

TRƯỜNG ĐẠI HỌC KIẾN TRÚC ĐÀ NẴNG
KHOA CÔNG NGHỆ THÔNG TIN



ĐỒ ÁN TỐT NGHIỆP

NGÀNH: CÔNG NGHỆ THÔNG TIN

ĐỀ TÀI:

NGUYÊN CỨU, XÂY DỰNG HỆ THỐNG QUẢN LÝ VÀ ĐẶT SÂN THỂ THAO TRỰC TUYẾN TẠI ĐÀ NẴNG

Người hướng dẫn: TS. Lâm Tùng Giang

Sinh viên thực hiện: TRẦN QUYẾT VIỆT

Mã số sinh viên: 2151220115

Lớp: 21CT2

Đà Nẵng, 11/2025

TRƯỜNG ĐẠI HỌC KIẾN TRÚC ĐÀ NẴNG
KHOA CÔNG NGHỆ THÔNG TIN



ĐỒ ÁN TỐT NGHIỆP

NGÀNH: CÔNG NGHỆ THÔNG TIN

ĐỀ TÀI:

**NGHIÊN CỨU, XÂY DỰNG HỆ THỐNG QUẢN LÝ VÀ ĐẶT SÂN THỂ
THAO TRỰC TUYẾN TẠI ĐÀ NẴNG**

Người hướng dẫn: TS. Lâm Tùng Giang

Sinh viên thực hiện: Trần Quyết Việt

Mã số sinh viên: 2151220115

Lớp: 21CT3

Đà Nẵng, 11/2025

TÓM TẮT

Tên đề tài: Nguyên cứu, xây dựng Hệ thống quản lý và đặt sân thể thao trực tuyến tại Đà Nẵng

Sinh viên thực hiện: Trần Quyết Việt

Mã số SV: 2151220115 - Lớp:

21CT3

Đề tài nhằm nghiên cứu công nghệ Django và NextJS, áp dụng vào việc phát triển hệ thống quản lý và đặt sân thể thao trực tuyến. Trang web này được thiết kế để giúp người dùng tìm kiếm, đặt sân thể thao, kết nối với các cơ sở thể thao và quản lý lịch đặt sân một cách hiệu quả.

LỜI NÓI ĐẦU

Để hoàn thành được Đồ án tốt nghiệp này em xin chân thành cảm ơn quý thầy cô trong khoa Công Nghệ Thông Tin - Trường Đại Học Kiến Trúc Đà Nẵng đã tận tình truyền đạt kiến thức trong những năm học vừa qua. Với vốn kiến thức được tiếp thu trong quá trình học không chỉ là nền tảng cho quá trình làm đồ án tốt nghiệp mà còn là hành trang quý báu để em bước vào đời một cách vững chắc và tự tin.

Em cũng xin gửi lời cảm ơn chân thành và sự tri ân sâu sắc của thầy Lâm Tùng Giang trong suốt thời gian vừa qua đã nhiệt tình chỉ dạy, giúp đỡ em để có thể hoàn thành tốt đồ án tốt nghiệp.

Trong quá trình làm bài đồ án tốt nghiệp, do trình độ lý luận cũng như kinh nghiệm thực tiễn còn hạn chế nên không thể tránh khỏi những thiếu sót, em rất mong nhận được ý kiến đóng góp của quý báu của các thầy cô để em có thể học hỏi thêm được nhiều kinh nghiệm, kiến thức.

Em xin trân trọng cảm ơn .

CAM ĐOAN

Em xin cam đoan kết quả đạt được trong đồ án tốt nghiệp là sản phẩm của riêng cá nhân dưới sự hướng dẫn của TS Lâm Tùng Giang, không sao chép lại của người khác. Trong toàn bộ nội dung của đồ án, những điều được trình bày hoặc là của bản thân hoặc là được tổng hợp từ nhiều nguồn tài liệu. Tất cả các tài liệu tham khảo đều có xuất xứ rõ ràng và được trích dẫn hợp pháp. Em xin hoàn toàn chịu trách nhiệm và chịu mọi hình thức kỷ luật theo quy định cho lời cam đoan của mình.

Sinh viên thực hiện

Viết

Trần Quyết Việt

MỤC LỤC

Contents

ĐỀ TÀI:	1
ĐỀ TÀI:	2
TÓM TẮT	5
LỜI NÓI ĐẦU	6
CAM ĐOAN	7
MỤC LỤC	8
MỤC CÁC BẢNG	11
DANH MỤC HÌNH ẢNH	12
DANH MỤC CÁC KÍ HIỆU, CHỮ VIẾT TẮT	13
Chương 1: GIỚI THIỆU TỔNG QUAN	1
1.1. Lý do chọn đề tài	1
1.2. Lịch sử nghiên cứu	1
1.3. Mục tiêu đề tài	2
Mục tiêu tổng quát:.....	2
Mục tiêu cụ thể:	2
1.4. Đối tượng, phạm vi nghiên cứu	3
Đối tượng nghiên cứu	3
Phạm vi nghiên cứu	3
Chương 2: CƠ SỞ LÝ THUYẾT	3
2.1. Giới thiệu về ngôn ngữ	3
2.1.1. Python	3
Những ưu điểm làm nên chất lượng của ngôn ngữ Python:	4
2.1.2. JavaScript	4
Những ưu điểm làm nên chất lượng của ngôn ngữ Javascript:	5
2.1.3. TypeScript	5
Tại sao nên sử dụng TypeScript:	6
2.2. Giới thiệu về Django REST Framework	6
2.2.1. Django REST Framework là gì?	6
2.2.2. Cấu trúc của Django REST Framework	7
2.2.3. Hướng dẫn cài đặt Django REST Framework	8
2.3. Giới thiệu về Framework NextJS	8
2.3.1. Framework NextJS là gì?	8
2.3.2. Lợi thế của NextJS	9

2.3.3. Nhược điểm của NextJS	10
2.4. Giới thiệu về cơ sở dữ liệu PostgreSQL	10
2.4.1. PostgreSQL là gì?	10
Đặc Điểm Nổi Bật:	10
2.4.2. Mối liên hệ và ràng buộc trong MySQL	11
Các kiểu quan hệ trong MySQL:	11
2.5. Giới thiệu về mô hình ngôn ngữ lớn Llama	12
2.5.1. Mô hình ngôn ngữ lớn Llama là gì?	12
2.5.2. Ứng dụng của mô hình Llama trong hệ thống	13
2.6. Công cụ phát triển Visual Studio Code	13
2.6.1. Visual Studio Code là gì	13
2.6.2. Tại sao nên sử dụng Visual Studio Code?	13
Chương 3: PHÂN TÍCH, THIẾT KẾ HỆ THỐNG	14
3.1. Mô tả bài toán	14
3.1.1. Yêu cầu người dùng thường (User)	14
3.1.2. Yêu cầu chủ sân (Owner)	15
3.1.3. Yêu cầu quản trị viên (Admin)	16
3.1.4. Chức năng Chatbot AI hỗ trợ người dùng	17
3.2. Phân tích hệ thống	17
3.2.1. Chức năng đăng nhập của người tìm việc	18
3.2.2. Chức năng đăng ký của người dùng	19
3.2.3. Chức năng tìm kiếm và xem thông tin sân	21
3.2.4. Chức năng đặt sân thể thao	22
3.2.5. Chức năng quản lý trung tâm của ADMIN	24
3.2.6. Chức năng quản lý sân của chủ sân	26
3.2.7. Chức năng theo dõi booking và thống kê của chủ sân	28
3.2.8. Chức năng hỗ trợ tìm kiếm thông tin qua chatbot	29
3.3. Thiết kế cơ sở dữ liệu	33
3.3.1. Mô hình dữ liệu quan hệ	33
3.3.2. Danh mục các bảng cơ sở dữ liệu	33
Thiết kế giao diện	36
Giao diện đăng nhập	36
3.3.3. Giao diện đăng ký	36
3.3.4. Giao diện trang chủ	37
3.3.5. Giao diện tìm kiếm sân	37
3.3.6. Giao diện chi tiết sân	38

3.3.7. Giao diện đặt sân	38
3.3.8. Giao diện xem lịch sử đặt sân	39
3.3.9. Giao diện quản lý người dùng	40
3.3.10. Giao diện quản trị tổng quan	40
3.3.11. Giao diện quản lý trung tâm (ADMIN)	41
3.3.12. Giao diện quản lý sân (OWNER)	41
3.3.13. Giao diện thống kê booking	42
3.3.14. Giao diện ChatBot AI	44
3.3.15. Một vài giao diện quản trị cho ADMIN	44
Chương 4: CÀI ĐẶT VÀ GIẢI QUYẾT CÁC VẤN ĐỀ VỀ KỸ THUẬT	47
4.1. Tích hợp Chatbot AI với mô hình Llama	47
4.1.1. Vấn đề	47
4.1.2. Cách giải quyết	47
4.2. Giới thiệu nền tảng FPT AI Cloud.....	47
4.2.1. Giới thiệu	47
4.2.2. Cách sử dụng	48
KẾT LUẬN.....	50
I. KẾT QUẢ ĐẠT ĐƯỢC.....	50
Về chương trình	50
Về giá trị thực tiễn	50
II. HẠN CHẾ	50
III. HƯỚNG PHÁT TRIỂN	51
TÀI LIỆU THAM KHẢO.....	51

MỤC CÁC BẢNG

Bảng 3-1 Mô tả Use Case đăng nhập.....	18
Bảng 3-2 Dòng chính Use Case đăng nhập	19
Bảng 3-3 Mô tả Use Case đăng ký	20
Bảng 3-4 Mô tả Use Case Tìm kiếm sân.....	22
Bảng 3-5 Mô tả Use Case đặt sân.....	23
Bảng 3-6 Mô tả Use Case quản lý trung tâm.....	25
Bảng 3-7 Mô tả Use Case quản lý sân của chủ sân	27
Bảng 3-8 Mô tả Use Case theo dõi booking và thống kê	29
Hình 3.14 Biểu đồ tuần tự theo dõi booking và thống kê.....	29
Bảng 3-9 Mô tả Use Case hỗ trợ tìm kiếm thông tin AI	31
Bảng 3-20 Mô tả các bảng cơ sở dữ liệu	33
Bảng 3-21 Mô tả bảng User.....	34
Bảng 3-22 Mô tả bảng SportField.....	34
Bảng 3-23 Mô tả bảng SportField.....	35
Bảng 3-24 Mô tả bảng RentalSlot.....	35
Bảng 3-25 Mô tả bảng SessionChat.....	35
Bảng 3-26 Mô tả bảng ProfessionalSkills	36

DANH MỤC HÌNH ẢNH

Hình 2.1 Django REST Framework	7
Hình 2.2 NextJS	9
Hình 2.3 PostgreSQL	10
Hình 2.4 Kiểu ràng buộc trong PostgreSQL	12
Hình 2.5 Llama (Large Language Model Meta AI)	13
Hình 3.1 Use Case chức năng đăng nhập	18
Hình 3.2 Sơ đồ tuần tự chức năng đăng nhập	19
Hình 3.3 Use Case chức năng đăng ký	19
Hình 3.4 Biểu đồ tuần tự đăng ký của người tìm việc	21
Hình 3.5 Use Case tìm kiếm sân	21
Hình 3.7 Use Case đặt sân	22
Hình 3.10 Biểu đồ tuần tự quản lý trung tâm	26
Hình 3.11 Use Case lý sân của chủ sân	27
Hình 3.12 Biểu đồ tuần tự quản lý sân của chủ sân	28
Hình 3.13 Use Case theo dõi booking và thống kê	28
Hình 3.15 Use Case hỗ trợ tìm kiếm thông tin	30
Hình 3.16 Biểu đồ tuần tự hỗ trợ tìm kiếm thông tin AI	32
Hình 3.17 Mô hình dữ liệu quan hệ	33
Hình 3.18 Giao diện đăng nhập	36
Hình 3.19 Giao diện đăng ký	36
Hình 4.1 FPT AI Cloud	48

DANH MỤC CÁC KÍ HIỆU, CHỮ VIẾT TẮT

Ký hiệu	Mô tả
API	Application Programming Interface
ES6	ECMA Script 6
SEO	Search Engine Optimization
SSR	Server-side Rendering
tag	Thẻ

Chương 1: GIỚI THIỆU TỔNG QUAN

1.1. Lý do chọn đề tài

Trong bối cảnh đời sống hiện đại, nhu cầu rèn luyện sức khỏe và tham gia các hoạt động thể thao ngày càng gia tăng, đặc biệt tại các thành phố lớn như Đà Nẵng. Việc tìm kiếm và đặt sân thể thao phù hợp đang trở thành nhu cầu thiết yếu của người dân.

Hiện nay, dù có nhiều cơ sở thể thao, nhưng việc quản lý và đặt sân vẫn còn nhiều bất cập. Người dùng thường gặp khó khăn trong việc tìm kiếm thông tin sân, kiểm tra tình trạng sẵn có và thực hiện đặt sân một cách thuận tiện. Đồng thời, các chủ sân cũng chưa có công cụ hiệu quả để quản lý lịch đặt sân, theo dõi doanh thu và tối ưu hóa việc khai thác cơ sở vật chất.

Và đó cũng là lý do em chọn đề tài xây dựng hệ thống quản lý và đặt sân thể thao trực tuyến với sứ mệnh là "Kết nối cộng đồng thể thao - Nâng cao sức khỏe người Việt"

1.2. Lịch sử nghiên cứu

Lịch sử phát triển của các hệ thống đặt sân thể thao trực tuyến có thể được phân thành các giai đoạn khác nhau, từ những phương thức truyền thống đến các ứng dụng công nghệ hiện đại.

Giai đoạn 1: Đặt sân truyền thống

- Ban đầu, việc đặt sân được thực hiện hoàn toàn thủ công thông qua điện thoại hoặc trực tiếp tại cơ sở thể thao.
- Chủ sân ghi chép lịch đặt bằng sổ sách, dễ xảy ra nhầm lẫn và khó quản lý khi số lượng booking tăng cao.

Giai đoạn 2: Hệ thống đặt sân cơ bản

- Xuất hiện các website đơn giản cho phép xem thông tin sân và liên hệ đặt sân.
- Bắt đầu có các trang web hiển thị danh sách cơ sở thể thao, thông tin liên hệ và giá cả.
- Người dùng vẫn phải liên hệ qua điện thoại để xác nhận đặt sân.

Giai đoạn 3: Ứng dụng đặt sân trực tuyến tích hợp

- Các hệ thống tiến hóa thành ứng dụng web đầy đủ với khả năng đặt sân trực tuyến, quản lý lịch sử và thanh toán.
- Tích hợp nhiều tính năng như kiểm tra tình trạng sân theo thời gian thực, thống kê doanh thu, quản lý người dùng.

- Sử dụng công nghệ chatbot AI và giao diện thân thiện để nâng cao trải nghiệm người dùng.

Xây dựng bằng Django và NextJS, hệ thống có thể nổi bật bằng cách tận dụng sức mạnh của backend Django trong xử lý logic nghiệp vụ phức tạp, kết hợp với giao diện hiện đại của NextJS mang lại trải nghiệm người dùng mượt mà. Hệ thống cũng tích hợp chatbot AI sử dụng mô hình ngôn ngữ lớn Llama để hỗ trợ người dùng tìm kiếm và đặt sân thông minh. Xu hướng số hóa trong lĩnh vực thể thao và nhu cầu quản lý hiệu quả làm cho dự án này trở thành một giải pháp cần thiết và phù hợp với thời đại.

1.3. Mục tiêu đề tài

Mục tiêu tổng quát:

Mục tiêu tổng quát của dự án "Nghiên cứu công nghệ Django và NextJS - Xây dựng Hệ thống Quản lý và Đặt sân Thể thao Trực tuyến" là cung cấp một nền tảng toàn diện, hiệu quả để kết nối người dùng với các cơ sở thể thao, đồng thời hỗ trợ chủ sân trong việc quản lý và tối ưu hóa hoạt động kinh doanh trong bối cảnh số hóa ngày càng phát triển.

Mục tiêu cụ thể:

Xây dựng Frontend với NextJS:

- Tạo giao diện người dùng hiện đại, thân thiện và dễ sử dụng.
- Tối ưu hóa trải nghiệm người dùng trong việc tìm kiếm sân, xem thông tin chi tiết và thực hiện đặt sân.
- Đảm bảo giao diện responsive hoạt động mượt mà trên cả máy tính và thiết bị di động.

Xây dựng Backend với Django:

- Thiết kế và triển khai backend với Django, một framework Python mạnh mẽ và bảo mật cao.
- Xây dựng các API RESTful để quản lý người dùng, trung tâm thể thao, sân thể thao và booking.
- Đảm bảo tích hợp an toàn và bảo mật trong việc truy cập cơ sở dữ liệu và xử lý dữ liệu cá nhân.

Website có các chức năng

Tích hợp tính năng tìm kiếm và đặt sân:

- Phát triển chức năng tìm kiếm sân theo nhiều tiêu chí như loại sân, khu vực, trung tâm, mức giá.
- Xây dựng quy trình đặt sân 3 bước (chọn sân → chọn ngày → chọn khung giờ) đơn giản và trực quan.

- Hiện thị tình trạng khung giờ theo thời gian thực để người dùng dễ dàng lựa chọn.

Hệ thống quản lý cho chủ sân:

- Xây dựng dashboard cho phép chủ sân quản lý trung tâm thể thao và danh sách sân của mình.
- Cung cấp công cụ theo dõi booking, thống kê doanh thu theo khoảng thời gian.
- Hỗ trợ thêm, sửa, xóa thông tin trung tâm và sân thể thao một cách linh hoạt.

Tích hợp Chatbot AI hỗ trợ thông minh:

- Tích hợp chatbot AI sử dụng mô hình Llama-3.3-70B-Instruct để tương tác tự nhiên với người dùng.
- Hỗ trợ tìm kiếm thông tin, gợi ý sân phù hợp và hướng dẫn quy trình đặt sân.
- Cung cấp trợ lý ảo thông minh giúp nâng cao trải nghiệm người dùng.

Hệ thống quản trị tổng thể:

- Xây dựng trang quản trị cho Admin quản lý toàn bộ người dùng, trung tâm, sân và booking.
- Cung cấp báo cáo thống kê tổng quan về hoạt động và doanh thu của hệ thống.
- Đảm bảo phân quyền rõ ràng giữa User, Owner và Admin.

