu trữ thông tin về các âm thanh báo thức mà người dùng có thể lựa chọn cho các sự kiện của mình.

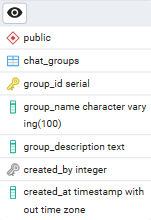
* **sound\_id**: Mã định danh duy nhất của âm thanh. Đây là Khóa chính.
* **user\_id**: Khóa ngoại liên kết với người tải lên (nếu là âm thanh tùy chỉnh).
* **sound\_name**: Tên của âm thanh.
* **sound\_url**: Đường dẫn URL đến file âm thanh.
* **is\_default**: Xác định đây có phải là âm thanh mặc định của hệ thống hay không.
* **uploaded\_at**: Thời điểm âm thanh được tải lên.



Hình 15. Bảng categories

Bảng categories được sử dụng để phân loại các sự kiện (events) và nhiệm vụ (tasks) của người dùng. Việc phân loại giúp người dùng quản lý lịch trình một cách có tổ chức hơn và dễ dàng lọc, thống kê công việc theo từng nhóm (ví dụ: Công việc, Cá nhân, Học tập).

* **category\_id**: Mã định danh duy nhất của danh mục. Đây là Khóa chính.
* **user\_id**: Khóa ngoại liên kết với người tạo danh mục (nếu là danh mục tùy chỉnh).
* **category\_name**: Tên của danh mục (ví dụ: "Công việc", "Sức khỏe").
* **color\_code**: Mã màu (hex code, ví dụ: #FF5733) đại diện cho danh mục trên giao diện lịch.
* **icon\_name**: Tên biểu tượng (icon) đại diện cho danh mục.
* **is\_system\_default**: Xác định đây có phải là danh mục mặc định của hệ thống hay không.
* **created\_at**: Thời điểm danh mục được tạo.



Hình 16. Bảng chat\_groups

Bảng chat\_groups quản lý thông tin về các nhóm trò chuyện được tạo ra để phục vụ tính năng cộng tác và làm việc nhóm giữa các người dùng.

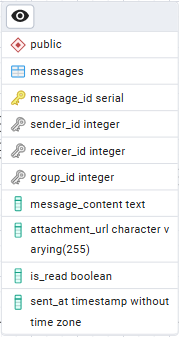
* **group\_id**: Mã định danh duy nhất của nhóm trò chuyện. Đây là Khóa chính.
* **group\_name**: Tên của nhóm trò chuyện.
* **creator\_id**: Khóa ngoại liên kết với ID người tạo nhóm.
* **description**: Mô tả ngắn gọn về mục đích của nhóm.
* **created\_at**: Thời điểm nhóm được tạo.
* **updated\_at**: Thời điểm nhóm có hoạt động/tin nhắn cuối cùng.



Hình 17. Bảng group\_members

Bảng group\_members là bảng trung gian (Junction Table) dùng để thiết lập mối quan hệ N-N (Nhiều-Nhiều) giữa Bảng users và Bảng chat\_groups, xác định những người dùng nào là thành viên của nhóm chat nào.

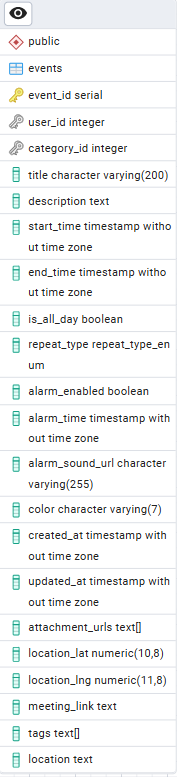
* **group\_member\_id**: Khóa chính cho bảng trung gian.
* **group\_id**: Khóa ngoại liên kết với nhóm trò chuyện.
* **user\_id**: Khóa ngoại liên kết với thành viên.
* **joined\_at**: Thời điểm người dùng tham gia nhóm.
* **role**: Vai trò của thành viên trong nhóm (ví dụ: member, admin).



Hình 18. Bảng messages

Bảng messages lưu trữ toàn bộ nội dung tin nhắn trao đổi trong các nhóm chat hoặc giữa các người dùng cá nhân (nếu hệ thống hỗ trợ chat cá nhân), phục vụ tính năng làm việc nhóm theo thời gian thực (real-time).

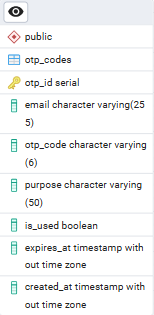
* **message\_id**: Mã định danh duy nhất của tin nhắn. Đây là Khóa chính.
* **group\_id**: Khóa ngoại liên kết với nhóm chat mà tin nhắn được gửi đến.
* **sender\_id**: Khóa ngoại liên kết với người gửi tin nhắn.
* **content**: Nội dung của tin nhắn (text).
* **attachments**: Mảng các đường dẫn file đính kèm trong tin nhắn.
* **sent\_at**: Thời điểm tin nhắn được gửi đi.
* **is\_edited**: Trạng thái xác định tin nhắn đã bị chỉnh sửa hay chưa.



Hình 19. Bảng messages

Bảng events là bảng cốt lõi, lưu trữ thông tin về các sự kiện lịch trình (Event) có thời gian xác định, là nền tảng cho chức năng lịch và nhắc nhở chính của website.

* **event\_id**: Mã định danh duy nhất của sự kiện. Đây là Khóa chính.
* **user\_id**: Khóa ngoại liên kết với người tạo sự kiện.
* **category\_id**: Khóa ngoại liên kết với danh mục của sự kiện.
* **title**: Tiêu đề sự kiện.
* **description**: Mô tả chi tiết.
* **start\_time**: Thời gian bắt đầu (bắt buộc).
* **end\_time**: Thời gian kết thúc (bắt buộc).
* **is\_all\_day**: Xác định sự kiện kéo dài cả ngày.
* **repeat\_type**: Loại lặp lại của sự kiện (ví dụ: daily, weekly).
* **alarm\_enabled**: Bật/tắt tính năng báo thức.
* **alarm\_time**: Thời điểm chuông báo thức reo.
* **alarm\_sound\_url**: Đường dẫn file âm thanh báo thức.
* **color**: Mã màu hiển thị trên lịch.
* **location\_text**: Tên địa điểm (dạng văn bản).
* **meeting\_link**: Liên kết họp trực tuyến (nếu có).
* **attachment\_urls**: Mảng các đường dẫn file đính kèm.
* **tags**: Danh sách tag/nhãn.
* **created\_at**: Thời điểm tạo sự kiện.
* **updated\_at**: Thời điểm cập nhật cuối cùng.



Hình 20. Bảng messages

Bảng otp\_codes được sử dụng để lưu trữ mã xác thực một lần (One-Time Password - OTP) cho các nghiệp vụ yêu cầu bảo mật như đặt lại mật khẩu hoặc xác thực 2 bước (2FA).

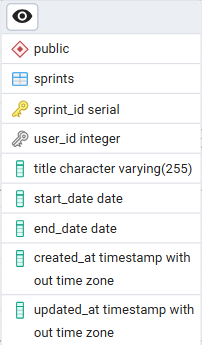
* **otp\_id**: Mã định danh duy nhất. Đây là Khóa chính.
* **user\_id**: Khóa ngoại liên kết với người dùng yêu cầu mã OTP.
* **otp\_code**: Mã OTP được tạo ra.
* **type**: Loại nghiệp vụ sử dụng mã OTP (ví dụ: password\_reset, email\_verification).
* **created\_at**: Thời điểm mã được tạo.
* **expires\_at**: Thời điểm mã hết hạn (thường là sau vài phút).
* **is\_used**: Trạng thái xác định mã đã được sử dụng hay chưa.



