

BỘ GIÁO DỤC VÀ ĐÀO TẠO  
ĐẠI HỌC PHENIKAA

oOo



School Management App

BÁO CÁO KỸ THUẬT PHẦN MỀM

GVHD: Trương Anh Hoàng

HÀ NỘI - 2025

Thông tin nhóm thực hiện

Mã sinh viên	Họ và tên	Chức vụ
22010124	Lê Thành Long	Nhóm trưởng
22010493	Hồ Xuân Hùng	Thành viên
22010044	Dương Thanh Bình	Thành viên

# Mục lục

Danh mục các hình ảnh	i
<b>1 Giới thiệu</b>	<b>1</b>
1.1 Mục đích . . . . .	1
1.2 Phạm vi . . . . .	1
1.3 Từ điển thuật ngữ . . . . .	1
1.4 Tài liệu tham khảo . . . . .	2
1.5 Tổng quát . . . . .	2
<b>2 Các yêu cầu chức năng</b>	<b>3</b>
2.1 Các tác nhân . . . . .	3
2.2 Các chức năng của hệ thống . . . . .	3
2.3 Biểu đồ use-case tổng quát . . . . .	4
2.4 Biểu đồ use-case phân rã . . . . .	5
2.5 Quy trình nghiệp vụ . . . . .	5
2.6 Đặc tả use-case . . . . .	6
<b>3 Các yêu cầu phi chức năng</b>	<b>7</b>
3.1 Các yêu cầu về hiệu năng . . . . .	7
3.2 Yêu cầu về bảo mật . . . . .	8
3.3 Yêu cầu về giao diện . . . . .	8
3.4 Ràng buộc . . . . .	8

## Danh mục các hình ảnh

1 Biểu đồ use-case tổng quát của hệ thống . . . . .	5
---	---

# 1. Giới thiệu

## 1.1 Mục đích

Báo cáo này được soạn thảo nhằm trình bày chi tiết về hệ thống quản lý trường học, một ứng dụng web hiện đại được phát triển bằng Next.js, React, Prisma và PostgreSQL. Mục tiêu chính là cung cấp một giải pháp toàn diện để quản lý các hoạt động trong trường học, bao gồm thông tin giáo viên, học sinh, phụ huynh, lớp học, môn học, kỳ thi, bài tập, sự kiện và thông báo. Hệ thống hướng đến việc cải thiện hiệu quả quản lý, tăng cường trải nghiệm người dùng thông qua giao diện đáp ứng, và đảm bảo bảo mật cũng như hiệu năng cao. Báo cáo này mô tả các yêu cầu chức năng, phi chức năng, quy trình nghiệp vụ, và các thành phần kỹ thuật của hệ thống, đồng thời cung cấp cái nhìn tổng quan về cách hệ thống đáp ứng nhu cầu của các bên liên quan.

## 1.2 Phạm vi

Phạm vi của hệ thống bao gồm:

- **Quản lý dữ liệu:** Quản lý thông tin chi tiết về giáo viên, học sinh, phụ huynh, lớp học, môn học, kỳ thi, bài tập, sự kiện và thông báo, với khả năng thêm, sửa, xóa thông qua các biểu mẫu động.
- **Giao diện người dùng:** Cung cấp các bảng điều khiển (dashboard) riêng biệt cho quản trị viên, giáo viên, học sinh và phụ huynh, với thiết kế đáp ứng trên các thiết bị từ điện thoại di động (320px) đến máy tính để bàn (1920px).
- **Tích hợp cơ sở dữ liệu:** Sử dụng Prisma để tương tác với cơ sở dữ liệu PostgreSQL, hỗ trợ truy vấn lồng, phân trang và lọc dữ liệu.
- **Xác thực và phân quyền:** Tích hợp Clerk để quản lý đăng nhập, phân quyền theo vai trò, và bảo vệ các tuyến đường (routes).
- **Biểu mẫu động:** Sử dụng React-Hook-Form và Zod để tạo biểu mẫu với xác thực dữ liệu, hỗ trợ select options động từ cơ sở dữ liệu.
- **Biểu đồ và lịch:** Hiển thị dữ liệu thống kê qua biểu đồ động (biểu đồ cột, đường, tròn) sử dụng Recharts và lịch sự kiện với Big Calendar, hỗ trợ lịch định kỳ.
- **Triển khai:** Hỗ trợ triển khai ứng dụng bằng Docker với cơ sở dữ liệu PostgreSQL, đảm bảo tính di động và dễ dàng mở rộng.