1.4. Đối tượng, phạm vi nghiên cứu

Đối tượng nghiên cứu

Nội dung: Công nghệ framework Django và NextJS, cơ sở dữ liệu

PostgreSQL, tích hợp AI chatbot với mô hình ngôn ngữ lớn Llama.

Đối tượng sử dụng: Người dân có nhu cầu tìm kiếm và đặt sân thể thao tại

Đà Nẵng, các chủ sân thể thao và quản trị viên hệ thống.

Phạm vi nghiên cứu

Phạm vi không gian: Các cơ sở thể thao tại thành phố Đà Nẵng.

Phạm vi nội dung: Xây dựng hệ thống quản lý và đặt sân thể thao trực tuyến bằng Django và NextJS, tích hợp chatbot AI hỗ trợ người dùng.

Chương 2: CƠ SỞ LÝ THUYẾT

2.1. Giới thiệu về ngôn ngữ

2.1.1. Python

Python là một ngôn ngữ lập trình bậc cao, thông dịch và hướng đối tượng được phát triển bởi Guido van Rossum và phát hành lần đầu vào năm 1991. Python được thiết kế với triết lý nhấn mạnh vào khả năng đọc hiểu code và cú pháp đơn giản, cho phép lập trình viên diễn đạt ý tưởng với ít dòng code hơn so với nhiều ngôn ngữ khác như C++ hay Java.

Python hỗ trợ đa dạng các mô hình lập trình bao gồm lập trình hướng đối

tượng, lập trình hàm và lập trình thủ tục. Đây là ngôn ngữ được sử dụng rộng rãi trong nhiều lĩnh vực từ phát triển web, phân tích dữ liệu, trí tuệ nhân tạo, học máy đến khoa học dữ liệu và tự động hóa.

Những ưu điểm làm nên chất lượng của ngôn ngữ Python:

- Cú pháp đơn giản và dễ học: Python có cú pháp rất gần với ngôn ngữ tự nhiên, giúp người mới bắt đầu dễ dàng tiếp cận. Việc sử dụng thụt lề (indentation) để xác định khối code thay vì dấu ngoặc nhọn làm cho code trở nên rõ ràng và dễ đọc hơn.
- Thư viện phong phú: Python có một hệ sinh thái thư viện cực kỳ đa dạng và mạnh mẽ. Từ Django, Flask cho phát triển web, đến NumPy, Pandas cho xử lý dữ liệu, TensorFlow và PyTorch cho machine learning, người dùng có thể dễ dàng tìm thấy công cụ phù hợp cho mọi nhu cầu.
- Đa nền tảng: Python có thể chạy trên nhiều hệ điều hành khác nhau như Windows, Linux, macOS mà không cần thay đổi code, đảm bảo tính linh hoạt và khả năng tái sử dụng cao.
- Cộng đồng lớn và tích cực: Python có một trong những cộng đồng lập trình viên lớn nhất thế giới, cung cấp vô số tài liệu học tập, hướng dẫn và hỗ trợ khi gặp vấn đề.
- Tích hợp tốt: Python có khả năng tích hợp dễ dàng với các ngôn ngữ khác như C, C++, Java, giúp tối ưu hiệu suất khi cần thiết.
- Năng suất cao: Với cú pháp ngắn gọn và thư viện phong phú, Python giúp lập trình viên phát triển ứng dụng nhanh chóng, giảm thời gian phát triển và chi phí bảo trì.

2.1.2. JavaScript

Javascript chính là một ngôn ngữ lập trình với khả năng đem tới sự sinh động khi thiết kế website. Đây là dạng ngôn ngữ theo kịch bản, dựa trên chính đối tượng phát triển có sẵn, hoặc là tự định nghĩa ra. Chính vì tính tiện lợi, hiệu quả ứng dụng cao mà ngôn ngữ lập trình này được tin dùng ngày càng nhiều, ứng dụng rộng rãi trong các website hiệu quả.

Việc sử dụng ngôn ngữ Javascript có thể ứng dụng cho mọi trình duyệt khác nhau, hiện được sử dụng phổ biến như Chrome, hay Firefox,... Hơn nữa, đây còn là ngôn ngữ lập trình hoạt động hiệu quả, được hỗ trợ đầy đủ trên các trình duyệt của thiết bị di động. Bởi thế mà việc sử dụng đa dạng, có thể đáp ứng tốt cho nhiều nhu cầu, những đòi hỏi khác nhau của người dùng.

Nhìn chung, ngôn ngữ lập trình Javascript khi sử dụng được đánh giá dễ học, được phát triển bởi Netscape và sử dụng trong hơn 92% các website ngày nay. Việc sử dụng JS có thể gắn trong một element trong website, hoặc có thể trong một sự kiện của trang web tiện lợi và hiệu quả. Chính việc có thể hoạt động nhanh, nhẹ và mượt mà,... nên ngôn ngữ lập trình này càng được tin tưởng và ứng dụng nhiều hơn cho nhu cầu của người dùng. Chúng ta có thể thêm trực tiếp Javascript trực tiếp với HTML, cũng có thể lưu lại trên một files hoàn toàn riêng biệt và sử dụng khi cần thiết.

Những ưu điểm làm nên chất lượng của ngôn ngữ Javascript:

- Sử dụng Javascript được thực hiện từ phía khách hàng: Điều này thức là các mã được tiến hành thực hiện thông qua bộ vi xử lý của chính người dùng, thay vì sử dụng trên máy chủ web. Chính đặc điểm này giúp việc tiết kiệm năng thông, đồng thời giảm những căng thẳng, hoạt động quá nhiều trên máy chủ web dễ dàng hơn.
- Là ngôn ngữ lập trình dễ dàng sử dụng: Đơn giản, dễ dàng tìm hiểu và sử dụng là đặc điểm nổi bật của ngôn ngữ lập trình này. Với những cú pháp khá tương đồng với tiếng Anh thì việc sử dụng Javascript trở nên dễ dàng, dễ tiếp cận hơn rất nhiều.
- Đánh giá cao ở độ nhanh chóng với người dùng: Việc sử dụng ngôn ngữ JS có khả năng đảm bảo khi mã được thực hiện trên thiết bị máy tính của người dùng thì quá trình xử lý, lần kết quả đều hoàn thành gần như ngay lập tức. Thời gian cụ thể phụ thuộc vào từng nhiệm vụ cụ thể mất thời gian nhất định, song thường nó rất nhanh có thể giúp vấn đề được giải quyết.
- Sử dụng đa dạng trên nhiều trình duyệt: Một ưu điểm nổi bật giúp JS ngày càng được tin dùng, ứng dụng phong phú là ở khả năng sử dụng thông qua nhiều trình duyệt khác nhau. Từ các trình duyệt trên máy tính như Chrome, tới Firefox,... hay những trình duyệt trên thiết bị di động,... đều có thể sử dụng được đầy đủ và hiệu quả. Chính vì sự thích hợp với nhiều trình duyệt, nhiều nền tảng khác nhau giúp nó được tin dùng nhiều hơn để đáp ứng tốt cho đòi hỏi của con người.

2.1.3. TypeScript

TypeScript là một dự án mã nguồn mở được phát triển bởi Microsoft, nó có thể được coi là một phiên bản nâng cao của Javascript bởi việc bổ sung tùy chọn kiểu tĩnh và lớp hướng đối tượng mà điều này không có ở Javascript. TypeScript có thể sử dụng để phát triển các ứng dụng chạy ở client-side (Angular2) và server-side (NodeJS).

TypeScript sử dụng tất cả các tính năng của của ECMAScript 2015 (ES6) như classes, modules. Không dừng lại ở đó nếu như ECMAScript 2017 ra đời thì mình tin chắc rằng TypeScript cũng sẽ nâng cấp phiên bản của mình lên để sử dụng mọi kỹ thuật mới nhất từ ECMAScript. Thực ra TypeScript không phải ra đời đầu tiên mà trước đây cũng có một số thư viện như CoffeScript và Dart được phát triển bởi Google, tuy nhiên điểm yếu là hai thư viện này sử dụng cú pháp mới hoàn toàn, điều này khác hoàn toàn với TypeScript, vì vậy tuy ra đời sau nhưng TypeScript vẫn đang nhận được sự đón nhận từ các lập trình viên.

Tại sao nên sử dụng TypeScript:

- Dễ phát triển dự án lớn: Với việc sử dụng các kỹ thuật mới nhất và lập trình hướng đối tượng nên TypeScript giúp chúng ta phát triển các dự án lớn một cách dễ dàng.
- Nhiều Framework lựa chọn: Hiện nay các Javascript Framework đã dần khuyến khích nên sử dụng TypeScript để phát triển, ví dụ như AngularJS 2.0 và Ionic 2.0.
- Hỗ trợ các tính năng của Javascript phiên bản mới nhất: TypeScript luôn đảm bảo việc sử dụng đầy đủ các kỹ thuật mới nhất của Javascript, ví dụ như version hiện tại là ECMAScript 2015 (ES6).
- Là mã nguồn mở: TypeScript là một mã nguồn mở nên bạn hoàn toàn có thể sử dụng mà không mất phí, bên cạnh đó còn được cộng đồng hỗ trợ.
- TypeScript là Javascript: Bản chất của TypeScript là biên dịch tạo ra các đoạn mã javascript nên bạn có thể chạy bất kì ở đâu miễn ở đó có hỗ trợ biên dịch Javascript. Ngoài ra bạn có thể sử dụng trộn lẫn cú pháp của Javascript vào bên trong TypeScript, điều này giúp các lập trình viên tiếp cận Typescript dễ dàng hơn.

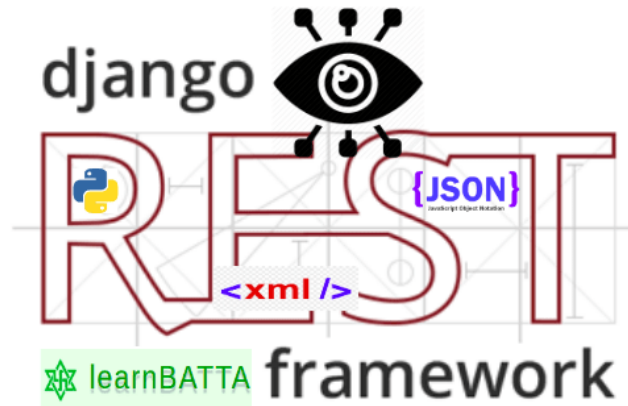
2.2. Giới thiệu về Django REST Framework

2.2.1. Django REST Framework là gì?

Django REST Framework (DRF) là một thư viện mạnh mẽ và linh hoạt được xây dựng trên nền tảng Django, chuyên dụng cho việc phát triển Web API (RESTful API). DRF được phát triển để đơn giản hóa quá trình tạo và quản lý các API endpoints, cung cấp các công cụ và tính năng cần thiết để xây dựng các API có hiệu suất cao, dễ bảo trì và tuân thủ các chuẩn REST.

Django REST Framework cung cấp một bộ công cụ hoàn chỉnh bao gồm serialization, authentication, permissions, viewsets và routing, giúp lập trình viên nhanh chóng xây dựng các API backend cho ứng dụng web và mobile. Framework này được thiết kế để tích hợp liền mạch với Django, tận dụng tất cả các tính năng

mạnh mẽ của Django như ORM, authentication system và admin interface.



viewsets in Django Rest Framework

Hình 2.1 Django REST Framework

2.2.2. Cấu trúc của Django REST Framework

Cấu trúc của Django REST Framework được tổ chức theo các thành phần chính sau:

- **Serializers (Bộ tuần tự hóa):** Serializers chịu trách nhiệm chuyển đổi dữ liệu phức tạp như Django models thành các định dạng có thể truyền tải qua mạng (JSON, XML) và ngược lại. Serializers cũng thực hiện validation dữ liệu đầu vào, đảm bảo dữ liệu hợp lệ trước khi lưu vào database.
- **View (Khung nhìn):** Views xử lý logic nghiệp vụ của API endpoints. DRF cung cấp nhiều loại views như APIView, generic views (ListAPIView, RetrieveAPIView, CreateAPIView, UpdateAPIView, DestroyAPIView) và ViewSets. ViewSets kết hợp nhiều operations (list, create, retrieve, update, destroy) vào một class duy nhất, giúp code ngắn gọn và dễ quản lý hơn.
- **Routers (Bộ định tuyến):** Routers tự động tạo URL patterns cho ViewSets. DRF cung cấp DefaultRouter và SimpleRouter để ánh xạ ViewSets với các URL endpoints tương ứng một cách tự động.
- **Authentication (Xác thực):** DRF hỗ trợ nhiều phương thức xác thực như Token Authentication, Session Authentication, JWT Authentication, OAuth2. Điều này cho phép kiểm soát truy cập vào API một cách linh hoạt và bảo mật.
- **Permissions (Phân quyền):** Hệ thống permissions cho phép kiểm soát chi tiết quyền truy cập của người dùng vào các API endpoints. DRF cung cấp các

permission classes như `IsAuthenticated`, `IsAdminUser`, `AllowAny`, và cho phép tạo custom permissions.

- **Pagination (Phân trang):** DRF cung cấp các class pagination tích hợp sẵn để phân trang kết quả trả về, giúp tối ưu hiệu suất khi làm việc với tập dữ liệu lớn.
- **Filtering và Searching:** DRF hỗ trợ filtering, searching và ordering dữ liệu thông qua các backend như `DjangoFilterBackend`, `SearchFilter` và `OrderingFilter`.
- **Browsable API:** DRF tự động tạo ra một giao diện web tương tác cho phép developers test và explore API endpoints trực tiếp từ trình duyệt.

2.2.3. Hướng dẫn cài đặt Django REST Framework

Để cài đặt Django REST Framework, bạn cần đảm bảo đã cài đặt Django trước. Dưới đây là hướng dẫn cài đặt:

Bước 1: Cài đặt Django (nếu chưa có)

```
``` pip install django ```
```

**Bước 2:** Cài đặt Django REST Framework

```
``` pip install  
djangorestframework ```
```

Bước 3: Thêm DRF vào `INSTALLED_APPS`

Mở file `settings.py` trong project Django và thêm `'rest_framework'` vào `INSTALLED_APPS`:

```
INSTALLED_APPS = [  
  
    ...  
  
    'rest_framework',  
]
```

Bước 4: Cấu hình REST Framework (tùy chọn)

Thêm cấu hình cơ bản cho REST Framework trong `settings.py`.

Bước 5: Tạo Serializer và ViewSet đầu tiên.

Bước 6: Cấu hình URLs.

Sau khi hoàn thành các bước trên, bạn có thể chạy server và truy cập vào <http://localhost:8000/api/> để xem API endpoints của mình.

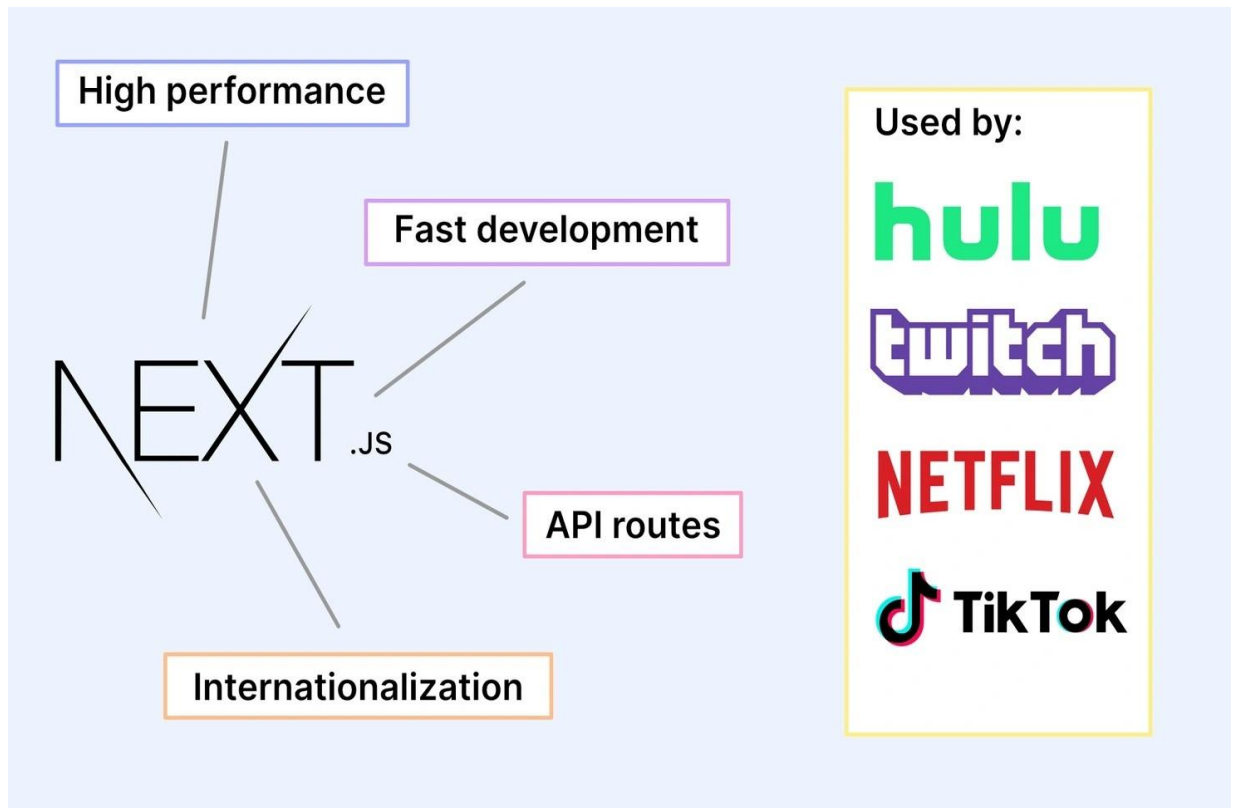
2.3. Giới thiệu về Framework NextJS

2.3.1. Framework NextJS là gì?

Next.js là một framework front-end React được phát triển dưới dạng open-source bổ sung các khả năng tối ưu hóa như render phía máy chủ (SSR) và tạo

trang web static. Next.js xây dựng dựa trên thư viện React, có nghĩa là các ứng dụng Next.js sử dụng core của React và chỉ thêm các tính năng bổ sung. Việc triển khai ứng dụng SSR cho phép máy chủ truy cập tất cả dữ liệu được yêu cầu và xử lý JavaScript cùng nhau để

hiển thị trang. Sau đó, trang được gửi lại toàn bộ cho trình duyệt và ngay lập tức được hiển thị. SSR cho phép các trang web load trong thời gian nhỏ nhất và tăng trải nghiệm người dùng với khả năng phản hồi nhanh hơn.