Hình 21. Bảng shared\_events

Bảng shared\_events là bảng trung gian ghi lại chi tiết việc chia sẻ lịch trình/sự kiện giữa người dùng, xác định sự kiện nào được chia sẻ với ai và với quyền hạn gì.

* **share\_id**: Mã định danh duy nhất của bản ghi chia sẻ. Đây là Khóa chính.
* **event\_id**: Khóa ngoại liên kết với sự kiện được chia sẻ.
* **shared\_by\_user\_id**: Khóa ngoại liên kết với người chia sẻ.
* **shared\_to\_user\_id**: Khóa ngoại liên kết với người nhận chia sẻ.
* **permission\_level**: Mức độ quyền hạn (ví dụ: read\_only, edit).
* **shared\_at**: Thời điểm sự kiện được chia sẻ.
* **is\_accepted**: Trạng thái người nhận đã chấp nhận lịch chia sẻ hay chưa.

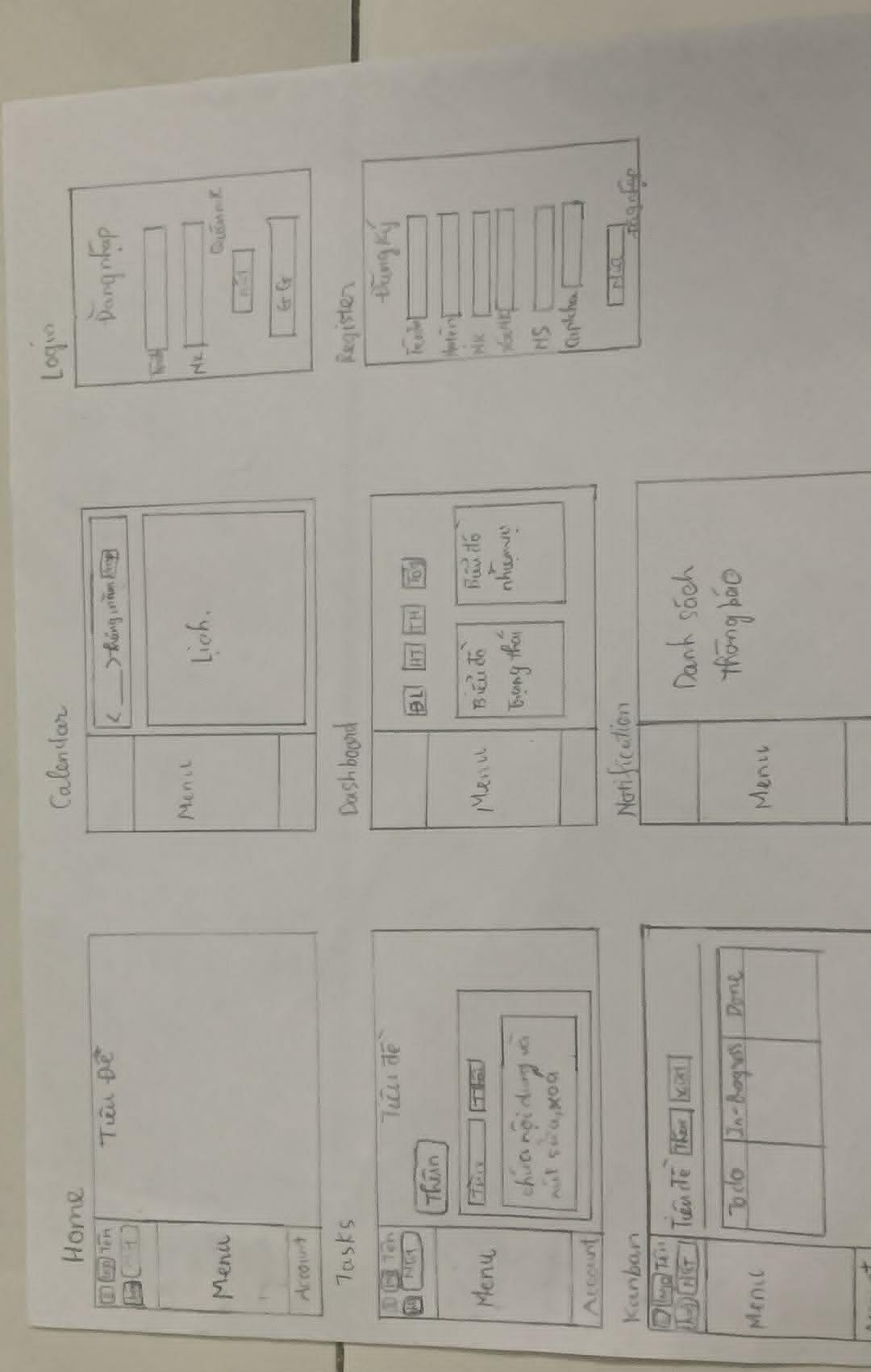


Hình 22. Bảng sprints

Bảng sprints được sử dụng để quản lý các chu kỳ làm việc ngắn hạn (Sprint) trong khuôn khổ phát triển phần mềm hoặc quản lý dự án, giúp nhóm theo dõi tiến độ công việc trong các khoảng thời gian cố định.

* **sprint\_id**: Mã định danh duy nhất của sprint. Đây là Khóa chính.
* **group\_id**: Khóa ngoại liên kết với nhóm/dự án đang thực hiện sprint.
* **sprint\_name**: Tên của sprint.
* **start\_date**: Ngày bắt đầu sprint.
* **end\_date**: Ngày kết thúc sprint.
* **status**: Trạng thái của sprint (planning, in\_progress, completed).
* **goal**: Mục tiêu cần đạt được trong sprint.
* **created\_at**: Thời điểm tạo sprint.