Báo cáo không bao gồm các chi tiết về triển khai phần cứng, chi phí vận hành, hoặc các tính năng ngoài ứng dụng web (ví dụ: ứng dụng di động riêng).

## 1.3 Từ điển thuật ngữ

- **Next.js:** Framework React hỗ trợ server-side rendering (SSR), static site generation (SSG), API routes và lazy loading để tối ưu hóa hiệu suất.
- **Prisma:** Công cụ ORM (Object-Relational Mapping) giúp đơn giản hóa việc tương tác với cơ sở dữ liệu quan hệ như PostgreSQL.

- **React-Hook-Form:** Thư viện quản lý biểu mẫu trong React, tối ưu hóa hiệu suất và hỗ trợ xác thực dữ liệu.
- **Zod:** Thư viện xác thực dữ liệu cho JavaScript/TypeScript, thường được sử dụng với React-Hook-Form.
- **Clerk:** Dịch vụ xác thực và quản lý vai trò người dùng, hỗ trợ đăng nhập an toàn và phân quyền.
- **Recharts:** Thư viện tạo biểu đồ động (cột, đường, tròn) trong React.
- **Big Calendar:** Thư viện React để hiển thị và quản lý lịch sự kiện, hỗ trợ lịch định kỳ.
- **Server Actions:** Tính năng của Next.js cho phép thực thi các hàm phía server từ giao diện người dùng.
- **React Toastify:** Thư viện hiển thị thông báo (notifications) trong ứng dụng React.
- **Cloudinary:** Dịch vụ lưu trữ và quản lý hình ảnh, được tích hợp để tải lên ảnh hồ sơ.

## 1.4 Tài liệu tham khảo

### Tài liệu tham khảo

- [1] Tài liệu chính thức của Next.js: <https://nextjs.org/docs>
- [2] Tài liệu Prisma: <https://www.prisma.io/docs>
- [3] Tài liệu React-Hook-Form: <https://react-hook-form.com>
- [4] Tài liệu Clerk: <https://clerk.dev/docs>
- [5] Tài liệu Recharts: <https://recharts.org/en-US>
- [6] Tài liệu Big Calendar: <https://github.com/jquense/react-big-calendar>
- [7] Tài liệu React Toastify: <https://fkhadra.github.io/react-toastify>

## 1.5 Tổng quát

Hệ thống quản lý trường học là một ứng dụng web toàn diện, được xây dựng trên nền tảng Next.js, tích hợp các công nghệ hiện đại như React, Prisma, PostgreSQL và Clerk. Ứng dụng cung cấp các bảng điều khiển được thiết kế riêng cho từng vai trò người dùng:

- **Admin Dashboard:** Hiển thị tổng quan dữ liệu trường học (số lượng học sinh, giáo viên, lớp học), biểu đồ thống kê (số lượng học sinh, điểm thi, tỷ lệ chuyên cần), và các công cụ quản lý dữ liệu (thêm, sửa, xóa).
- **Teacher Dashboard:** Cho phép giáo viên xem lịch trình giảng dạy, quản lý bài tập, nhập điểm thi, và xem thông tin lớp học.

- **Student Dashboard:** Cung cấp thông tin cá nhân, lịch học, bài tập, kết quả thi và thông báo từ trường.
- **Parent Dashboard:** Hỗ trợ phụ huynh theo dõi thông tin học sinh, lịch học, thông báo và sự kiện.

Hệ thống sử dụng Tailwind CSS để đảm bảo giao diện đáp ứng và trực quan, tích hợp biểu đồ động với Recharts và lịch sự kiện với Big Calendar. Prisma quản lý cơ sở dữ liệu PostgreSQL, hỗ trợ truy vấn lồng và phân trang. Clerk đảm bảo xác thực và phân quyền an toàn, trong khi React-Hook-Form và Zod cung cấp biểu mẫu động với xác thực mạnh mẽ. Ứng dụng được triển khai bằng Docker để đảm bảo tính di động và dễ dàng mở rộng.