Hình 2.2 NextJS

2.3.2. Lợi thế của NextJS

- SEO website rất tốt và giúp giảm chi phí, giúp trang web của bạn hiển thị cao hơn trên các trang kết quả của công cụ tìm kiếm. SSR làm cho các trang web xếp hạng tốt hơn cho SEO vì chúng tải nhanh hơn và nhiều nội dung trang web có thể được quét bởi các trình theo dõi SEO
- Next.js cũng cho phép chỉnh sửa thẻ <head> của một trang web, điều mà không thể thực hiện trong React. Các tag <head> là một phần cốt lõi của siêu dữ liệu trên trang web và góp phần vào đánh giá xếp hạng SEO của trang web.
- Tích hợp bộ nhớ đệm và tối ưu hóa tĩnh tự động
- Server phụ trách hiển thị các trang
- Framework của React vì thế nếu làm tốt với React thì có thể code Next.js dễ hơn
- Tinh chỉnh cấu hình webpack / babel

- Hỗ trợ tích hợp cho Route Page, CSS, JSX và TypeScript
- Nhanh chóng thêm các plugin để tùy chỉnh Next.js theo nhu cầu của trang cụ thể

2.3.3. Nhược điểm của NextJS

Nhược điểm thực sự duy nhất của NextJS là nó là một opinionated framework, có nghĩa là nó có một phương pháp và bộ công cụ cụ thể mà nó muốn bạn sử dụng để xây dựng các ứng dụng của mình.

2.4. Giới thiệu về cơ sở dữ liệu PostgreSQL

2.4.1. PostgreSQL là gì?

PostgreSQL (thường được gọi là Postgres) là một hệ quản trị cơ sở dữ liệu quan hệ đối tượng (ORDBMS) mã nguồn mở mạnh mẽ, được phát triển từ năm 1986 tại Đại học California, Berkeley. PostgreSQL được biết đến với độ tin cậy cao, tính năng phong phú và khả năng mở rộng tốt, làm cho nó trở thành một trong những hệ quản trị cơ sở dữ liệu được sử dụng rộng rãi nhất trên thế giới.



Hình 2.3 PostgreSQL

Đặc Điểm Nổi Bật:

- Tuân thủ chuẩn SQL: PostgreSQL tuân thủ nghiêm ngặt các chuẩn SQL, đảm bảo tính tương thích và khả năng chuyển đổi giữa các hệ thống.
- Hỗ trợ dữ liệu phức tạp: PostgreSQL hỗ trợ nhiều kiểu dữ liệu tiên tiến như JSON, JSONB, Array, XML, và cho phép người dùng tự định nghĩa kiểu dữ liệu riêng.
- ACID Compliance: PostgreSQL đảm bảo tính toàn vẹn dữ liệu thông qua việc tuân thủ các thuộc tính ACID (Atomicity, Consistency, Isolation, Durability).
- Khả năng mở rộng: PostgreSQL hỗ trợ nhiều extension mạnh mẽ như PostGIS cho

dữ liệu không gian địa lý, pg_stat_statements cho giám sát hiệu suất, và nhiều extension khác.

- Hiệu suất cao: PostgreSQL được tối ưu hóa cho các truy vấn phức tạp với các tính năng như indexing tiên tiến (B-tree, Hash, GiST, GIN), query optimization, và parallel query execution.
- Bảo mật tốt: PostgreSQL cung cấp nhiều tính năng bảo mật như Row Level Security (RLS), SSL connection, và hỗ trợ nhiều phương thức xác thực.
- Cộng đồng mạnh: PostgreSQL có một cộng đồng phát triển tích cực với documentation đầy đủ và hỗ trợ tốt.

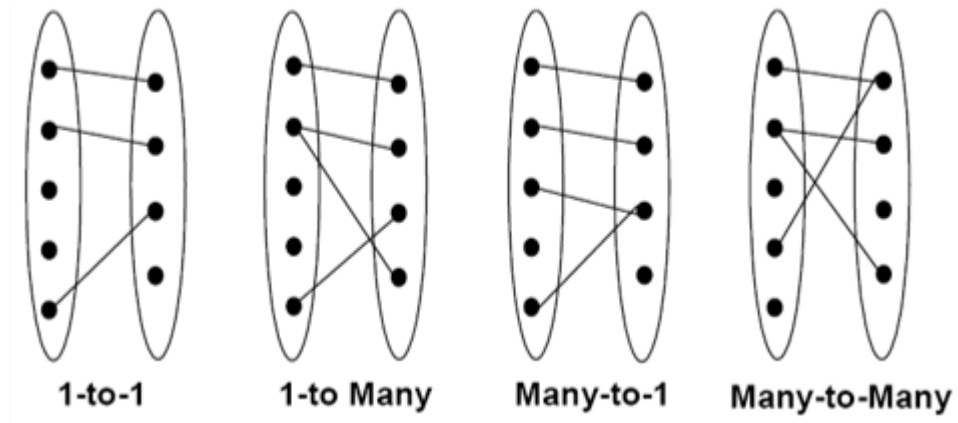
2.4.2. Mối liên hệ và ràng buộc trong MySQL

PostgreSQL hỗ trợ đầy đủ các ràng buộc (constraints) để đảm bảo tính toàn vẹn và chính xác của dữ liệu.

- PRIMARY KEY: Xác định duy nhất mỗi bản ghi trong bảng. Một bảng chỉ có thể có một PRIMARY KEY, có thể bao gồm một hoặc nhiều cột.
- FOREIGN KEY: Thiết lập mối quan hệ giữa các bảng, đảm bảo tính toàn vẹn tham chiếu. FOREIGN KEY trong bảng này tham chiếu đến PRIMARY KEY của bảng khác.
- UNIQUE: Đảm bảo tất cả giá trị trong cột hoặc nhóm cột là duy nhất.
- NOT NULL: Đảm bảo cột không chứa giá trị NULL.
- CHECK: Đảm bảo giá trị trong cột thỏa mãn điều kiện cụ thể.
- DEFAULT: Gán giá trị mặc định cho cột khi không có giá trị được chỉ định

Các kiểu quan hệ trong MySQL:

- One-to-One (1:1): Một bản ghi trong bảng A liên kết với đúng một bản ghi trong bảng B.
- One-to-Many (1:N): Một bản ghi trong bảng A có thể liên kết với nhiều bản ghi trong bảng B.
- Many-to-Many (M:N): Nhiều bản ghi trong bảng A có thể liên kết với nhiều bản ghi trong bảng B, thường được thực hiện thông qua bảng trung gian (junction table).



Hình 2.4 Kiểu ràng buộc trong PostgreSQL

2.5. Giới thiệu về mô hình ngôn ngữ lớn Llama

2.5.1. Mô hình ngôn ngữ lớn Llama là gì?

Llama (Large Language Model Meta AI) là một họ mô hình ngôn ngữ lớn (Large Language Model - LLM) được phát triển bởi Meta AI. Đây là các mô hình trí tuệ nhân tạo được huấn luyện trên khối lượng dữ liệu văn bản khổng lồ, cho phép chúng hiểu và tạo ra ngôn ngữ tự nhiên với độ chính xác cao.

Mô hình Llama được thiết kế để thực hiện nhiều tác vụ xử lý ngôn ngữ tự nhiên (NLP) như trả lời câu hỏi, tóm tắt văn bản, dịch thuật, viết nội dung sáng tạo, và hỗ trợ đối thoại. Điểm đặc biệt của Llama là được phát hành dưới dạng mã nguồn mở, cho phép cộng đồng nghiên cứu và phát triển các ứng dụng dựa trên mô hình này.



Hình 2.5 Llama (Large Language Model Meta AI)

2.5.2. Ứng dụng của mô hình Llama trong hệ thống

- Trả lời câu hỏi tự nhiên: Người dùng có thể đặt câu hỏi bằng ngôn ngữ tự nhiên về các sân thể thao, giá cả, vị trí, và nhận được câu trả lời chính xác ngay lập tức.
- Gợi ý thông minh: Chatbot có thể đề xuất các sân thể thao phù hợp dựa trên nhu cầu, vị trí, mức giá và thời gian của người dùng.
- Hướng dẫn sử dụng: Chatbot có thể hướng dẫn người dùng từng bước cách tìm kiếm sân, đặt sân, và quản lý lịch đặt sân của họ.
- Xử lý ngôn ngữ đa dạng: Mô hình có khả năng hiểu và phản hồi trong nhiều ngữ cảnh khác nhau, từ các câu hỏi đơn giản đến các yêu cầu phức tạp.
- Tương tác linh hoạt: Người dùng có thể tương tác với chatbot thông qua cửa sổ chat toàn màn hình hoặc nút chatbot nổi xuất hiện trên hầu hết các trang của hệ thống.

2.6. Công cụ phát triển Visual Studio Code

2.6.1. Visual Studio Code là gì

Visual Studio Code chính là ứng dụng cho phép biên tập, soạn thảo các đoạn code để hỗ trợ trong quá trình thực hiện xây dựng, thiết kế website một cách nhanh chóng. Visual Studio Code hay còn được viết tắt là VS Code. Trình soạn thảo này vận hành mượt mà trên các nền tảng như Windows, macOS, Linux. Hơn thế nữa, VS Code còn cho khả năng tương thích với những thiết bị máy tính có cấu hình tầm trung vẫn có thể sử dụng dễ dàng.

Visual Studio Code hỗ trợ đa dạng các chức năng Debug, đi kèm với Git, có Syntax Highlighting. Đặc biệt là tự hoàn thành mã thông minh, Snippets, và khả năng cải tiến mã nguồn. Nhờ tính năng tùy chỉnh, Visual Studio Code cũng cho phép các lập trình viên thay đổi Theme, phím tắt, và đa dạng các tùy chọn khác. Mặc dù trình soạn thảo Code này tương đối nhẹ, nhưng lại bao gồm các tính năng mạnh mẽ.

2.6.2. Tại sao nên sử dụng Visual Studio Code?

- Hỗ trợ đa nền tảng: Linux, Mac, Windows.
- Hỗ trợ đa ngôn ngữ: Python, JavaScript, TypeScript, C/C++, Java, JSON, HTML, CSS,...
- Ít dung lượng: Chỉ khoảng 100MB, khởi động nhanh chóng.

- Tính năng mạnh mẽ: Extensions phong phú, tích hợp Git, terminal tích hợp, debugging tools.
- Intellisense chuyên nghiệp: Tự động hoàn thành code, gợi ý thông minh, và phát hiện lỗi ngay khi viết.
- Giao diện thân thiện: Giao diện tối giản, dễ sử dụng, có thể tùy chỉnh theo sở thích.
- Cộng đồng lớn: Số lượng người sử dụng đông đảo tạo nên cộng đồng hỗ trợ rộng rãi với hàng ngàn extension miễn phí.
- Miễn phí và mã nguồn mở: Hoàn toàn miễn phí, được phát triển bởi Microsoft với cộng đồng đóng góp tích cực.

Chương 3: PHÂN TÍCH, THIẾT KẾ HỆ THỐNG

3.1. Mô tả bài toán

- Tìm hiểu về hoạt động, cách vận hành của hệ thống quản lý và đặt sân thể thao.
- Xác định các giải pháp xây dựng hệ thống quản lý và đặt sân thể thao như thế nào.
- Xây dựng website có các chức năng cơ bản như:

 Tìm kiếm công việc: Cho phép người dùng tìm kiếm công việc dựa trên lĩnh vực, mức lương.

 Xem thông tin công việc: Hiển thị thông tin chi tiết về công việc bao gồm mô tả công việc, yêu cầu công việc, mức lương, địa điểm làm việc, thời gian làm việc,....

 Đăng ký và đăng nhập: Cho phép người dùng tạo tài khoản, đăng nhập để quản lý hồ sơ cá nhân và lưu lại công việc quan tâm.

 Ứng tuyển công việc: Cung cấp chức năng ứng tuyển công việc, gửi hồ sơ xin việc trực tuyến tới nhà tuyển dụng.

3.1.1. Yêu cầu người dùng thường (User)

Chức năng đăng ký và đăng nhập

Khi người dùng muốn tạo tài khoản mới, họ truy cập vào trang đăng ký và điền thông tin cá nhân như họ tên, email, mật khẩu. Sau khi điền đầy đủ và ấn nút "Đăng ký", hệ thống sẽ kiểm tra tính hợp lệ của thông tin. Nếu thông tin hợp lệ, hệ thống sẽ tạo tài khoản và thông báo đăng ký thành công. Khi đăng nhập, người dùng nhập tài khoản và mật khẩu. Nếu hệ thống xác thực đúng, người dùng sẽ được chuyển đến trang chủ và có thể sử dụng đầy đủ các chức năng của hệ thống. Khi đăng nhập thành công, người dùng sẽ được cấp token để xác thực các request tiếp theo

Chức năng quản lý thông tin cá nhân

Người dùng có thể xem và chỉnh sửa thông tin tài khoản như họ tên, email, số điện thoại, địa chỉ. Hệ thống cho phép cập nhật thông tin một cách dễ dàng để đảm bảo dữ liệu luôn chính xác.

Chức năng tìm kiếm và xem thông tin trung tâm, sân thể thao

Người dùng có thể tìm kiếm các trung tâm thể thao theo tên, địa chỉ, quận/huyện. Khi chọn một trung tâm, hệ thống hiển thị thông tin chi tiết bao gồm địa chỉ, hình ảnh và danh sách các sân thuộc trung tâm đó. Người dùng cũng có thể xem danh sách các sân thể thao và lọc theo trung tâm, loại sân (bóng đá, bóng rổ, cầu lông,...), khu vực, mức giá. Thông tin chi tiết của mỗi sân bao gồm tên sân, loại sân, giá thuê theo giờ, hình ảnh và thông tin trung tâm quản lý.

Chức năng đặt sân thể thao

Quy trình đặt sân được thiết kế đơn giản với 3 bước:

Bước 1 - Chọn sân: Người dùng chọn sân thể thao mong muốn từ danh sách hoặc kết quả tìm kiếm.

Bước 2 - Chọn ngày: Chọn ngày muốn đặt sân từ lịch.

Bước 3 - Chọn khung giờ: Hệ thống hiển thị các khung giờ trong ngày với tình trạng (trống/đã được đặt). Người dùng chọn khung giờ trống phù hợp.

Sau khi xác nhận đặt sân, hệ thống lưu thông tin booking và thông báo kết quả cho người dùng. Người dùng cũng có thể hủy lượt đặt sân trong lịch cá nhân nếu được phép

Chức năng xem và quản lý lịch sử đặt sân

Người dùng có thể xem danh sách tất cả các lượt đặt sân của mình, bao gồm thông tin về sân, trung tâm, ngày giờ, giá và trạng thái. Hệ thống cho phép lọc lịch sử theo trạng thái (đã đặt, hoàn thành, đã hủy). Người dùng có thể xem riêng các booking sắp tới, đã hoàn thành, đã hủy và thực hiện các thao tác như hủy hoặc đặt lại một số lượt đặt khi cần thiết.

3.1.2. Yêu cầu chủ sân (Owner)

Chủ sân có đầy đủ các chức năng của người dùng thường, đồng thời có thêm các chức năng quản lý riêng:

Chức năng xem tổng quan hoạt động

Chủ sân có thể xem dashboard tổng quan hiển thị số lượng trung tâm, số lượng sân thuộc sở hữu, tổng doanh thu và tổng số booking trong khoảng thời gian lựa chọn. Dashboard cung cấp cái nhìn tổng thể về hoạt động kinh doanh

Chức năng quản lý trung tâm thể thao:

Chủ sân có thể xem danh sách các trung tâm do mình quản lý, thêm trung tâm mới với thông tin như tên và địa chỉ, chỉnh sửa thông tin trung tâm (tên, địa chỉ, hình ảnh) và xóa trung tâm khi không còn hoạt động (nếu thỏa điều kiện hệ thống).

Chức năng quản lý sân thể thao:

Đối với mỗi trung tâm, chủ sân có thể:

- Xem danh sách các sân thuộc trung tâm
- Thêm sân mới: chọn trung tâm, đặt tên sân, chọn loại sân (bóng đá, bóng rổ, cầu lông,...), thiết lập giá, trạng thái hoạt động và thêm hình ảnh
- Chỉnh sửa thông tin sân (tên, loại, giá, trạng thái) và cập nhật/bổ sung hình ảnh
- Xóa sân không còn sử dụng

Chức năng theo dõi booking và doanh thu:

Chủ sân có thể xem danh sách các booking liên quan đến các sân mình sở hữu, với khả năng lọc theo trung tâm, sân, ngày, trạng thái, người dùng.

Hệ thống cung cấp thống kê doanh thu chi tiết theo khoảng thời gian, theo trung tâm, theo trạng thái booking. Dữ liệu được hiển thị dưới dạng số liệu và biểu đồ trực quan, giúp chủ sân dễ dàng phân tích và ra quyết định kinh doanh.

3.1.3. Yêu cầu quản trị viên (Admin)

Quản trị viên có quyền quản lý toàn bộ hệ thống:

Chức năng xem tổng quan hệ thống:

Admin có thể xem dashboard tổng quan hiển thị số lượng người dùng, trung tâm, sân thể thao, tổng doanh thu và tổng số lượt đặt sân trong toàn hệ thống.

Chức năng quản lý người dùng:

Admin có toàn quyền quản lý tài khoản người dùng bao gồm:

- Xem danh sách toàn bộ tài khoản.
- Tạo mới tài khoản với vai trò User/Owner/Admin.
- Chỉnh sửa thông tin người dùng, phân quyền (thay đổi vai trò) và đặt lại mật khẩu.
- Xóa tài khoản không còn sử dụng.

Chức năng quản lý trung tâm thể thao:

- Xem danh sách tất cả trung tâm, tìm kiếm theo tên, địa chỉ, khu vực.
- Tạo mới trung tâm và gán cho một chủ sân (Owner) cụ thể.
- Chỉnh sửa thông tin trung tâm, thay đổi chủ sở hữu khi cần thiết.

- Xóa trung tâm và quản lý hình ảnh minh họa.

Chức năng quản lý sân thể thao:

- Xem danh sách tất cả sân trên hệ thống kèm thông tin trung tâm, loại sân, giá, trạng thái.
- Tìm kiếm, lọc sân theo trung tâm, loại sân, địa chỉ, giá.
- Tạo mới sân, chỉnh sửa thông tin sân, xóa sân, quản lý hình ảnh sân.