### 2.3.2. Phác thảo website

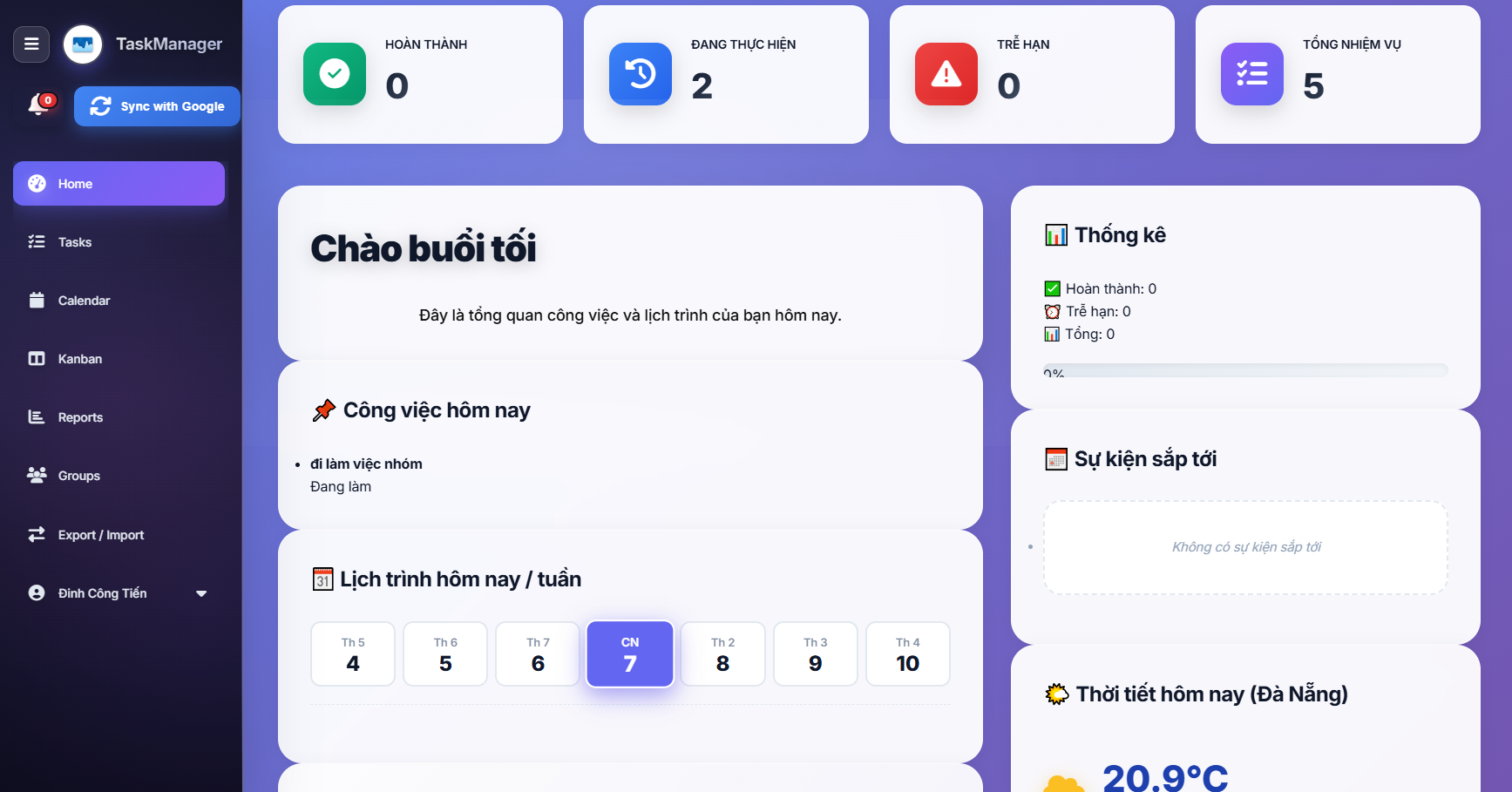


Hình 23. Phác thảo website

# CHƯƠNG 3. XÂY DỰNG WEBSITE VÀ KẾT QUẢ

## 3.1. Giao diện Home

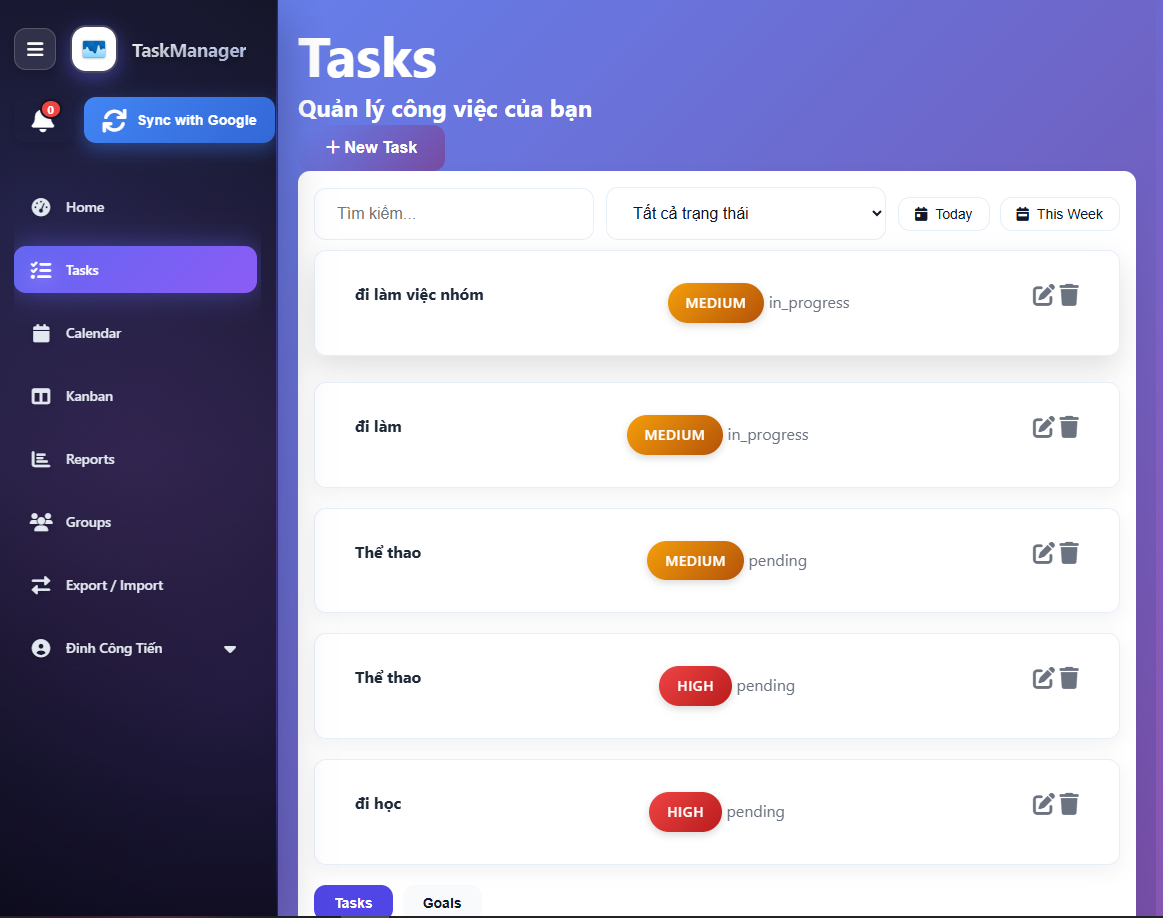
Giao diện Home là trang chính mà người dùng truy cập ngay sau khi đăng nhập thành công. Trang này cung cấp cái nhìn tổng quan về hoạt động cá nhân, bao gồm danh sách các nhiệm vụ sắp tới (upcoming tasks), sự kiện lịch gần nhất, và các thông báo mới từ nhóm chat hoặc cập nhật hệ thống. Thanh sidebar bên trái cho phép truy cập nhanh đến các module khác như Tasks, Calendar, và Chat. Phần chính giữa hiển thị dashboard với các widget thống kê cơ bản như tỷ lệ hoàn thành nhiệm vụ trong tuần. Giao diện được tối ưu để hiển thị nhanh chóng, giúp người dùng nắm bắt tình hình công việc chỉ trong vài giây.