## 2. Các yêu cầu chức năng

### 2.1 Các tác nhân

Hệ thống hỗ trợ bốn tác nhân chính:

- **Quản trị viên (Admin):** Có toàn quyền quản lý hệ thống, bao gồm thêm, sửa, xóa thông tin giáo viên, học sinh, phụ huynh, lớp học, môn học, kỳ thi, bài tập, sự kiện và thông báo. Quản trị viên cũng xem báo cáo và thống kê tổng quan.
- **Giáo viên:** Quản lý thông tin lớp học, nhập điểm thi, tạo bài tập, và xem lịch trình giảng dạy. Chỉ truy cập dữ liệu liên quan đến lớp học hoặc học sinh họ phụ trách.
- **Học sinh:** Xem thông tin cá nhân, lịch học, bài tập, kết quả thi và thông báo từ trường.
- **Phụ huynh:** Theo dõi thông tin học sinh (con cái), xem lịch học, thông báo và sự kiện.

### 2.2 Các chức năng của hệ thống

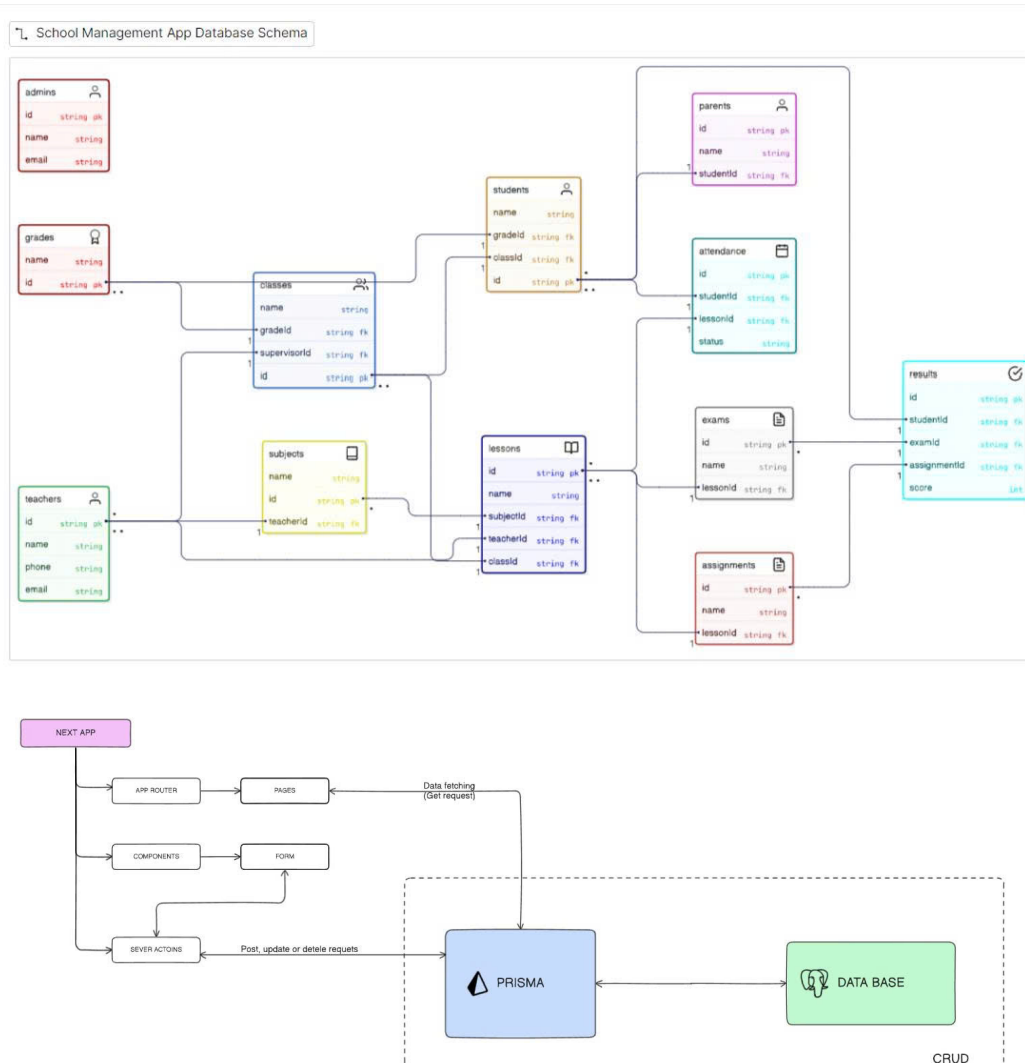
Hệ thống cung cấp các chức năng chính:

- **Quản lý dữ liệu:**
  - Thêm, sửa, xóa thông tin giáo viên, học sinh, phụ huynh, lớp học, môn học, kỳ thi, bài tập, sự kiện và thông báo.
  - Tải lên hình ảnh hồ sơ (giáo viên, học sinh) thông qua Cloudinary.
- **Hiển thị dữ liệu:**
  - Biểu đồ động (biểu đồ cột, đường, tròn) với Recharts để hiển thị thống kê (số lượng học sinh, điểm thi, tỷ lệ chuyên cần).
  - Lịch sự kiện với Big Calendar, hỗ trợ lịch định kỳ.
  - Danh sách dữ liệu (giáo viên, học sinh, lớp học) với phân trang và bộ lọc tìm kiếm.