Chức năng quản lý booking và thống kê:

- Xem toàn bộ các booking, lọc theo trung tâm, sân, chủ sân, người dùng, thời gian, trạng thái
- Theo dõi tổng doanh thu, số lượng booking theo khoảng thời gian và theo trạng thái
- Xem thống kê doanh thu theo từng trung tâm dưới dạng bảng và biểu đồ
- Phân tích dữ liệu để hỗ trợ ra quyết định quản lý hệ thống.

3.1.4. Chức năng Chatbot AI hỗ trợ người dùng

Hệ thống tích hợp chatbot AI sử dụng mô hình ngôn ngữ lớn Llama để:

Hỗ trợ tìm kiếm thông tin:

Người dùng có thể đặt câu hỏi tự nhiên về trung tâm, sân thể thao, loại sân, khu vực, giá cả. Chatbot phân tích câu hỏi và trả lời dưới dạng hội thoại, giúp người dùng nhanh chóng nắm được thông tin cần thiết.

Hỗ trợ quy trình đặt sân:

- Gợi ý các sân phù hợp với nhu cầu người dùng (ví dụ: sân bóng đá ở quận Hải Châu, giá dưới 300.000đ/giờ)
- Hướng dẫn từng bước cách tìm sân, chọn ngày, chọn giờ và thực hiện đặt sân trên hệ thống
- Giải đáp thắc mắc trong quá trình sử dụng

Hỗ trợ trong nhiều ngữ cảnh:

- Cửa sổ chat toàn màn hình: Cho trải nghiệm tương tác tập trung
- Nút chatbot nổi: Hiển thị trên hầu hết các trang, cho phép người dùng vừa thao tác trên giao diện (xem sân, đặt sân) vừa có thể hỏi AI để được hỗ trợ ngay lập tức

Chatbot giúp nâng cao trải nghiệm người dùng bằng cách cung cấp hỗ trợ 24/7, giảm thời gian tìm kiếm thông tin và đơn giản hóa quy trình đặt sân.

3.2. Phân tích hệ thống

3.2.1. Chức năng đăng nhập của người tìm việc

- Biểu đồ Use Case đăng nhập



Hình 3.1 Use Case chức năng đăng nhập

- Mô tả Use Case đăng nhập của người tìm việc

Use Case	Đăng nhập người tìm việc
Mục tiêu	Người dùng đăng nhập vào tài khoản để sử dụng hệ thống
Tác nhân	Người dùng (User/Owner/Admin)
Tiền điều kiện	Người dùng đã có tài khoản và truy cập trang đăng nhập
Kết quả đầu ra	Người dùng đăng nhập thành công và có quyền truy cập vào các chức năng tương ứng với vai trò
Luồng sự kiện chính	1. Người dùng truy cập trang đăng nhập. 2. Hệ thống hiển thị form đăng nhập yêu cầu email và mật khẩu. 3. Người dùng nhập thông tin đăng nhập. 4. Hệ thống kiểm tra tính hợp lệ của thông tin và xác định vai trò người dùng. 5. Nếu thông tin đúng, hệ thống tạo token xác thực và chuyển hướng người dùng đến trang chủ tương ứng với vai trò. 6. Token được lưu để xác thực các request tiếp theo
Luồng sự kiện phụ	Nếu thông tin đăng nhập không hợp lệ, hệ thống thông báo cho người tìm việc và yêu cầu nhập lại.

Bảng 3-1 Mô tả Use Case đăng nhập

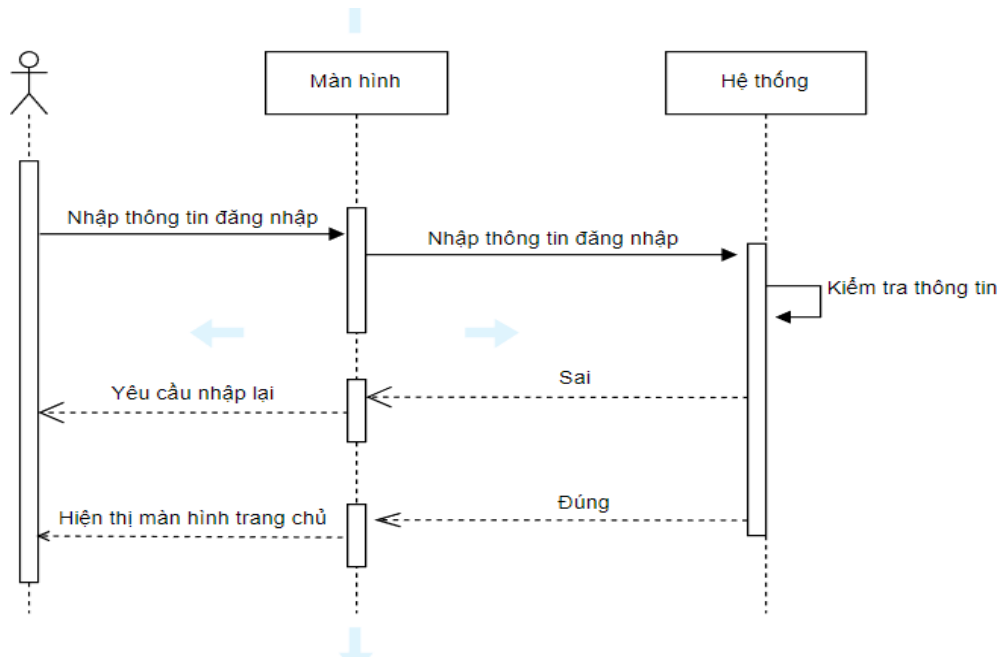
- Dòng chính Use Case đăng nhập

STT	Thực hiện	Hành động
1	Người dùng	Truy cập vào trang đăng nhập
2	Hệ thống	Hiển thị form đăng nhập với các trường tài khoản và mật khẩu.

3	Người dùng	Nhập tài khoản và mật khẩu, nhấn nút "Đăng nhập"
4	Hệ thống	Kiểm tra thông tin đăng nhập với database
5	Hệ thống	Nếu đúng: tạo token, xác định vai trò và chuyển hướng đến trang chủ Nếu sai: hiển thị thông báo lỗi

Bảng 3-2 Dòng chính Use Case đăng nhập

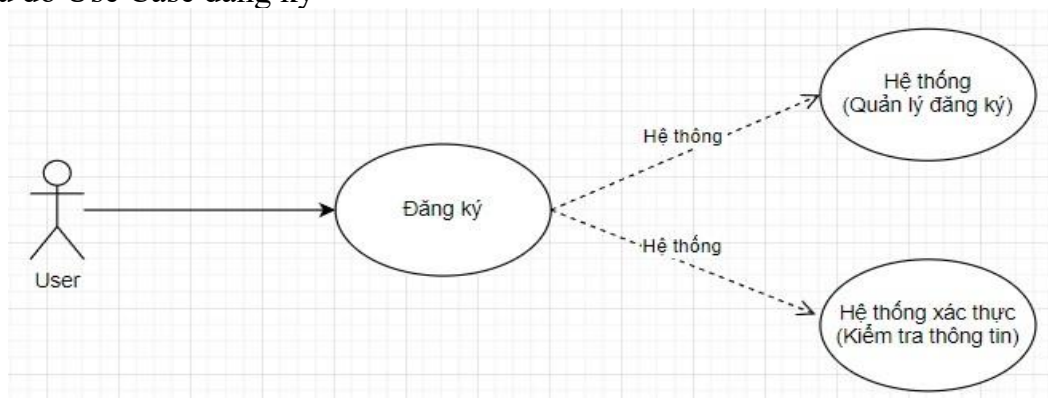
- Biểu đồ tuần tự đăng nhập



Hình 3.2 Sơ đồ tuần tự chức năng đăng nhập

3.2.2. Chức năng đăng ký của người dùng

- Biểu đồ Use Case đăng ký



Hình 3.3 Use Case chức năng đăng ký

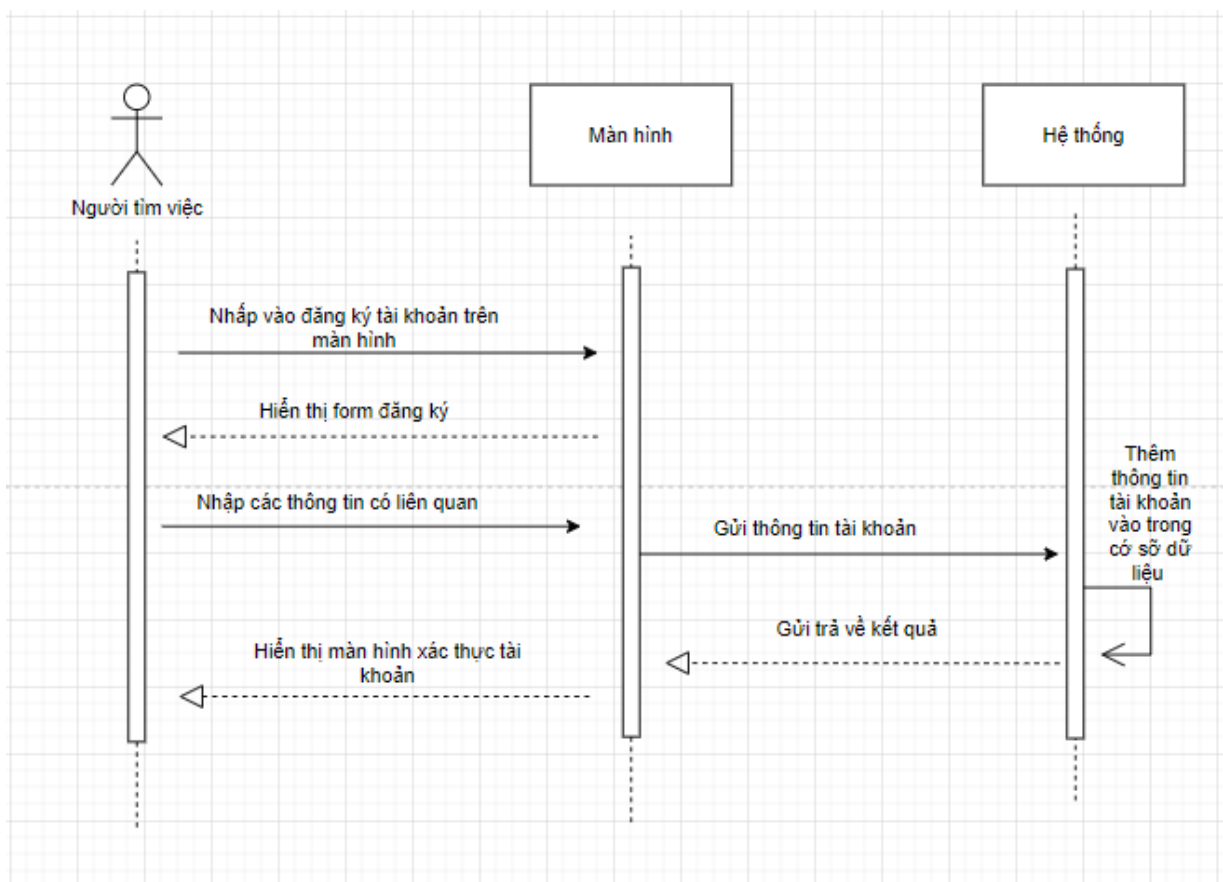
- Mô tả Use Case đăng ký

Use Case	Đăng ký tài khoản
----------	-------------------

Mục tiêu	Người dùng tạo tài khoản mới để sử dụng hệ thống
Tác nhân	Người dùng mới
Tiền điều kiện	Người dùng truy cập được trang đăng ký
Kết quả đầu ra	Tài khoản được tạo thành công và người dùng có thể đăng nhập
Luồng sự kiện chính	1. Người dùng truy cập trang đăng ký 2. Hệ thống hiển thị form đăng ký 3. Người dùng nhập thông tin: họ tên, email, mật khẩu, xác nhận mật khẩu 4. Người dùng xác nhận đăng ký 5. Hệ thống kiểm tra tính hợp lệ: email/tài khoản chưa tồn tại, mật khẩu khớp, đủ độ dài 6. Nếu hợp lệ, hệ thống tạo tài khoản và thông báo đăng ký thành công 7. Người dùng được chuyển hướng đến trang đăng nhập
Luồng sự kiện phụ	Nếu thông tin không hợp lệ hoặc email/tài khoản đã tồn tại, hệ thống thông báo lỗi cụ thể và yêu cầu nhập lại

Bảng 3-3 Mô tả Use Case đăng ký

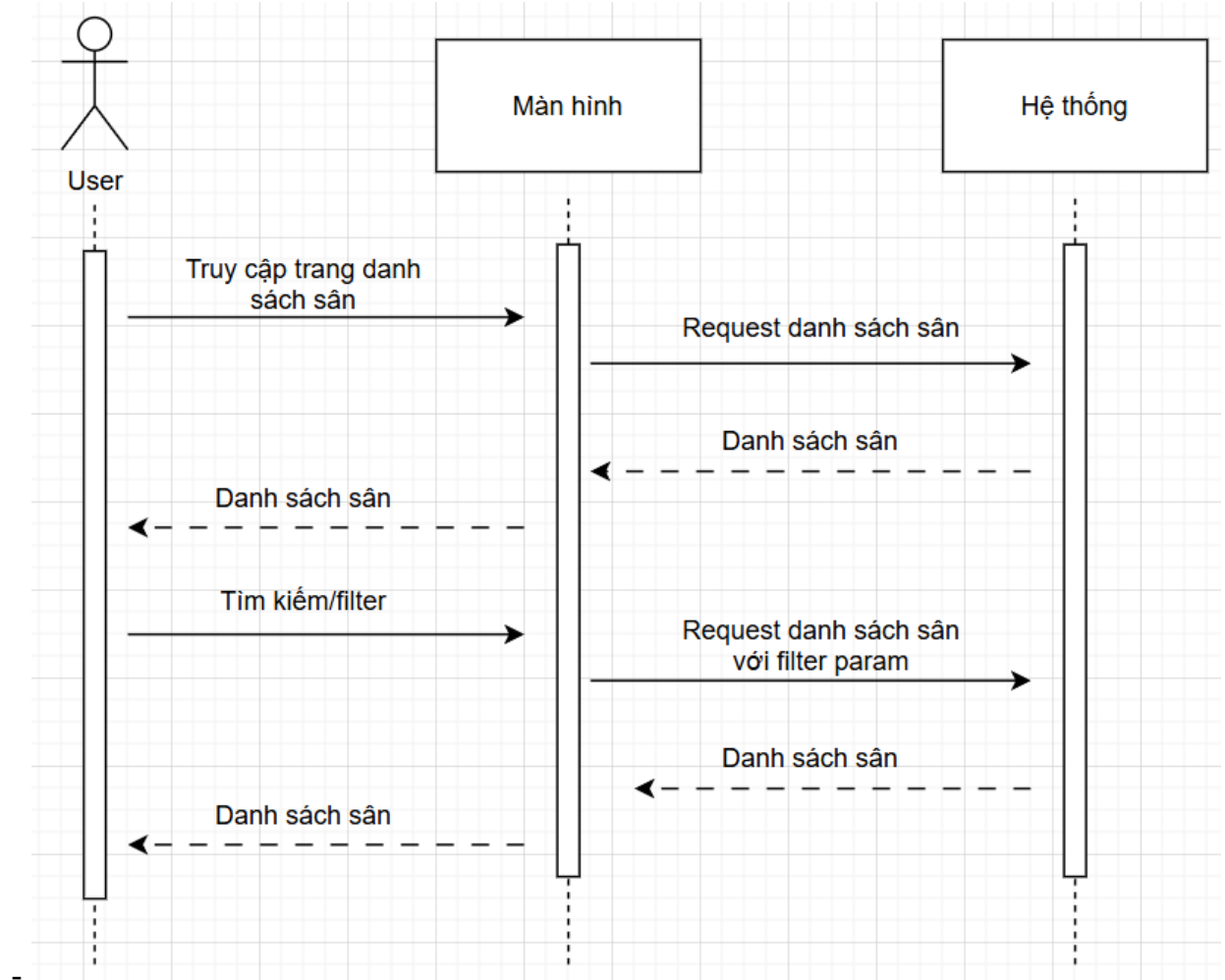
- Biểu đồ tuần tự đăng ký



Hình 3.4 Biểu đồ tuần tự đăng ký của người tìm việc

3.2.3. Chức năng tìm kiếm và xem thông tin sân

- Biểu đồ Use Case tìm kiếm sân



Hình 3.5 Use Case tìm kiếm sân

- Mô tả Use Case cập nhật tiêu đề hồ sơ

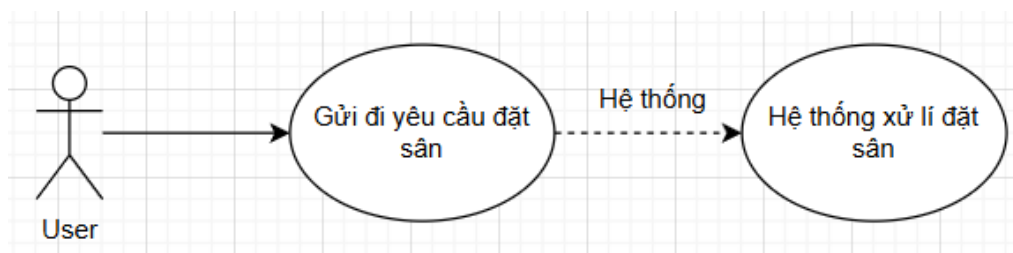
Use Case	Tìm kiếm và xem thông tin sân
Mục tiêu	Người dùng tìm kiếm và xem thông tin về sân thể thao
Tác nhân	Người dùng
Tiền điều kiện	Người dùng truy cập vào hệ thống
Kết quả đầu ra	Hiển thị danh sách sân phù hợp với tiêu chí tìm kiếm
Luồng sự kiện chính	1. Người dùng truy cập trang tìm kiếm sân

	2. Hệ thống hiển thị bộ lọc: loại sân, khu vực, trung tâm, mức giá 3. Người dùng chọn tiêu chí lọc và tìm kiếm 4. Hệ thống truy vấn database và trả về danh sách sân phù hợp 5. Người dùng chọn một sân để xem chi tiết 6. Hệ thống hiển thị thông tin chi tiết: tên, loại, giá, hình ảnh, địa chỉ, trung tâm quản lý
Luồng sự kiện phụ	Nếu không tìm thấy sân phù hợp, hệ thống thông báo "Không tìm thấy kết quả" và gợi ý điều chỉnh tiêu chí tìm kiếm