Hình 24. Giao diện Home

## 3.2. Giao diện Tasks

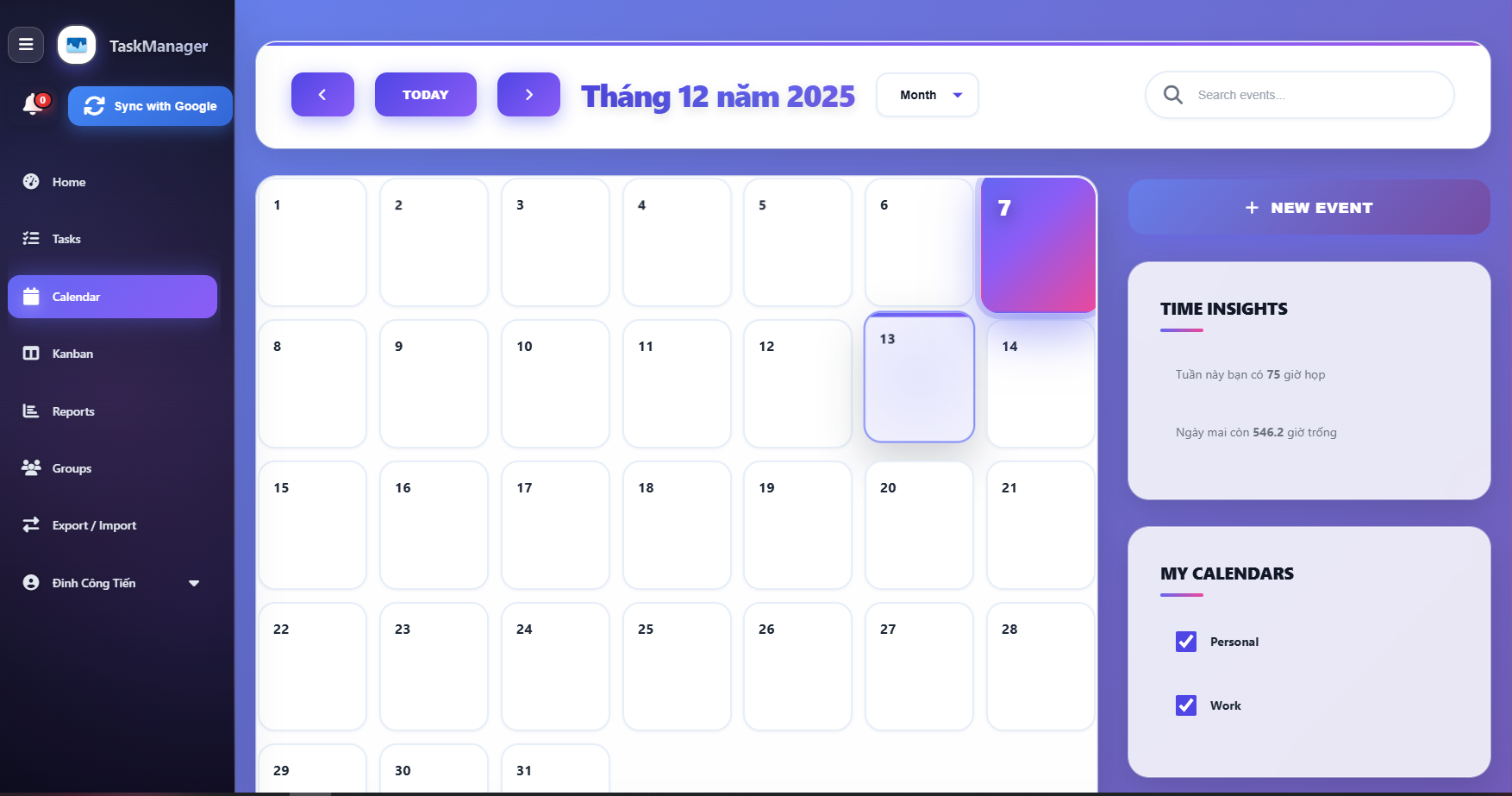
Giao diện Tasks tập trung vào việc quản lý danh sách công việc cá nhân. Người dùng có thể thêm mới nhiệm vụ bằng cách nhập tiêu đề, mô tả, hạn chót, và mức độ ưu tiên (thấp, trung bình, cao). Danh sách nhiệm vụ được hiển thị dưới dạng bảng hoặc lưới, với các bộ lọc theo trạng thái (To Do, In Progress, Done) và tìm kiếm theo từ khóa. Chức năng kéo-thả (drag-and-drop) được tích hợp để thay đổi trạng thái nhanh chóng. Trang này kết nối trực tiếp với backend qua API để lưu trữ và đồng bộ dữ liệu thời gian thực.



Hình 25. Giao diện Tasks

## 3.3. Giao diện Calendar

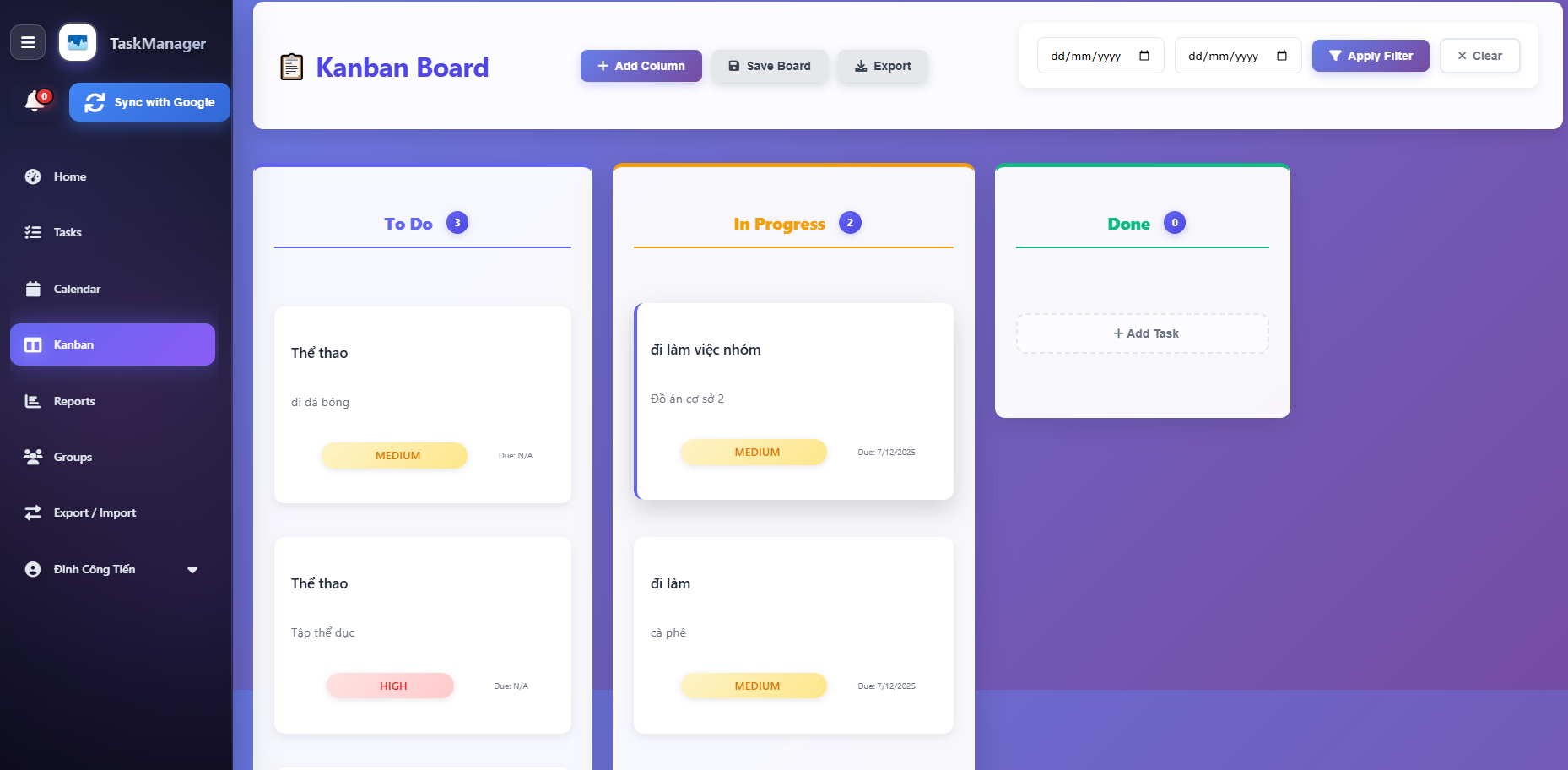
Giao diện Calendar cung cấp chế độ xem lịch trình theo tháng, tuần hoặc ngày, giúp người dùng lập kế hoạch sự kiện một cách trực quan. Người dùng có thể tạo sự kiện mới bằng cách chọn ngày giờ, thêm mô tả, địa điểm, và màu sắc đại diện. Hệ thống tự động kiểm tra xung đột thời gian và hiển thị nhắc nhở. Các sự kiện được đồng bộ với module Tasks để liên kết công việc với lịch trình. Giao diện sử dụng thư viện JavaScript để render lịch động, đảm bảo tính responsive trên thiết bị di động.