- **Biểu mẫu động:**
  - Tạo, cập nhật, xóa dữ liệu qua biểu mẫu sử dụng React-Hook-Form và Zod.
  - Hỗ trợ hiển thị có điều kiện (conditional rendering) và select options động từ cơ sở dữ liệu.
- **Phân quyền và xác thực:**
  - Sử dụng Clerk để quản lý đăng nhập và phân quyền theo vai trò.
  - Bảo vệ tuyến đường và dữ liệu dựa trên vai trò người dùng.
- **Thông báo:**
  - Gửi thông báo qua React Toastify khi có sự kiện mới hoặc hành động quan trọng.
- **Tích hợp cơ sở dữ liệu:**
  - Prisma thực hiện truy vấn lồng, phân trang và lọc dữ liệu.
  - Server Actions xử lý các thao tác tạo, cập nhật, xóa dữ liệu.
- **Tối ưu hóa hiệu suất:**
  - Lazy Loading với ‘dynamic()’ của Next.js để giảm thời gian tải trang.
  - Phân trang và bộ lọc tìm kiếm để xử lý dữ liệu lớn.

## 2.3 Biểu đồ use-case tổng quát

**Mô tả:** Biểu đồ use-case tổng quát mô tả các tác nhân (quản trị viên, giáo viên, học sinh, phụ huynh) tương tác với hệ thống thông qua các chức năng như đăng nhập, quản lý dữ liệu, xem biểu đồ, xem lịch và nhận thông báo. Mỗi tác nhân được liên kết với các chức năng cụ thể dựa trên vai trò. **Mô tả:** Biểu đồ use-case tổng quát mô tả các tác nhân (quản trị viên, giáo viên, học sinh, phụ huynh) tương tác với hệ thống thông qua các chức năng như đăng nhập, quản lý dữ liệu, xem biểu đồ, xem lịch và nhận thông báo. Mỗi tác nhân được liên kết với các chức năng cụ thể dựa trên vai trò.



Hình 1: Biểu đồ use-case tổng quát của hệ thống

## 2.4 Biểu đồ use-case phân rã

Mô tả: Biểu đồ use-case phân rã chi tiết hóa các chức năng:

- Quản trị viên: Thêm/sửa/xóa giáo viên, học sinh, lớp học, thông báo.
- Giáo viên: Quản lý bài tập, nhập điểm thi, xem lịch giảng dạy.
- Học sinh: Xem lịch học, bài tập, kết quả thi.
- Phụ huynh: Xem thông tin học sinh, thông báo, sự kiện.

## 2.5 Quy trình nghiệp vụ

1. **Đăng nhập:** Người dùng đăng nhập qua Clerk, được phân quyền theo vai trò.
2. **Quản lý dữ liệu:** Quản trị viên truy cập bảng điều khiển admin để thêm, sửa, xóa dữ liệu thông qua biểu mẫu động.