Bảng 3-4 Mô tả Use Case Tìm kiếm sân

3.2.4. Chức năng đặt sân thể thao

- Biểu đồ Use Case đặt sân



Hình 3.7 Use Case đặt sân

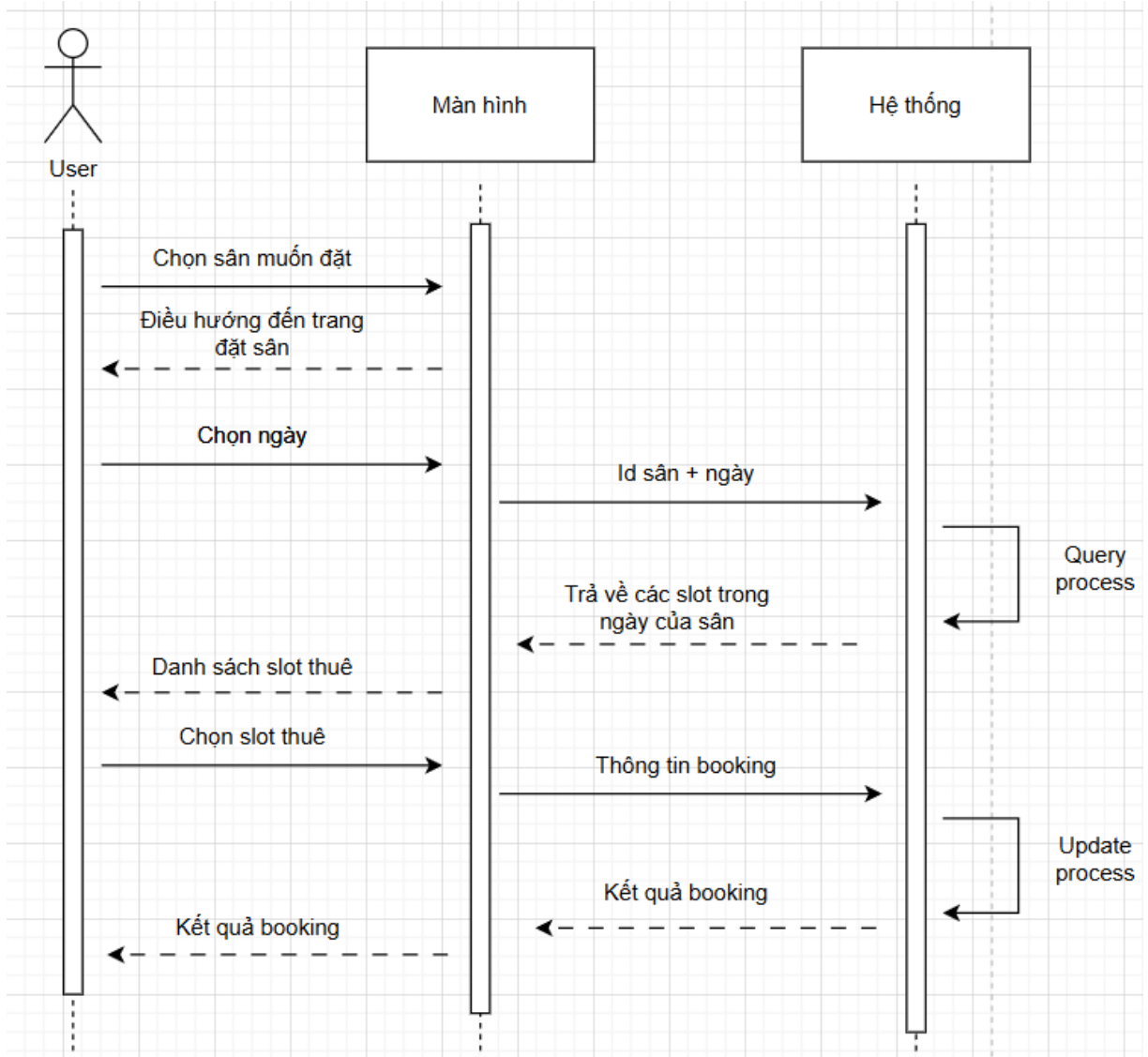
- Mô tả Use Case cập nhật thông tin cá nhân

Use Case	Đặt sân thể thao
Mục tiêu	Người dùng đặt sân thể thao cho thời gian cụ thể
Tác nhân	Người dùng đã đăng nhập
Tiền điều kiện	Người dùng đã đăng nhập và đã chọn sân muốn đặt
Kết quả đầu ra	Booking được tạo thành công và lưu vào hệ thống

Luồng sự kiện chính	1. Người dùng chọn sân từ danh sách hoặc trang chi tiết sân 2. Hệ thống hiển thị lịch để chọn ngày 3. Người dùng chọn ngày muốn đặt 4. Hệ thống hiển thị các khung giờ trong ngày với trạng thái (trống/đã đặt) 5. Người dùng chọn khung giờ trống 6. Hệ thống hiển thị tóm tắt đặt sân: sân, ngày, giờ, giá 7. Người dùng xác nhận đặt sân 8. Hệ thống lưu booking, cập nhật trạng thái khung giờ và thông báo thành công
Luồng sự kiện phụ	- Nếu người dùng chưa đăng nhập, hệ thống yêu cầu đăng nhập trước - Nếu khung giờ đã được đặt bởi người khác trước đó, hệ thống thông báo và yêu cầu chọn khung giờ khác - Người dùng có thể hủy và quay lại bất kỳ bước nào

Bảng 3-5 Mô tả Use Case đặt sân

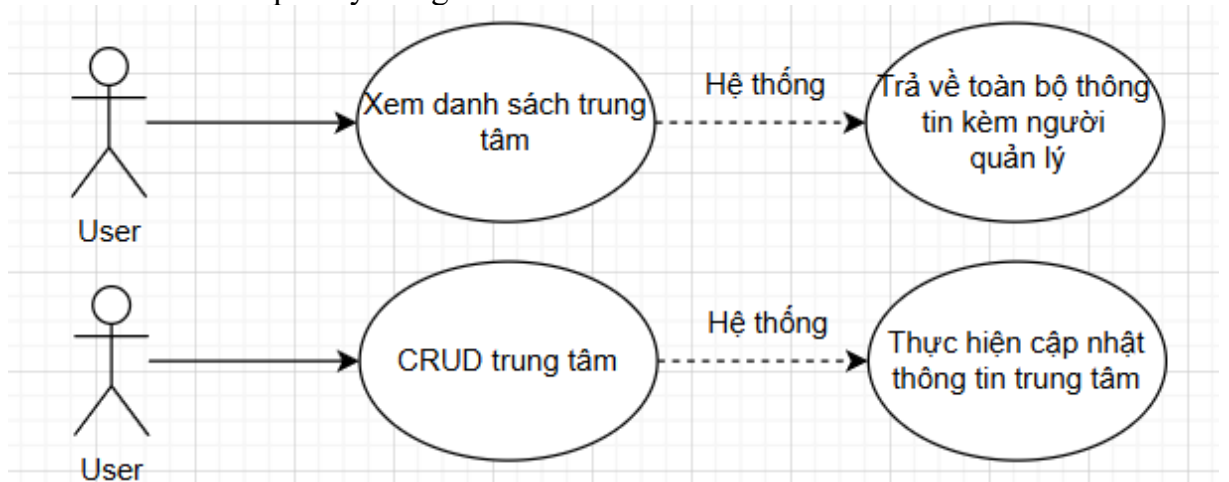
- Biểu đồ tuần tự đặt sân



Hình 3.8 Biểu đồ đồ tuần tự đặt sân

3.2.5. Chức năng quản lý trung tâm của ADMIN

- Biểu đồ Use Case quản lý trung tâm



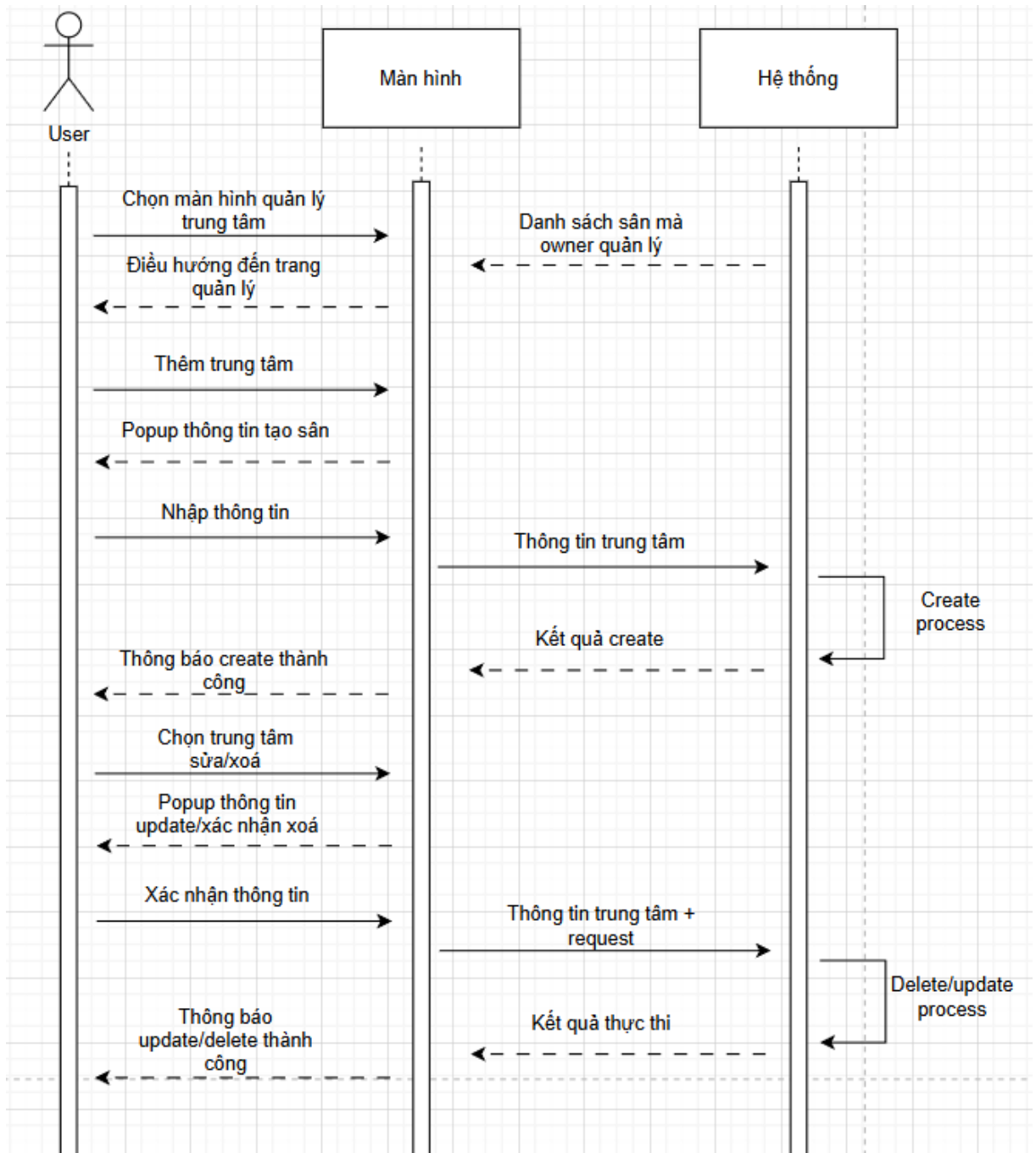
Hình 3.9 Use Case quản lý trung tâm

- Mô tả Use Case quản lý trung tâm:

Use Case	Quản lý trung tâm thể thao
Mục tiêu	Admin quản lý thông tin các trung tâm thể thao của mình
Tác nhân	Admin
Tiền điều kiện	Admin đã đăng nhập với vai trò Admin
Kết quả đầu ra	Thông tin trung tâm được thêm/sửa/xóa thành công
Luồng sự kiện chính	Thêm trung tâm: 1. Admin chọn "Thêm trung tâm mới" 2. Hệ thống hiển thị form nhập thông tin 3. Admin nhập: tên trung tâm, địa chỉ, hình ảnh 4. Hệ thống validate và lưu thông tin 5. Thông báo thêm thành công Sửa trung tâm: 1. Admin chọn trung tâm cần sửa 2. Hệ thống hiển thị form với thông tin hiện tại 3. Admin chỉnh sửa thông tin 4. Hệ thống cập nhật và thông báo Xóa trung tâm: 1. Admin chọn xóa trung tâm 2. Hệ thống kiểm tra điều kiện (không có sân hoặc booking đang hoạt động) 3. Nếu thỏa điều kiện, xóa và thông báo
Luồng sự kiện phụ	- Nếu thông tin không hợp lệ, hiển thị lỗi chi tiết - Nếu không thể xóa do còn sân/booking, thông báo lý do

Bảng 3-6 Mô tả Use Case quản lý trung tâm

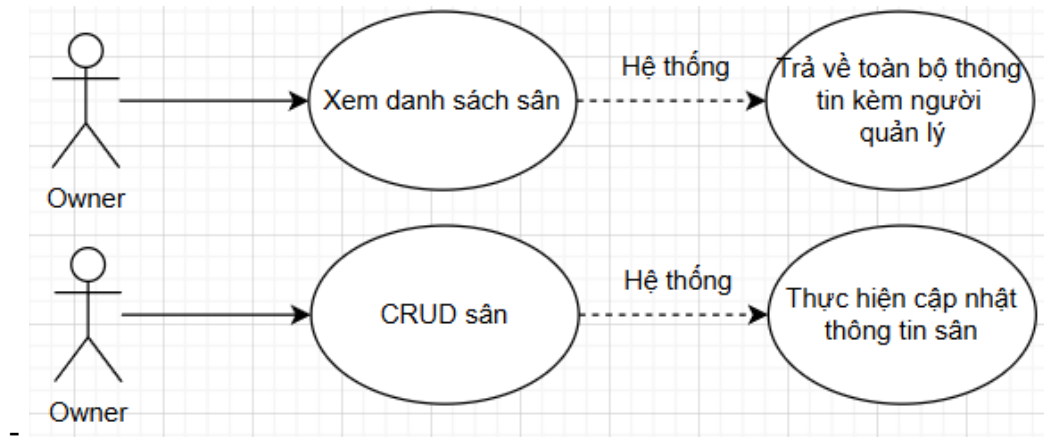
- Biểu đồ tuần tự quản lý trung tâm



Hình 3.10 Biểu đồ tuần tự quản lý trung tâm

3.2.6. Chức năng quản lý sân của chủ sân

- Biểu đồ Use Case quản lý sân của chủ sân



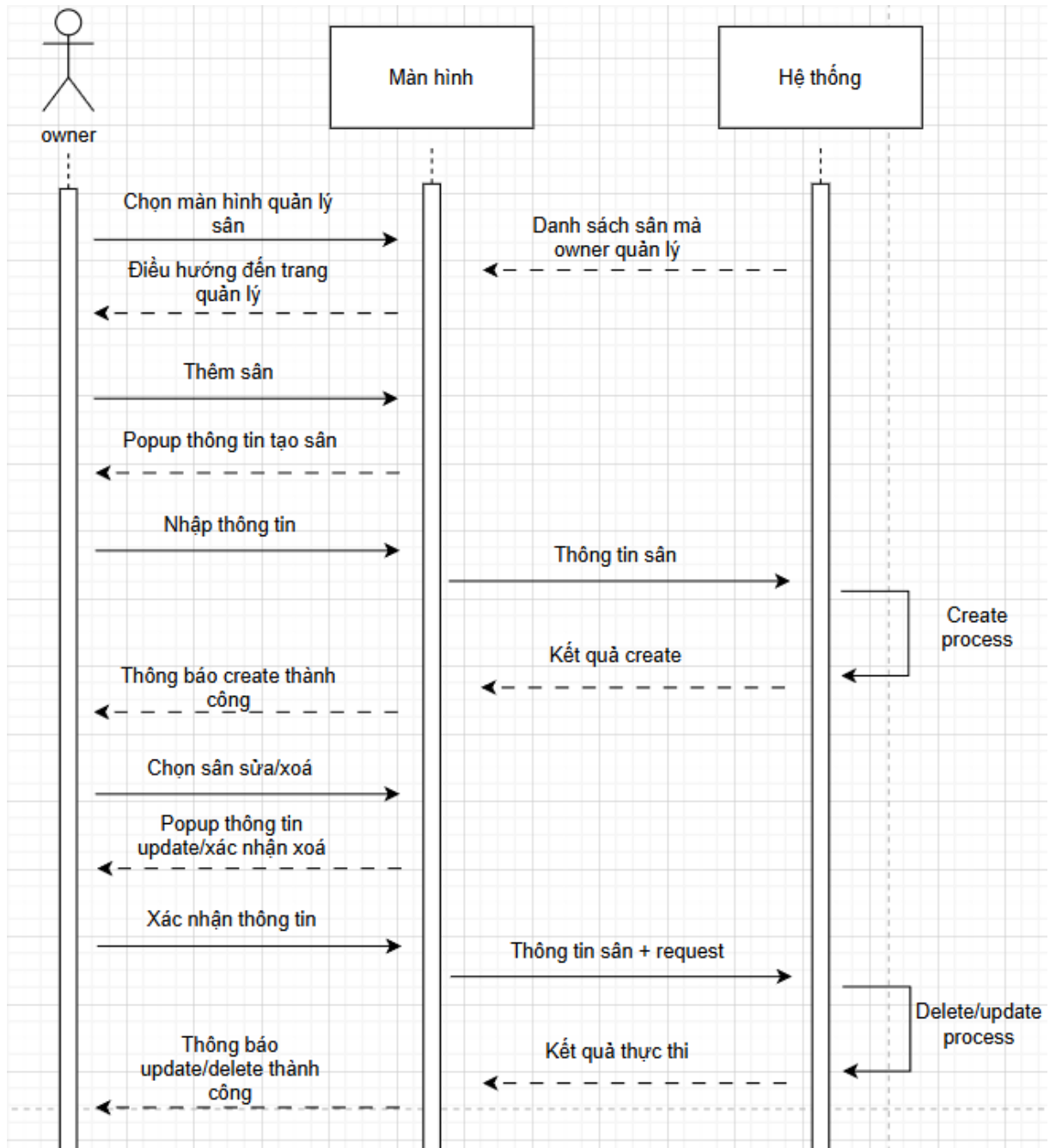
Hình 3.11 Use Case lý sân của chủ sân

- Mô tả Use Case lý sân của chủ sân

Use Case	Quản lý sân thể thao
Mục tiêu	Chủ sân quản lý các sân thuộc trung tâm của mình
Tác nhân	Chủ sân (Owner)
Tiền điều kiện	Chủ sân đã có ít nhất một trung tâm
Kết quả đầu ra	Thông tin sân được thêm/sửa/xóa thành công
Luồng sự kiện chính	Thêm sân: 1. Chủ sân chọn trung tâm và "Thêm sân mới" 2. Hệ thống hiển thị form 3. Chủ sân nhập: tên sân, loại sân, giá theo giờ, trạng thái, hình ảnh 4. Hệ thống lưu thông tin sân Sửa sân: 1. Chủ sân chọn sân cần sửa 2. Chỉnh sửa thông tin (giá, trạng thái, hình ảnh,...) 3. Hệ thống cập nhật Xóa sân: 1. Chủ sân chọn xóa sân 2. Hệ thống kiểm tra booking tương lai 3. Nếu không có booking, xóa sân
Luồng sự kiện phụ	- Validate giá phải > 0 - Không thể xóa sân có booking chưa hoàn thành