Hình 26. Giao diện Calendar

## 3.4. Giao diện Kanban

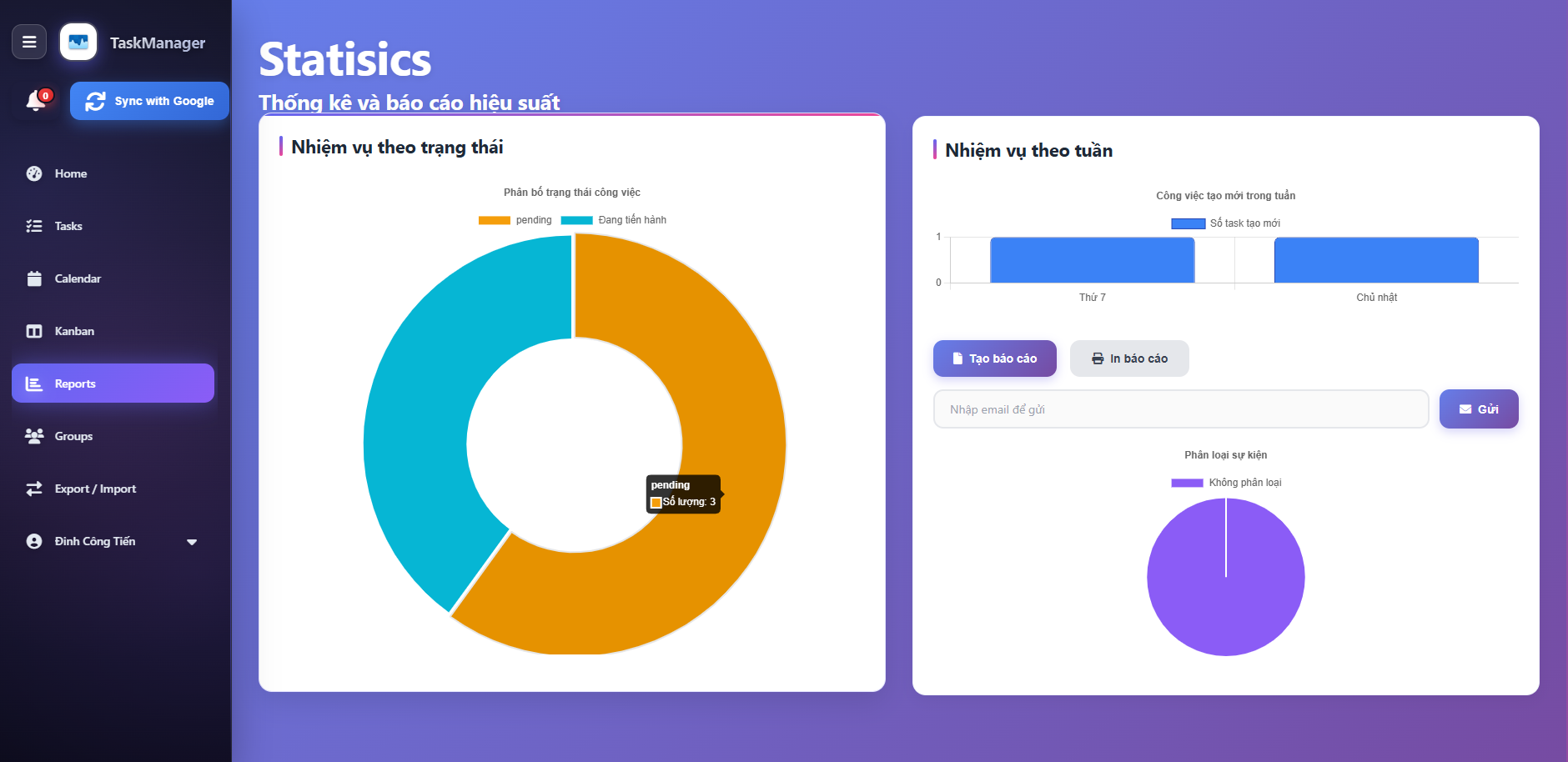
Giao diện Kanban là điểm nổi bật của hệ thống, hiển thị các nhiệm vụ dưới dạng bảng kéo-thả theo phong cách Trello. Các cột đại diện cho trạng thái (To Do, In Progress, Done), và người dùng có thể di chuyển thẻ nhiệm vụ giữa các cột để cập nhật tiến độ. Mỗi thẻ hiển thị thông tin ngắn gọn như tiêu đề, hạn chót, và người phụ trách. Chức năng này sử dụng Socket.io để cập nhật thời gian thực nếu có thay đổi từ người dùng khác trong nhóm. Giao diện giúp tăng tính trực quan và hiệu quả quản lý công việc.