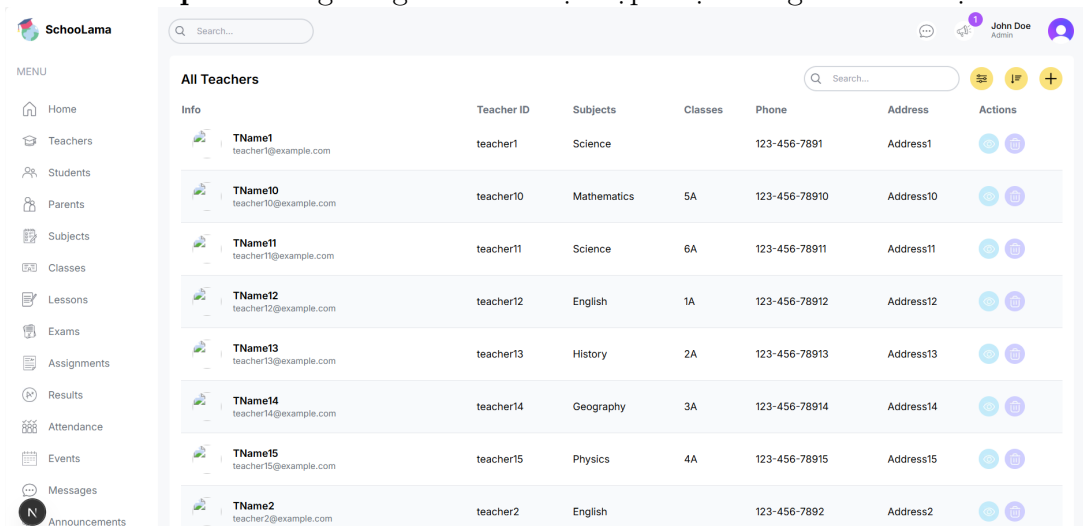


3. **Xem và quản lý lịch trình:** Giáo viên và học sinh xem lịch sự kiện qua Big Calendar. Giáo viên có thể thêm lịch định kỳ.
4. **Quản lý bài tập và điểm thi:** Giáo viên tạo bài tập, nhập điểm thi. Học sinh và phụ huynh xem kết quả.
5. **Thông báo:** Hệ thống gửi thông báo qua React Toastify khi có sự kiện hoặc thay đổi dữ liệu.
6. **thống kê:** Quản trị viên và giáo viên xem biểu đồ thống kê trên bảng điều khiển.

## 2.6 Đặc tả use-case

### Use-case 1: Quản lý giáo viên

- **Tác nhân:** Quản trị viên
- **Mô tả:** Quản trị viên thêm, sửa, xóa thông tin giáo viên.
- **Điều kiện tiên quyết:** Quản trị viên đã đăng nhập và có quyền admin qua Clerk.
- **Luồng chính:**
  1. Truy cập trang quản lý giáo viên từ bảng điều khiển admin.
  2. Chọn hành động: Thêm, sửa, xóa giáo viên.
  3. Nếu thêm/sửa: Nhập thông tin (họ tên, email, môn học, ảnh hồ sơ) vào biểu mẫu React-Hook-Form với xác thực Zod.
  4. Nếu xóa: Xác nhận qua modal.
  5. Gửi yêu cầu đến Server Actions để thực thi trên cơ sở dữ liệu PostgreSQL qua Prisma.
  6. Hiển thị thông báo thành công/lỗi qua React Toastify.
- **Luồng phụ:**
  - Nếu dữ liệu đầu vào không hợp lệ, hiển thị lỗi xác thực.
  - Nếu giáo viên liên kết với lớp học, yêu cầu xác nhận trước khi xóa.
- **Kết quả:** Thông tin giáo viên được cập nhật trong cơ sở dữ liệu.