Bảng 3-7 Mô tả Use Case quản lý sân của chủ sân

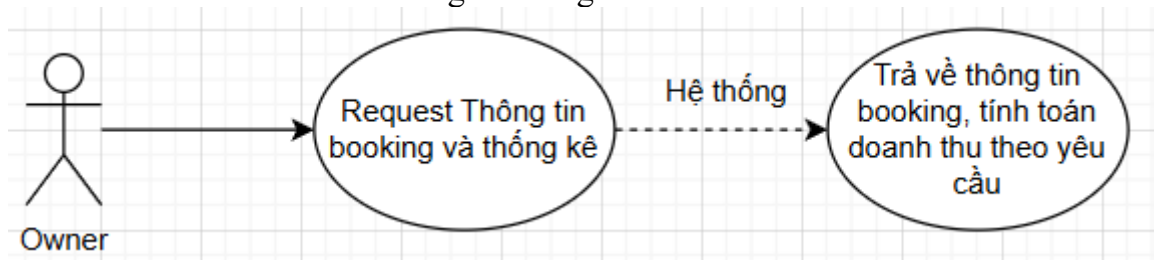
- Biểu đồ tuần tự quản lý sân của chủ sân



Hình 3.12 Biểu đồ tuần tự quản lý sân của chủ sân

3.2.7. Chức năng theo dõi booking và thống kê của chủ sân

- Biểu đồ Use Case theo dõi booking và thống kê



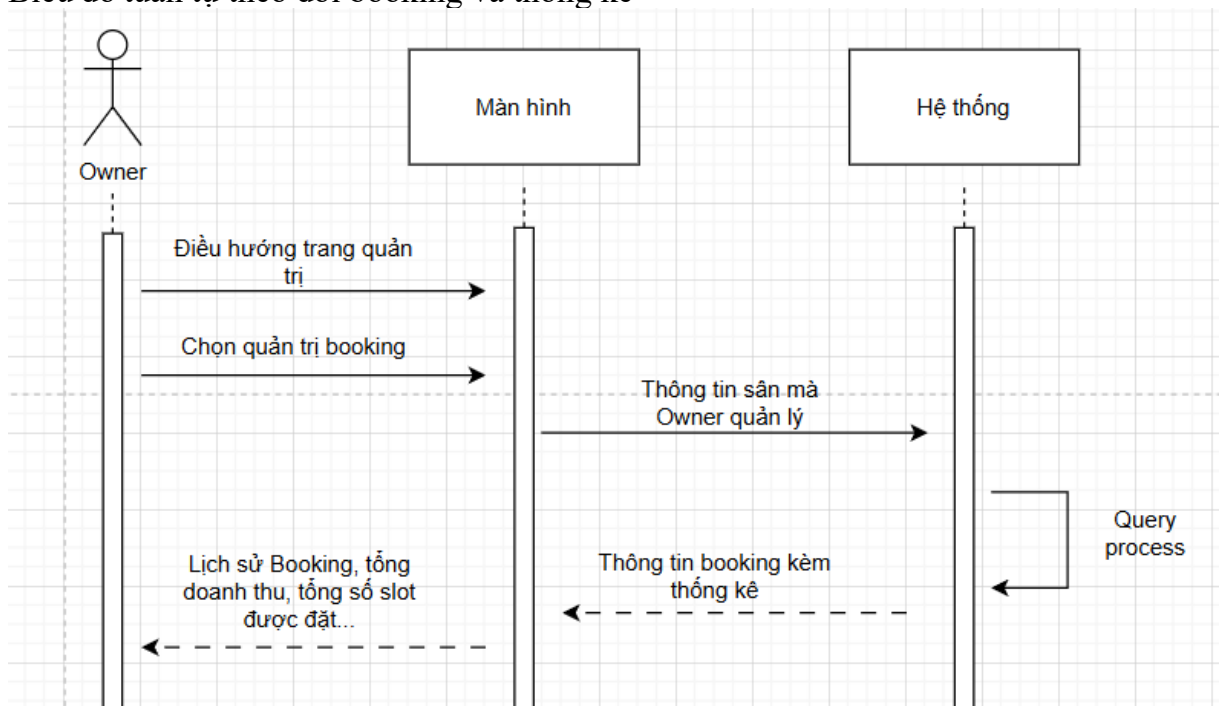
Hình 3.13 Use Case theo dõi booking và thống kê

- Mô tả Use Case theo dõi booking và thống kê

Use Case	Theo dõi booking và thống kê doanh thu
Mục tiêu	Chủ sân theo dõi các booking và xem báo cáo doanh thu
Tác nhân	Chủ sân (Owner)
Tiền điều kiện	Chủ sân đã đăng nhập
Kết quả đầu ra	Hiển thị danh sách booking và báo cáo thống kê
Luồng sự kiện chính	1. Chủ sân truy cập trang thống kê 2. Hệ thống hiển thị dashboard với: - Số lượng trung tâm, sân - Tổng booking và doanh thu
Luồng sự kiện phụ	Không thể thống kê sân có booking chưa hoàn thành

Bảng 3-8 Mô tả Use Case theo dõi booking và thống kê

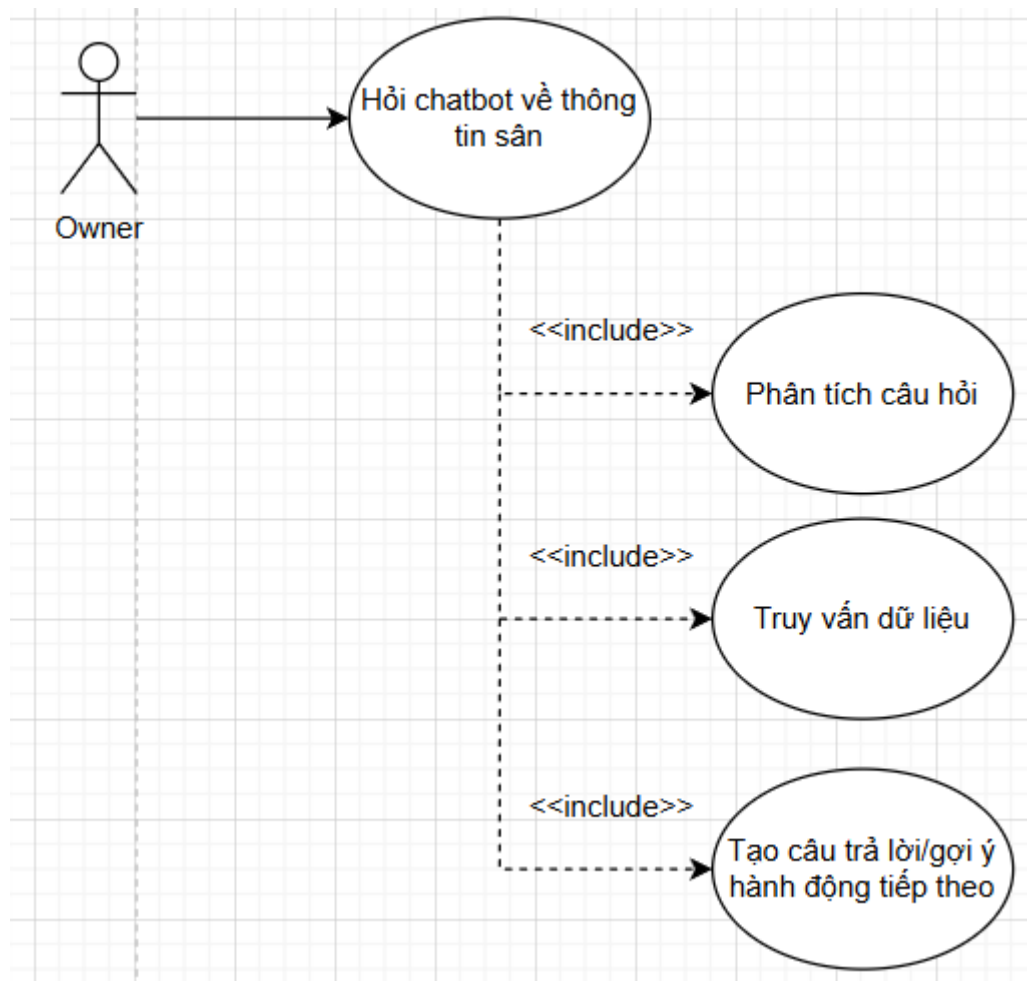
- Biểu đồ tuần tự theo dõi booking và thống kê



Hình 3.14 Biểu đồ tuần tự theo dõi booking và thống kê

3.2.8. Chức năng hỗ trợ tìm kiếm thông tin qua chatbot

- Biểu đồ Use Case hỗ trợ tìm kiếm thông tin



Hình 3.15 Use Case hỗ trợ tìm kiếm thông tin

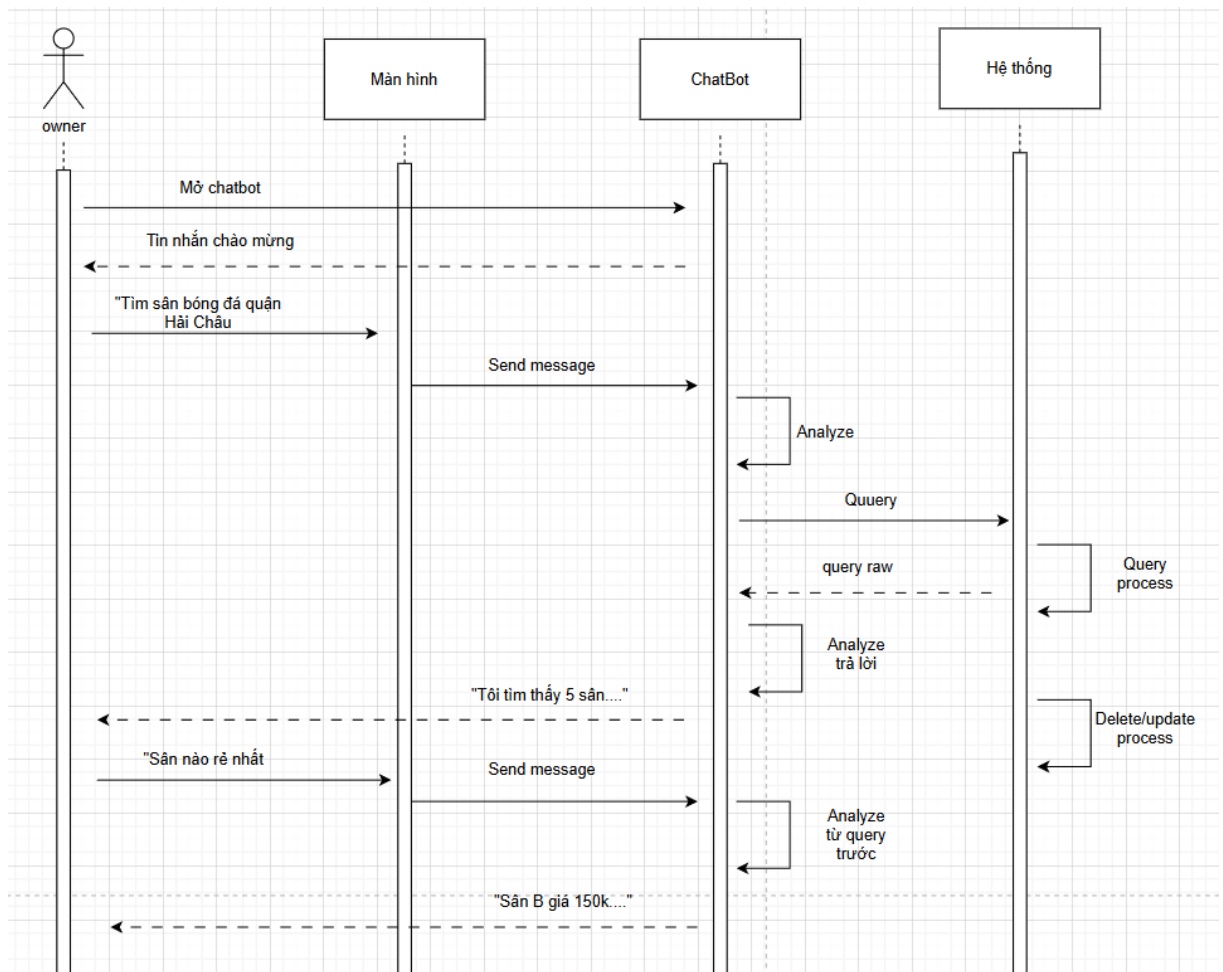
- Mô tả Use Case hỗ trợ tìm kiếm thông tin AI

Use Case	Hỗ trợ tìm kiếm thông tin qua chatbot AI
Mục tiêu	Người dùng nhận được thông tin về trung tâm, sân thể thao, giá cả, khu vực thông qua hội thoại tự nhiên với chatbot
Tác nhân	Người dùng
Tiền điều kiện	- Người dùng truy cập được hệ thống - Chatbot đã được tích hợp và hoạt động - Có kết nối với API Llama-3.3-70B-Instruct
Kết quả đầu ra	Người dùng nhận được câu trả lời chính xác, hữu ích về thông tin họ cần tìm
Luồng sự kiện chính	1. Người dùng mở cửa sổ chatbot (từ nút chatbot nổi hoặc trang chatbot toàn màn hình) 2. Hệ thống hiển thị giao diện chat với tin nhắn chào mừng từ chatbot: <i>"Xin chào! Tôi là trợ lý ảo. Tôi có thể giúp bạn tìm sân thể thao, kiểm tra giá, hoặc hướng dẫn đặt sân. Bạn cần hỗ trợ"</i>

	gì?" 3. Người dùng nhập câu hỏi bằng ngôn ngữ tự nhiên, ví dụ: - "Tìm sân bóng đá ở quận Hải Châu" - "Sân cầu lông giá rẻ gần đây" - "Trung tâm ABC có mấy sân?" - "Giá thuê sân tennis bao nhiêu?" 4. Hệ thống gửi câu hỏi đến Chatbot Service 5. Chatbot Service gọi API Llama-3.3-70B-Instruct để phân tích ý định và thực thể trong câu hỏi: - Intent (ý định): tìm sân, hỏi giá, hỏi thông tin trung tâm - Entities (thực thể): loại sân, khu vực, tên trung tâm, mức giá 6. Dựa trên kết quả phân tích, hệ thống truy vấn database để lấy dữ liệu phù hợp: - Nếu hỏi về sân → query bảng Courts với các điều kiện lọc - Nếu hỏi về trung tâm → query bảng Centers - Nếu hỏi về giá → query giá của các sân phù hợp 7. Hệ thống đưa dữ liệu vào prompt và gọi lại Llama-3.3 để tạo câu trả lời tự nhiên 8. Chatbot trả về câu trả lời dạng văn bản kèm thông tin cụ thể: <i>"Tôi tìm thấy 5 sân bóng đá ở quận Hải Châu. Dưới đây là một số gợi ý:</i> <i>1. Sân A - Trung tâm X - 200,000đ/giờ</i> <i>2. Sân B - Trung tâm Y - 180,000đ/giờ</i> <i>...</i> <i>Bạn có muốn xem chi tiết sân nào không?"</i> 9. Người dùng có thể tiếp tục hỏi để thu hẹp kết quả hoặc nhấn vào link để xem chi tiết 10. Chatbot lưu lịch sử hội thoại để duy trì ngữ cảnh
Luồng sự kiện phụ	6a. Nếu không tìm thấy kết quả phù hợp: - Chatbot trả lời: <i>"Xin lỗi, tôi không tìm thấy sân nào phù hợp với yêu cầu của bạn. Bạn có thể thử mở rộng phạm vi tìm kiếm hoặc thay đổi tiêu chí không?"</i> - Gợi ý các lựa chọn thay thế 6b. Nếu câu hỏi không rõ ràng: - Chatbot hỏi lại: <i>"Bạn có thể nói rõ hơn về loại sân bạn cần không? (bóng đá, cầu lông, tennis,...)"</i> API timeout hoặc lỗi: - Hiện thị thông báo: <i>"Xin lỗi, tôi đang gặp sự cố. Vui lòng thử lại sau hoặc liên hệ hỗ trợ."</i> - Cho phép người dùng sử dụng chức năng tìm kiếm thông thường