Hình 27. Giao diện Kanban

## 3.5. Giao diện Reports

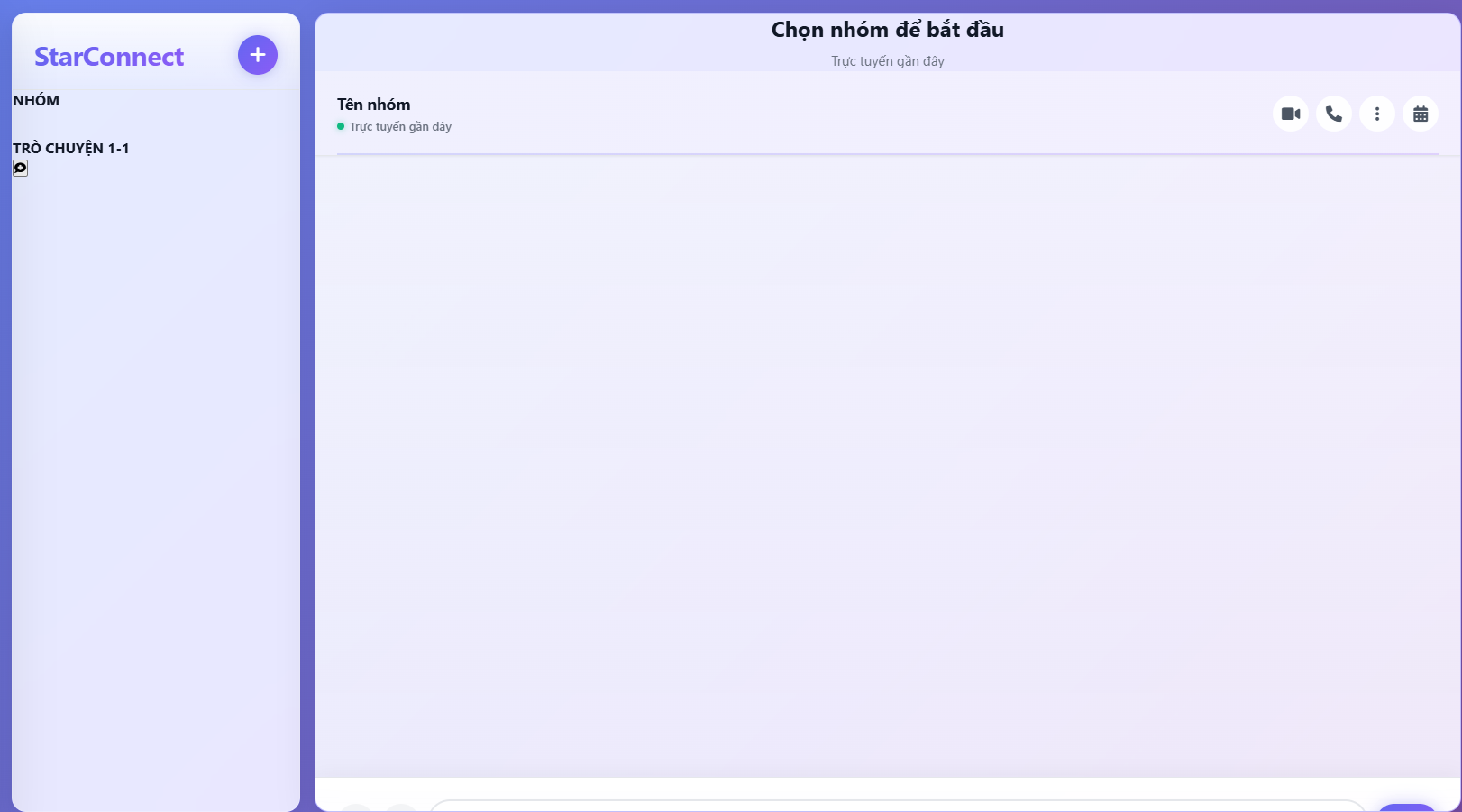
Giao diện Reports cung cấp các báo cáo thống kê về hiệu suất làm việc, bao gồm biểu đồ cột cho tỷ lệ hoàn thành nhiệm vụ, biểu đồ tròn cho phân bổ thời gian theo loại sự kiện, và danh sách công việc trễ hạn. Người dùng có thể chọn khoảng thời gian (tuần, tháng, quý) để lọc dữ liệu. Dữ liệu được lấy từ cơ sở dữ liệu PostgreSQL và render bằng thư viện Chart.js. Trang này giúp người dùng đánh giá và cải thiện năng suất cá nhân.



Hình 28. Giao diện Reports

## 3.6. Giao diện Groups

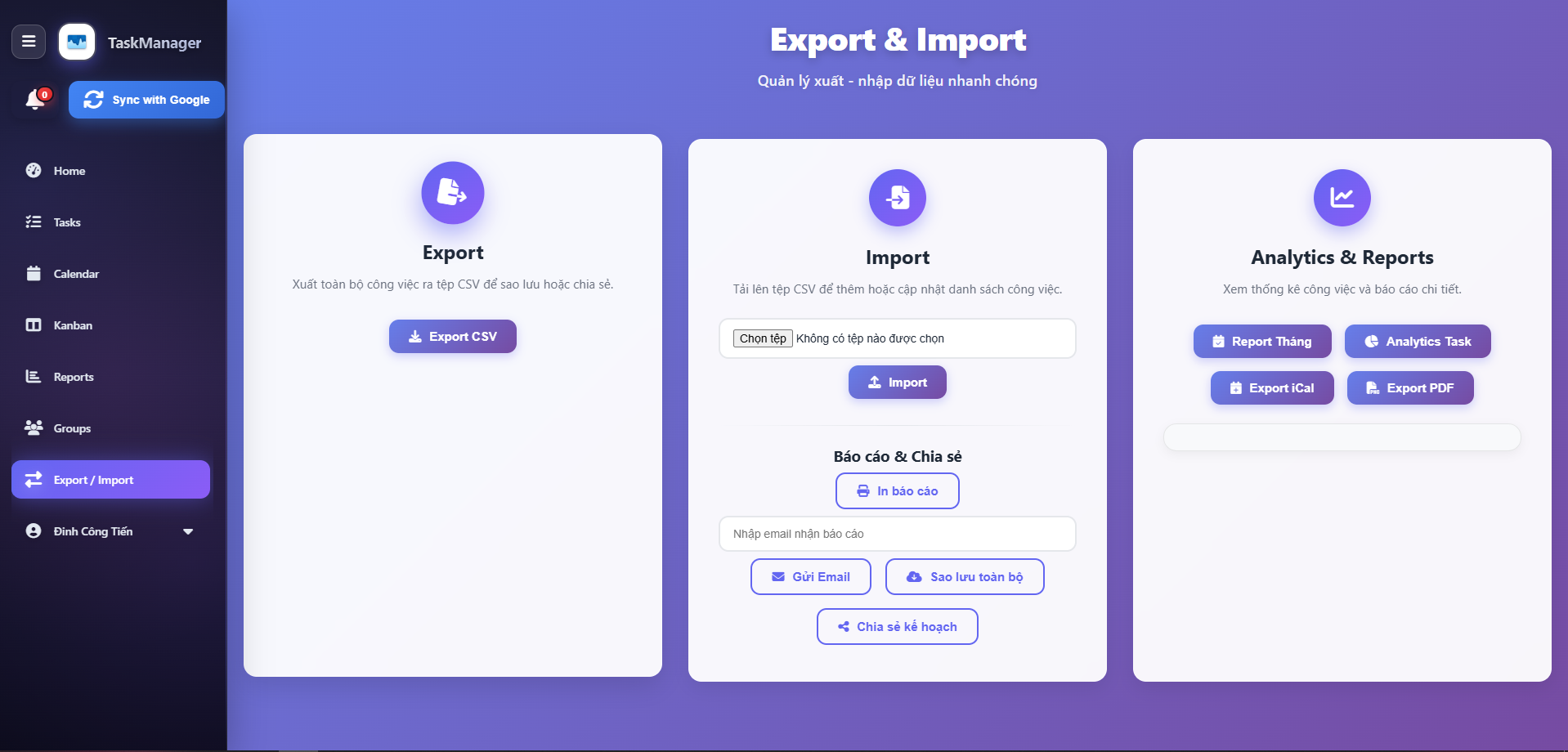
Giao diện Groups cho phép người dùng tạo và tham gia các nhóm làm việc chung, biến ứng dụng từ công cụ cá nhân thành nền tảng cộng tác nhóm đơn giản.