## Giao diện trực quan

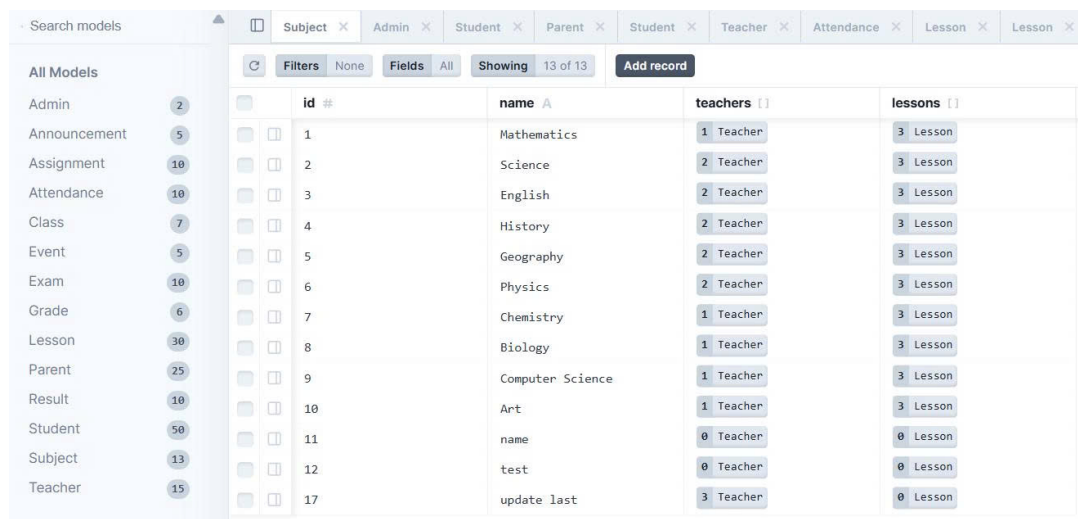
### Use-case 2: Xem lịch học

- **Tác nhân:** Học sinh
- **Mô tả:** Học sinh xem lịch học cá nhân.
- **Điều kiện tiên quyết:** Học sinh đã đăng nhập.
- **Luồng chính:**
  1. Truy cập bảng điều khiển học sinh.
  2. Chọn mục lịch học, hiển thị qua Big Calendar.
  3. Hệ thống dùng Prisma truy vấn dữ liệu lịch học liên quan đến học sinh.
  4. Lịch học hiển thị với sự kiện được đánh dấu theo ngày/giờ.
- **Luồng phụ:** Nếu không có lịch học, hiển thị thông báo "Chưa có lịch học".
- **Kết quả:** Học sinh thấy lịch học cá nhân với giao diện trực quan.

## 3. Các yêu cầu phi chức năng

### 3.1 Các yêu cầu về hiệu năng

- **Thời gian phản hồi:** Yêu cầu từ giao diện hoàn thành trong 2 giây dưới tải bình thường.
- **Khả năng chịu tải:** Hỗ trợ ít nhất 100 người dùng đồng thời mà không giảm hiệu suất.
- **Tối ưu hóa tải trang:** Sử dụng Lazy Loading với 'dynamic()' để giảm thời gian tải trang dưới 1 giây trên mạng 4G.
- **Phân trang:** Danh sách dữ liệu lớn phân trang với tối đa 20 mục mỗi trang.



The screenshot shows a web application interface with a table of records. The table has columns for 'id #', 'name A', 'teachers []', and 'lessons []'. The 'name A' column contains subjects like Mathematics, Science, English, History, Geography, Physics, Chemistry, Biology, Computer Science, Art, name, test, and update last. The 'teachers []' column shows the number of teachers (1 or 2) and the role 'Teacher'. The 'lessons []' column shows the number of lessons (3 or 0) and the role 'Lesson'. The interface includes a search bar, filters, fields, and pagination controls.

id #	name A	teachers []	lessons []
1	Mathematics	1 Teacher	3 Lesson
2	Science	2 Teacher	3 Lesson
3	English	2 Teacher	3 Lesson
4	History	2 Teacher	3 Lesson
5	Geography	2 Teacher	3 Lesson
6	Physics	2 Teacher	3 Lesson
7	Chemistry	1 Teacher	3 Lesson
8	Biology	1 Teacher	3 Lesson
9	Computer Science	1 Teacher	3 Lesson
10	Art	1 Teacher	3 Lesson
11	name	0 Teacher	0 Lesson
12	test	0 Teacher	0 Lesson
17	update last	3 Teacher	0 Lesson

### 3.2 Yêu cầu về bảo mật

- **Xác thực:** Sử dụng Clerk với hỗ trợ MFA nếu cần.
- **Phân quyền:** Bảo vệ tuyến đường bằng Clerk, chỉ người dùng có vai trò phù hợp truy cập được.
- **Xác thực dữ liệu:** Sử dụng Zod để ngăn chặn tấn công SQL injection hoặc XSS.
- **Bảo vệ dữ liệu:** Mã hóa dữ liệu nhạy cảm trong cơ sở dữ liệu.

### 3.3 Yêu cầu về giao diện

- **Đáp ứng:** Hoạt động mượt mà trên thiết bị từ 320px đến 1920px.
- **Thẩm mỹ:** Sử dụng Tailwind CSS để thiết kế giao diện hiện đại, nhất quán.
- **Biểu đồ và lịch:** Biểu đồ (Recharts) và lịch (Big Calendar) hiển thị trực quan, dễ hiểu.
- **Hiệu ứng động:** Tích hợp hiệu ứng (chuyển đổi biểu đồ, modal) để tăng tương tác.

### 3.4 Ràng buộc

- **Tương thích:** Hoạt động trên Chrome, Firefox, Safari phiên bản mới nhất.
- **Cơ sở dữ liệu:** Sử dụng PostgreSQL, triển khai qua Docker.
- **Thư viện:** Chỉ dùng Next.js, Prisma, React-Hook-Form, Zod, Clerk, Recharts, Big Calendar, React Toastify, Cloudinary.
- **Triển khai:** Hỗ trợ triển khai trên Vercel hoặc máy chủ cục bộ với Docker.