Bảng 3-9 Mô tả Use Case hỗ trợ tìm kiếm thông tin AI

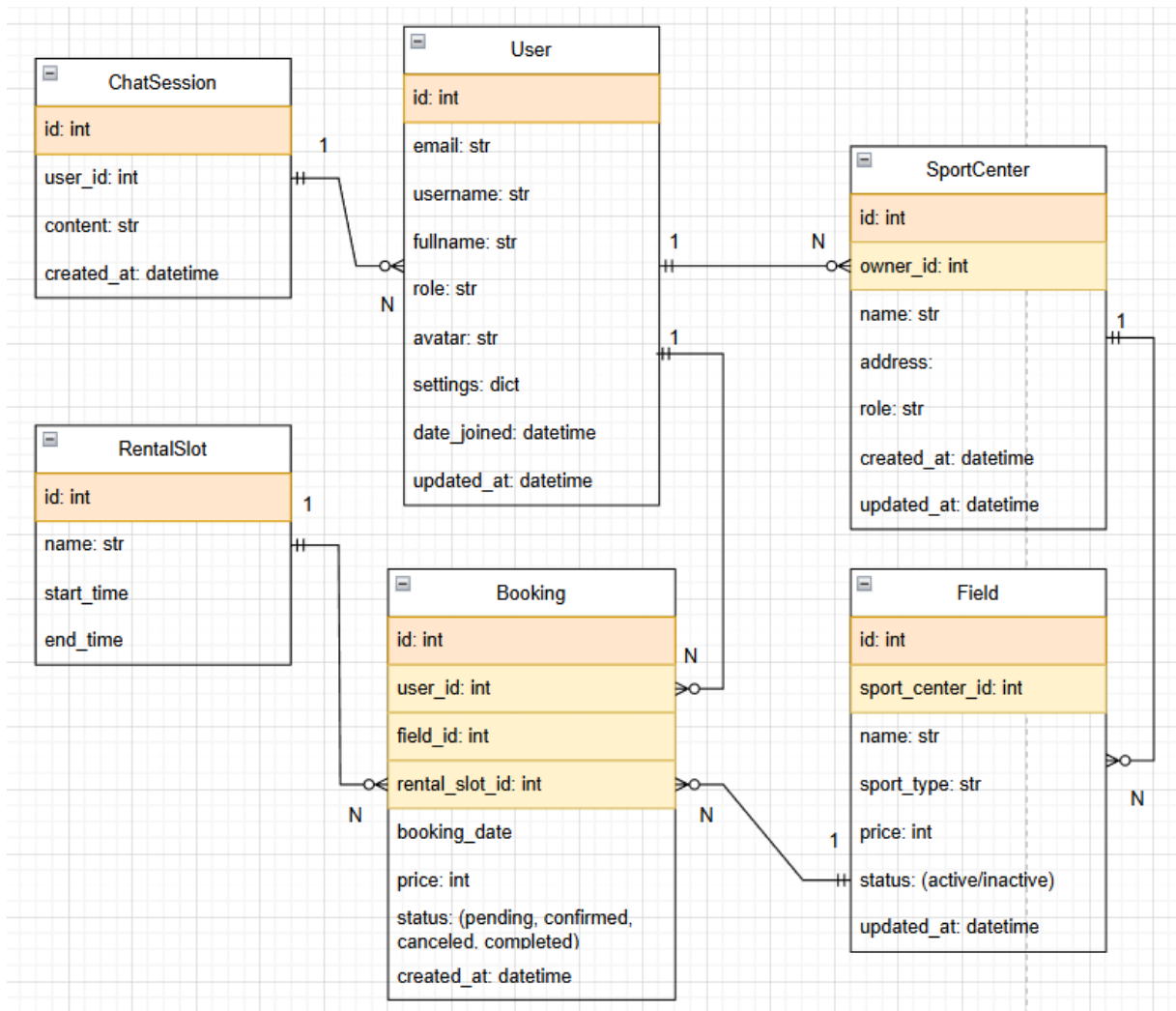
- Biểu đồ tuần tự hỗ trợ tìm kiếm thông tin AI



Hình 3.16 Biểu đồ tuần tự hỗ trợ tìm kiếm thông tin AI

3.3. Thiết kế cơ sở dữ liệu

3.3.1. Mô hình dữ liệu quan hệ



Hình 3.17 Mô hình dữ liệu quan hệ

3.3.2. Danh mục các bảng cơ sở dữ liệu

STT	Tên bảng	Mô tả
1	User	Lưu thông tin người dùng (User, Owner, Admin)
2	SportCenter	Lưu thông tin trung tâm thể thao
3	SportField	Lưu thông tin sân thể thao
4	Booking	Lưu thông tin đặt sân
5	RentalSlot	Lưu thông tin khung giờ
6	ChatSession	Lưu phiên chat với ChatBot AI

Bảng 3-20 Mô tả các bảng cơ sở dữ liệu

Tên trường	Kiểu dữ liệu	Key	Ghi chú
id	UUID	Primary Key	Mã người dùng
email	String(255)	Unique	Email đăng ký
account	String(255)	Unique	Tên đăng nhập
password	String(255)		Mật khẩu đã mã hóa
full_name	String(255)		Họ và tên
phone	String(20)		Số điện thoại
address	Text		Địa chỉ
role	Enum		USER/OWNER/ADMIN
status	Enum		ACTIVE/INACTIVE
created_at	DateTime		Ngày tạo tài khoản
updated_at	DateTime		Ngày cập nhật

Bảng 3-21 Mô tả bảng User

Tên trường	Kiểu dữ liệu	Key	Ghi chú
id	int	Primary Key	Mã trung tâm
owner_id	UUID	Foreign Key	Mã chủ sân (User)
center_name	String(255)		Tên trung tâm
address	Text		Địa chỉ chi tiết
description	Text		Mô tả trung tâm
address	String		Vị trí công việc
status	Enum		ACTIVE/INACTIVE
created_at	DateTime		Ngày tạo
updated_at	DateTime		Ngày cập nhật

Bảng 3-22 Mô tả bảng SportField

Tên trường	Kiểu dữ liệu	Key	Ghi chú
id	int	Primary Key	Mã sân
Center_id	int	Foreign Key	Mã trung tâm
name	String(255)		Tên sân
type	Enum		FOOTBALL/BASKETBALL/BADMINTON/PICK_A_BALL
status	Enum		ACTIVE/INACTIVE
price	Decimal		Giá thuê/giờ
des	Text		Mô tả sân
created_at	DateTime		Ngày tạo
update_at	DateTime		Ngày cập nhật

Bảng 3-23 Mô tả bảng SportField

Tên trường	Kiểu dữ liệu	Key	Ghi chú
id	int	Primary Key	Mã giờ thuê
name	String		Tên giờ thuê
start_time	String		Bắt đầu
end_time	String		Kết thúc thuê

Bảng 3-24 Mô tả bảng RentalSlot

Tên trường	Kiểu dữ liệu	Key	Ghi chú
id	int	Primary Key	Mã đặt sân
user_id	UUID	Foreign Key	Mã người đặt
field_id	int	Foreign Key	Mã sân
rental_id	int	Foreign Key	Mã khung giờ
booking_date	Date		Ngày đặt sân
price	Decimal		Tổng tiền
status	Enum		PENDING/CONFIRMED/COMPLETED

Bảng 3-25 Mô tả bảng SessionChat

Tên trường	Kiểu dữ liệu	Key	Ghi chú
id	String	Primary Key	Mã session chat

user_id	UUID		Mã người dùng
created_at	Datetime		Ngày tạo

Bảng 3-26 Mô tả bảng ProfessionalSkills

Thiết kế giao diện

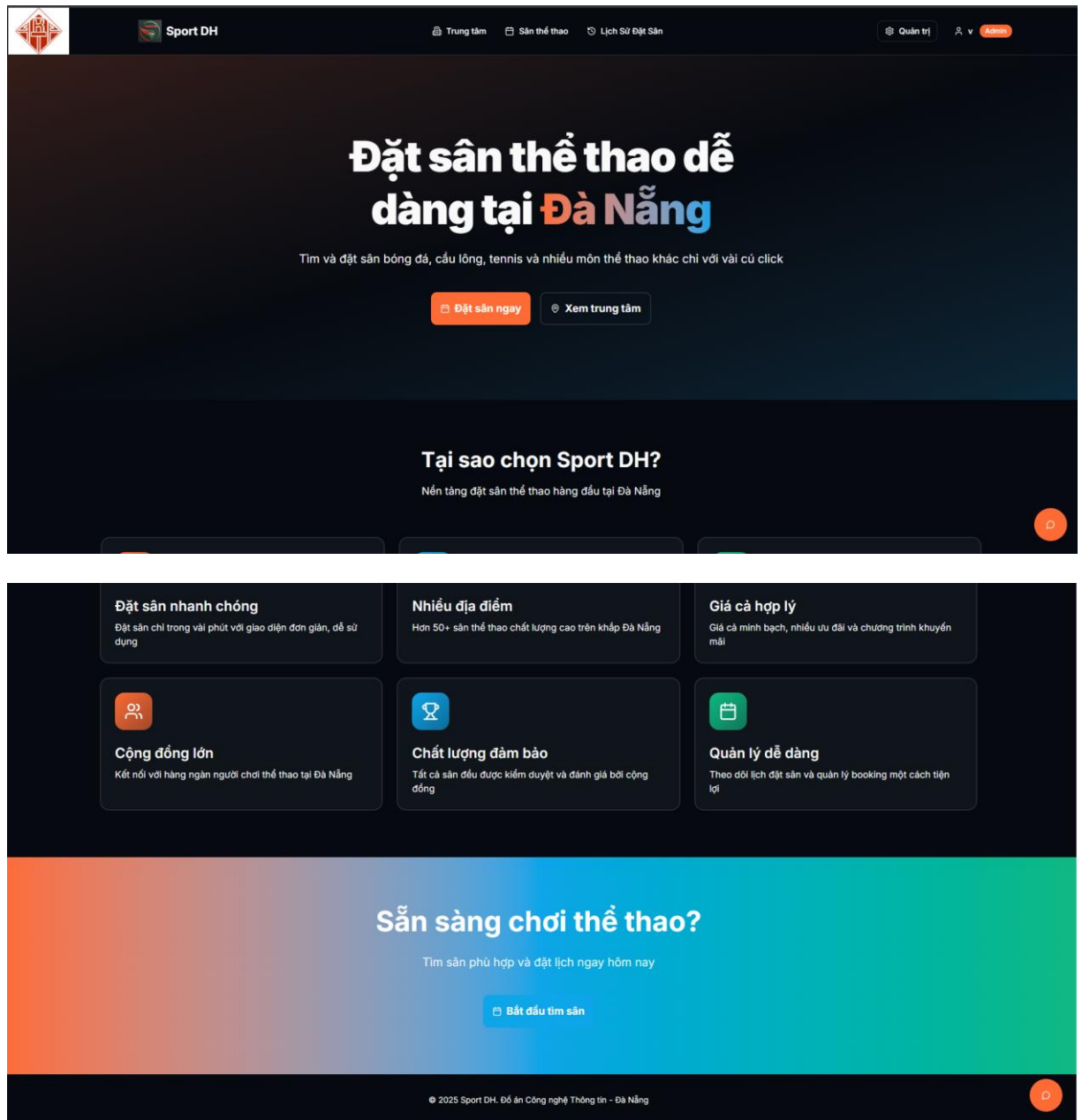
Giao diện đăng nhập

Hình 3.18 Giao diện đăng nhập

3.3.3. Giao diện đăng ký

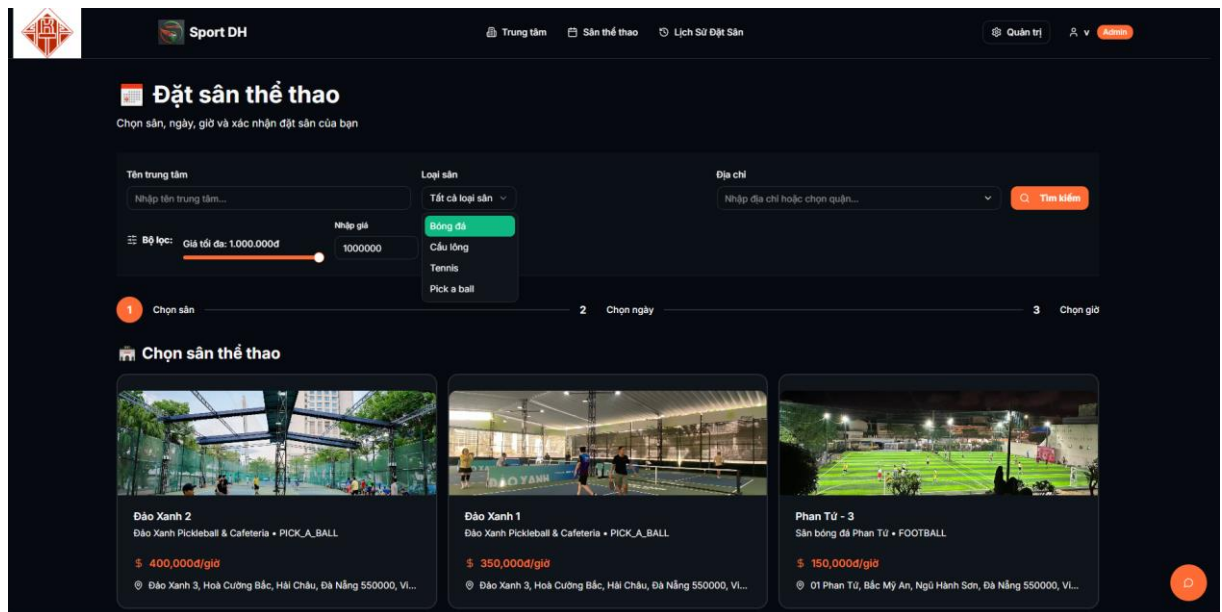
Hình 3.19 Giao diện đăng ký

3.3.4. Giao diện trang chủ



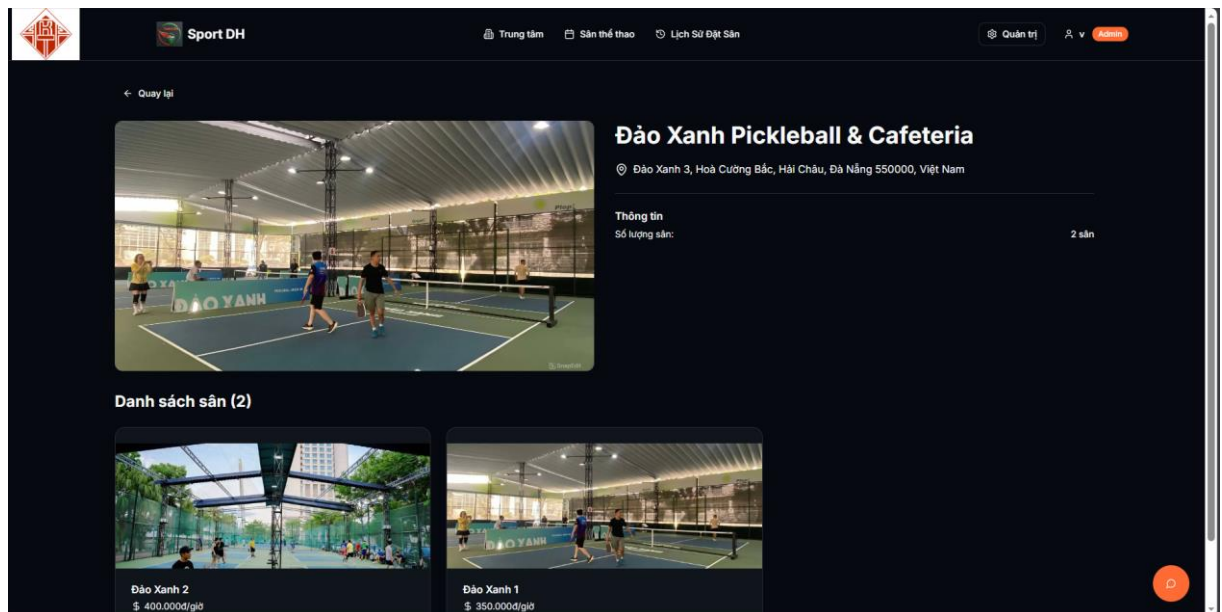
Hình 3.20 Giao diện trang chủ

3.3.5. Giao diện tìm kiếm sân



Hình 3.21 Giao diện tìm kiếm sân

3.3.6. Giao diện chi tiết sân



Hình 3.22 Giao diện chi tiết sân

3.3.7. Giao diện đặt sân

Sport DH Trung tâm Sân thể thao Lịch sử Đặt sân Quản trị Admin

Đặt sân thể thao

Chọn sân, ngày, giờ và xác nhận đặt sân của bạn

- 1 Chọn sân
- 2 Chọn ngày
- 3 Chọn giờ

Trung Vương 4
Sân bóng đá Trung Vương - BKS • FOOTBALL

Chọn ngày

Chọn ngày

CN 7 T2 8 T3 9 **Hôm nay** T4 10 T5 11 T6 12 T7 13

Chọn giờ
Thứ Ba, 9 tháng 12, 2025

SÁNG (6h-12h)

06:30 - 07:30 Trống 200,000đ	07:30 - 08:30 Trống 200,000đ	08:30 - 09:30 Trống 200,000đ	09:30 - 10:30 Trống 200,000đ
------------------------------------	------------------------------------	------------------------------------	------------------------------------

CHIỀU (12h-18h)

10:30 - 11:30 Trống 200,000đ	11:30 - 12:30 Trống 200,000đ	13:30 - 14:30 Trống 200,000đ	14:30 - 15:30 Trống 200,000đ
------------------------------------	------------------------------------	------------------------------------	------------------------------------

TỐI (18h-22h)

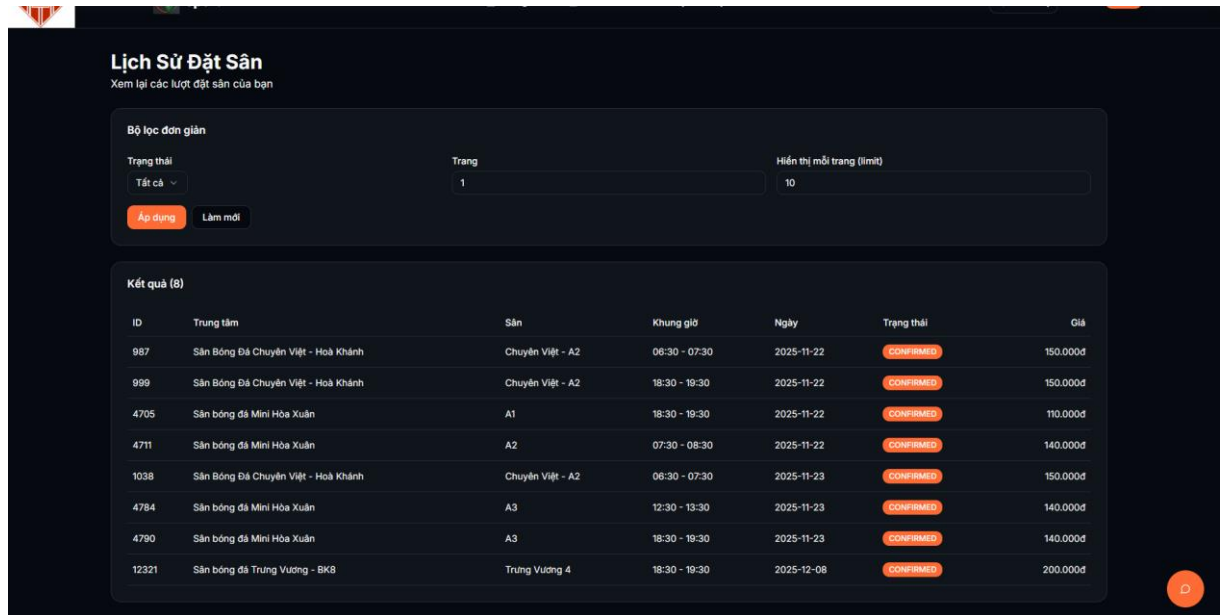
16:30 - 17:30 Trống 200,000đ	17:30 - 18:30 Trống 200,000đ	18:30 - 19:30 Trống 200,000đ	19:30 - 20:30 Trống 200,000đ
------------------------------------	------------------------------------	------------------------------------	------------------------------------