Hình 29. Giao diện Groups

## 3.7. Giao diện Export/Import

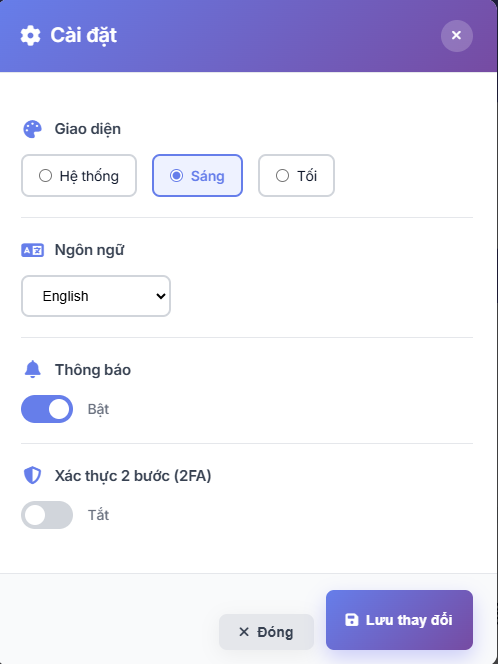
Giao diện Export/Import cho phép người dùng xuất dữ liệu nhiệm vụ hoặc lịch trình ra file CSV/Excel để sao lưu hoặc chia sẻ, và nhập dữ liệu từ file tương tự để đồng bộ nhanh chóng. Chức năng này sử dụng thư viện Multer cho upload file và xử lý dữ liệu qua backend. Giao diện đơn giản với nút chọn file và xác nhận, đảm bảo tính bảo mật bằng cách kiểm tra định dạng file trước khi xử lý.



Hình 30. Giao diện Export/Import

## 3.8. Giao diện Cài đặt

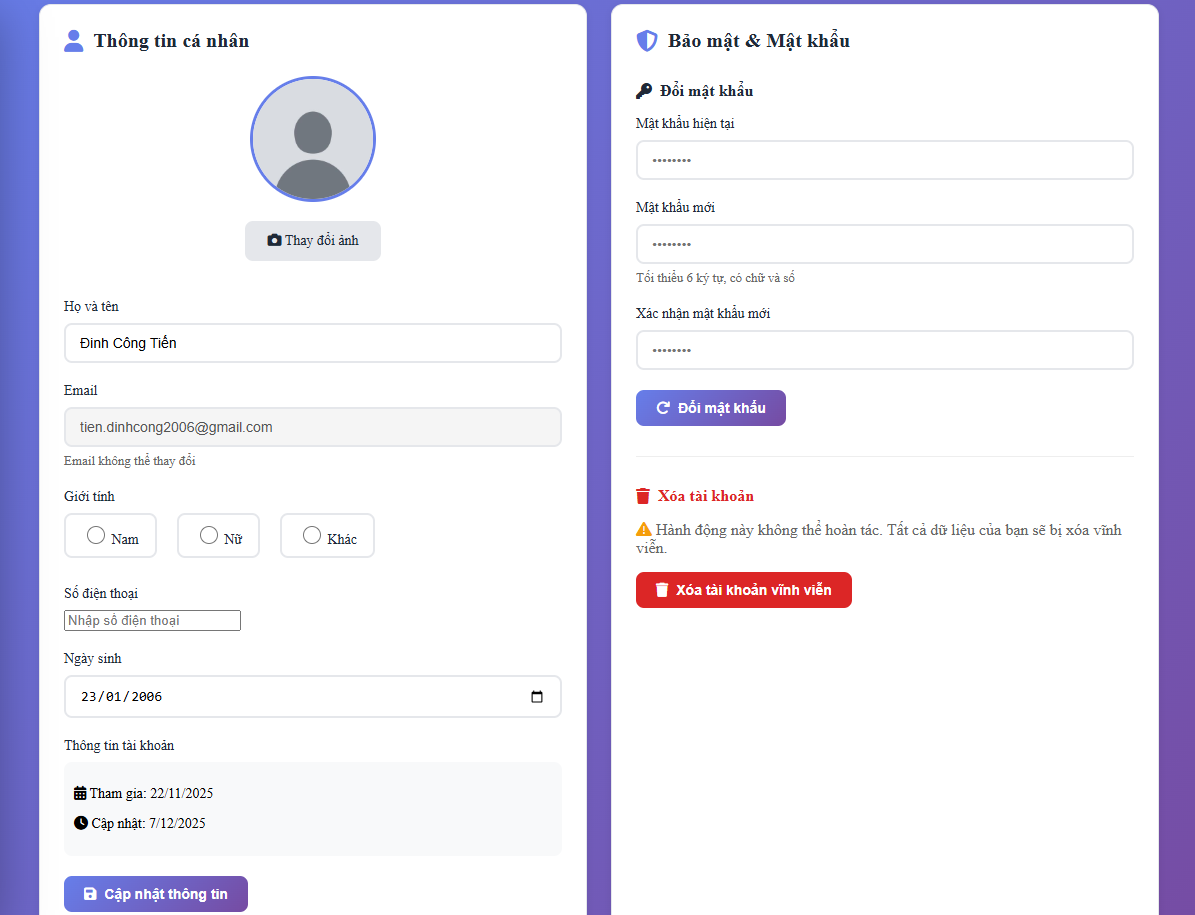
Giao diện Cài đặt cho phép người dùng tùy chỉnh tài khoản, bao gồm kích hoạt bảo mật 2 lớp (2FA), thay đổi ngôn ngữ, chủ đề giao diện, và thiết lập thông báo email. Trang này cũng hỗ trợ tích hợp đăng nhập bên thứ ba như Google OAuth. Các thay đổi được lưu trữ an toàn trong cơ sở dữ liệu và áp dụng ngay lập tức.



Hình 31. Giao diện Cài đặt

## 3.9. Giao diện Profile

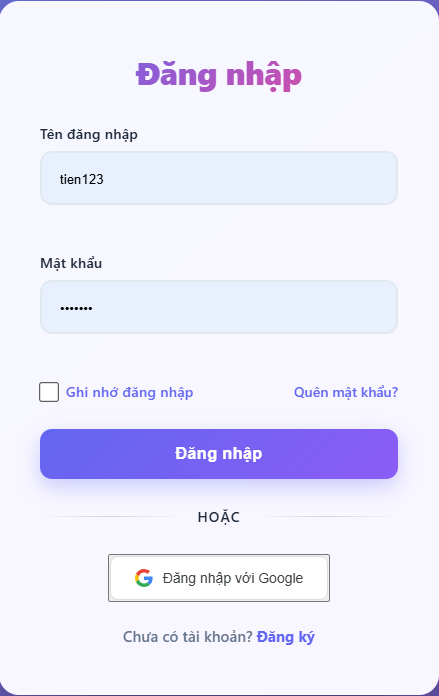
Giao diện Profile hiển thị thông tin cá nhân của người dùng như tên, email, ảnh đại diện, và lịch sử hoạt động gần đây. Người dùng có thể chỉnh sửa thông tin cơ bản và tải lên ảnh mới. Trang này được bảo vệ bằng session management để chỉ chủ tài khoản mới truy cập được.



Hình 32. Giao diện Profile

## 3.10. Giao diện Đăng nhập

Giao diện Đăng nhập là trang đầu tiên mà người dùng tiếp cận, với form nhập email/mật khẩu và tùy chọn đăng nhập qua Google. Thiết kế đơn giản, tập trung vào tính bảo mật với mã hóa mật khẩu bằng Bcrypt. Nếu đăng nhập thất bại, hệ thống hiển thị thông báo lỗi cụ thể.



Hình 33. Giao diện Đăng nhập

## 3.11. Giao diện Đăng ký

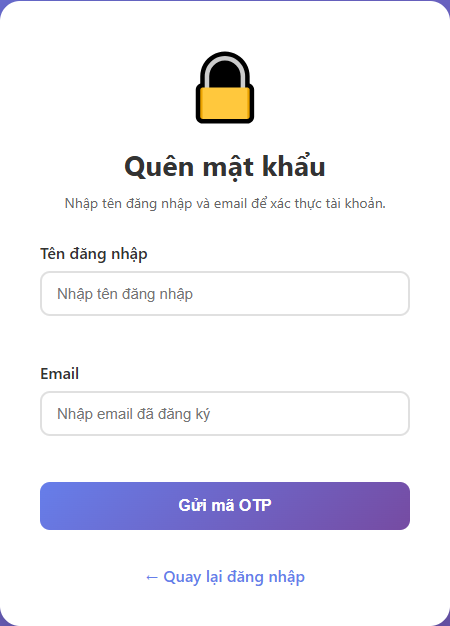
Giao diện Đăng ký yêu cầu người dùng nhập thông tin cơ bản như email, mật khẩu, và xác nhận mật khẩu. Sau khi submit, hệ thống gửi mã OTP qua email để xác thực. Giao diện sử dụng Nodemailer để gửi email và kiểm tra tính hợp lệ của dữ liệu đầu vào.



Hình 34. Giao diện Đăng ký

## 3.12. Giao diện Quên mật khẩu

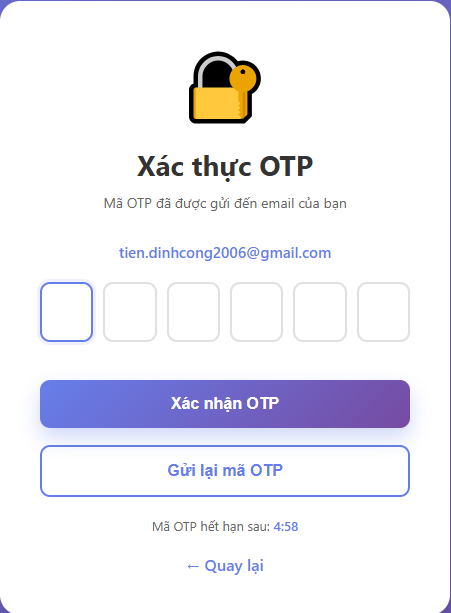
Giao diện Quên mật khẩu cho phép người dùng nhập email để nhận liên kết hoặc mã OTP khôi phục. Quy trình bao gồm gửi email xác thực và hướng dẫn đặt mật khẩu mới. Chức năng này giúp tăng tính tiện lợi và bảo mật cho người dùng.

****

Hình 35. Giao diện Quên mật khẩu

## 3.13. Giao diện Xác thực OTP

Giao diện Xác thực OTP hiển thị form nhập mã 6 chữ số được gửi qua email, dùng cho đăng ký, quên mật khẩu hoặc 2FA. Hệ thống giới hạn thời gian mã OTP (thường 5 phút) và cho phép gửi lại nếu cần. Giao diện đơn giản để tránh nhầm lẫn.

****

Hình 36. Giao diện Xác thực OTP

# CHƯƠNG 4. KẾT LUẬN

## 4.1 Kết luận

Sau quá trình nghiên cứu, phân tích và hiện thực hóa website “quản lý lịch trình cá nhân” đã hoàn thành các mục tiêu đề ra ban đầu. Về mặt sản phẩm, hệ thống đã cung cấp một giải pháp quản lý "tất cả trong một" hoạt động ổn định trên môi trường web. Các chức năng cốt lõi như quản lý nhiệm vụ với bảng Kanban trực quan, sắp xếp lịch trình trên Calendar, và hệ thống Chat nhóm đã được triển khai thành công, giúp người dùng giải quyết bài toán phân mảnh dữ liệu khi làm việc.

Về mặt kỹ thuật, đồ án đã xây dựng được một kiến trúc phần mềm vững chắc dựa trên nền tảng Node.js và Express Framework. Việc áp dụng mô hình phân lớp (Controller - Service - Model) giúp mã nguồn rõ ràng, dễ bảo trì và mở rộng. Hệ thống cơ sở dữ liệu PostgreSQL được thiết kế chuẩn hóa, đảm bảo tính toàn vẹn dữ liệu với các ràng buộc chặt chẽ. Đặc biệt, việc tích hợp thành công các dịch vụ bên thứ ba như Google OAuth để đăng nhập nhanh và Gmail SMTP để gửi mã xác thực OTP đã nâng cao đáng kể trải nghiệm và tính bảo mật cho người dùng.

Qua quá trình thực hiện đồ án, bản thân tôi đã tích lũy được nhiều kiến thức và kinh nghiệm quý báu. Tôi đã hiểu sâu hơn về quy trình phát triển phần mềm từ khâu lên ý tưởng, thiết kế cơ sở dữ liệu đến lập trình và kiểm thử. Kỹ năng lập trình Full-stack được cải thiện rõ rệt, đặc biệt là khả năng xử lý bất đồng bộ trong JavaScript và tư duy thiết kế giao diện người dùng hiện đại. Đồng thời, tôi cũng học được cách giải quyết các vấn đề phát sinh trong thực tế như xử lý lỗi, tối ưu hóa hiệu năng và bảo mật thông tin người dùng.

## 4.2 Hướng phát triển

Mặc dù đã đạt được những kết quả nhất định, hệ thống vẫn còn một số hạn chế cần được khắc phục và nâng cấp trong tương lai để trở nên hoàn thiện hơn.

Trước hết, về tính năng thời gian thực, hệ thống hiện tại vẫn còn phụ thuộc vào cơ chế Polling (gọi API định kỳ) cho một số tác vụ cập nhật. Hướng phát triển tiếp theo là chuyển đổi toàn bộ các tính năng như Chat, Thông báo và Cập nhật trạng thái Kanban sang sử dụng công nghệ WebSocket (thông qua thư viện Socket.io). Điều này sẽ giúp giảm tải cho máy chủ, tăng tốc độ phản hồi và mang lại trải nghiệm mượt mà hơn cho người dùng khi làm việc nhóm.

Thứ hai, ứng dụng có thể được tích hợp thêm trí tuệ nhân tạo (AI) để hỗ trợ người dùng tốt hơn. Cụ thể, hệ thống có thể phân tích thói quen làm việc của người dùng để gợi ý lịch trình tối ưu, tự động sắp xếp thứ tự ưu tiên công việc hoặc tóm tắt nội dung các cuộc thảo luận nhóm. Việc này sẽ biến ứng dụng từ một công cụ ghi chép thụ động thành một trợ lý ảo thông minh thực sự.

Cuối cùng, để đáp ứng nhu cầu truy cập mọi lúc mọi nơi, việc phát triển phiên bản ứng dụng di động (Mobile App) là một hướng đi cần thiết. Sử dụng các công nghệ đa nền tảng như React Native hoặc Flutter sẽ cho phép tận dụng lại hệ thống API Backend hiện có, giúp tiết kiệm thời gian phát triển mà vẫn đảm bảo tính đồng bộ dữ liệu giữa các nền tảng Web và Mobile.

# TÀI LIỆU THAM KHẢO

1. Google Calendar - <https://calendar.google.com/calendar>
2. Node.js Official Documentation – <https://nodejs.org/docs/latest/api/>
3. PostgreSQL Documentation – <https://www.postgresql.org/docs/>
4. Bcrypt – Password hashing for Node.js – <https://www.npmjs.com/package/bcrypt>
5. Chart.js – Simple yet flexible JavaScript charting – <https://www.chartjs.org>
6. W3Schools & GeeksforGeeks – <https://www.w3schools.com/>
7. GitHub Repository của đồ án - <https://github.com/>