Xác nhận đặt sân
Kiểm tra thông tin trước khi xác nhận

- Sân thể thao: Sân #20
- Ngày đặt: Thứ Hai, 8 tháng 12, 2025
- Thời gian: 18:30 - 19:30
- Trạng thái: Chờ

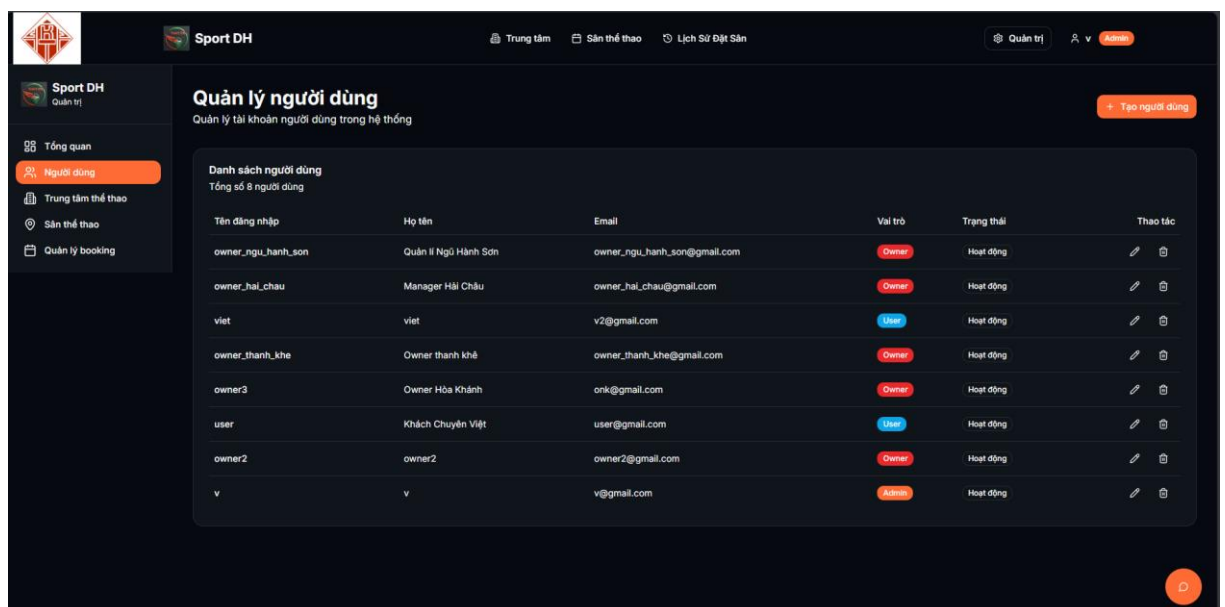
Hình 3.23 Giao diện đặt sân

3.3.8. Giao diện xem lịch sử đặt sân



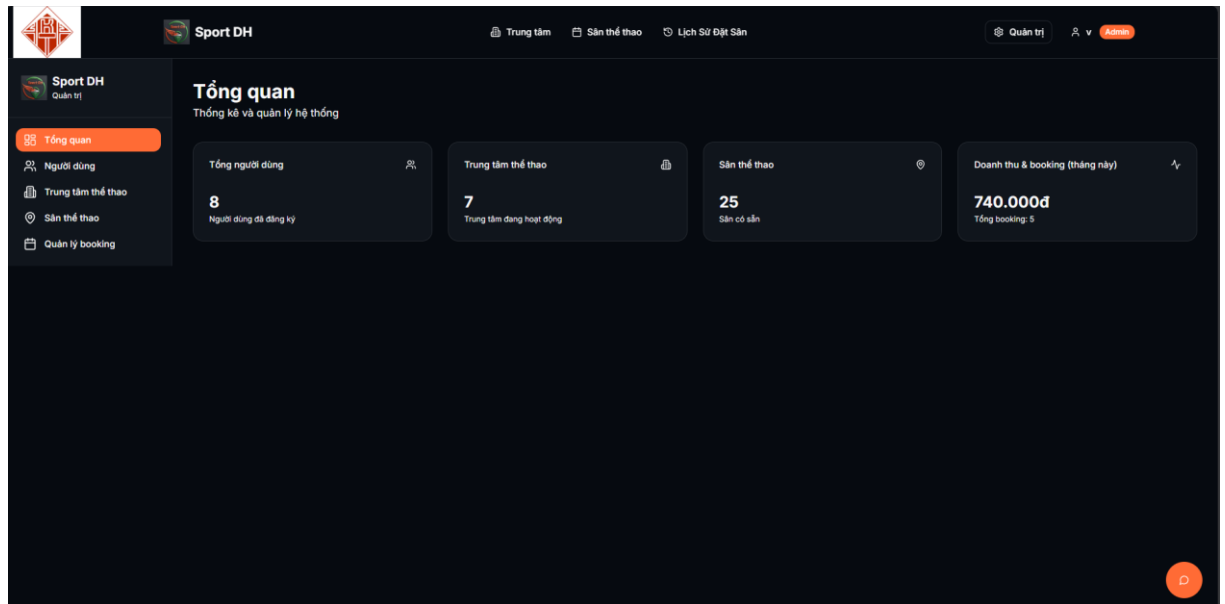
Hình 3.24 Giao diện lịch sử đặt sân

3.3.9. Giao diện quản lý người dùng



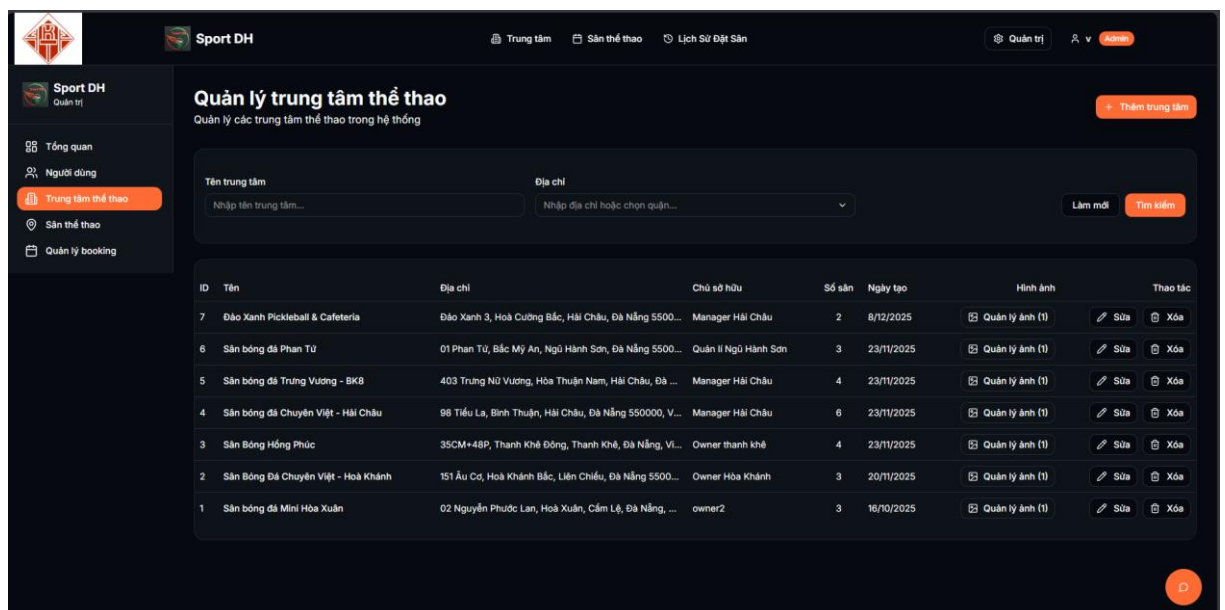
Hình 3.25 Giao diện quản lý người dùng

3.3.10. Giao diện quản trị tổng quan



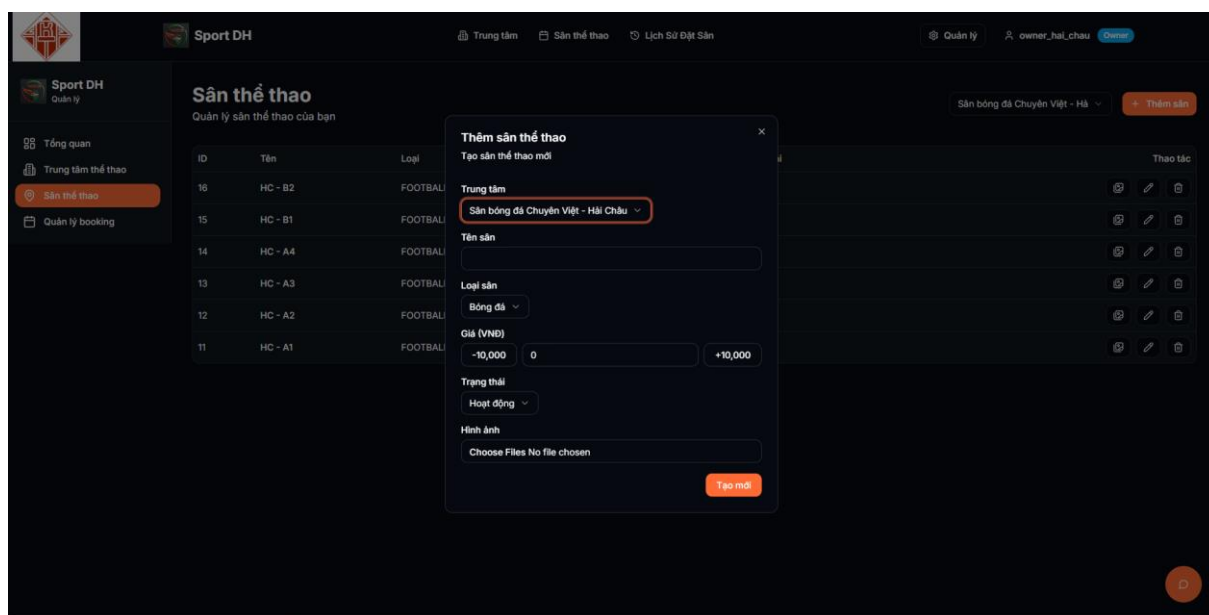
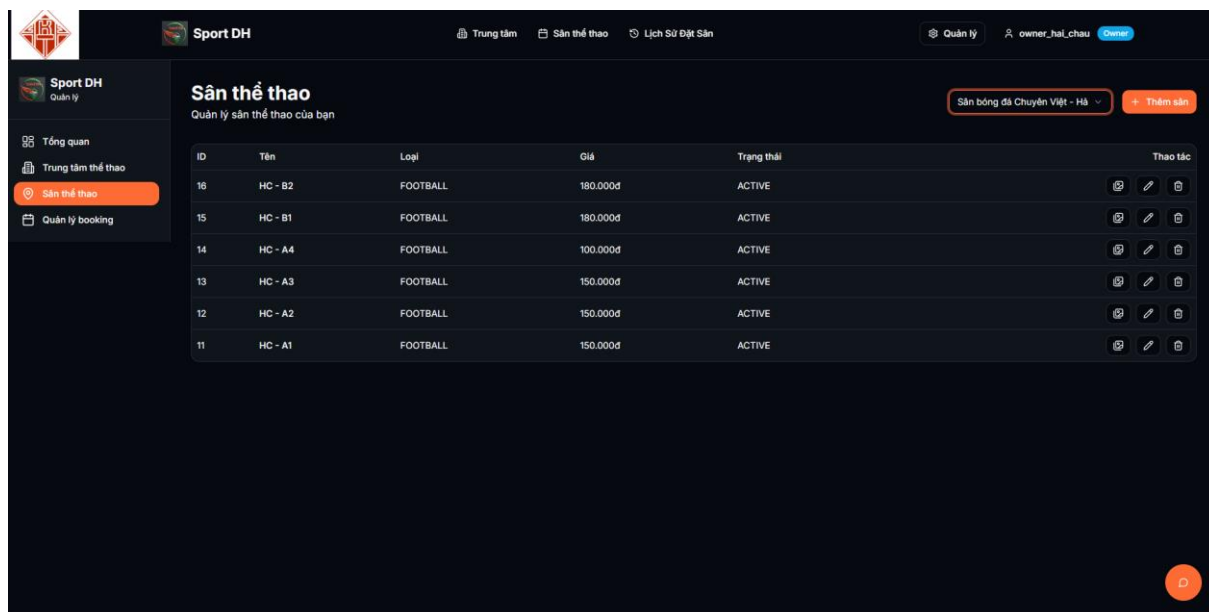
Hình 3.26 Giao diện quản trị tổng quan

3.3.11. Giao diện quản lý trung tâm (ADMIN)



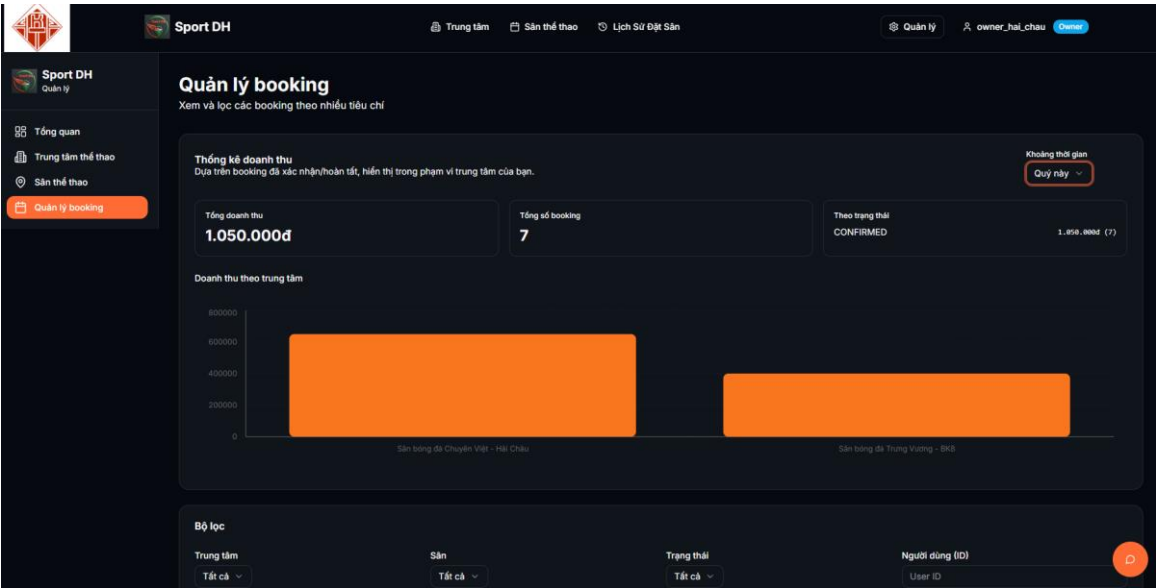
Hình 3.27 Giao diện quản lý trung tâm (ADMIN)

3.3.12. Giao diện quản lý sân (OWNER)



Hình 3.28 Giao diện quản lý sân (OWNER)

3.3.13. Giao diện thống kê booking



Bộ lọc

Trung tâm: Tất cả (v) | Sân: Tất cả (v) | Trạng thái: Tất cả (v) | Người dùng (ID): User ID

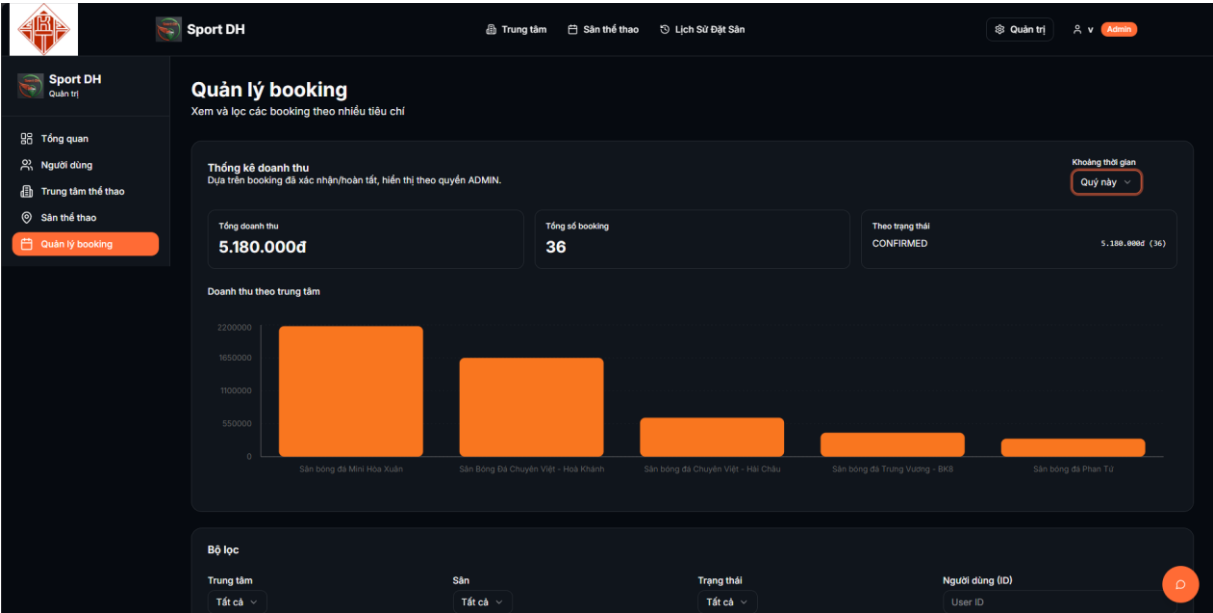
Khung giờ (ID): Rental Slot ID | Ngày sau (booking_date_after): mm/dd/yyyy | Ngày trước (booking_date_before): mm/dd/yyyy | Ngày đúng (booking_date_): mm/dd/yyyy

Tháng: VD: 12 | Năm: VD: 2025 | Trang: 1 | Hiển thị mỗi trang (limit): 10

Áp dụng **Làm mới**

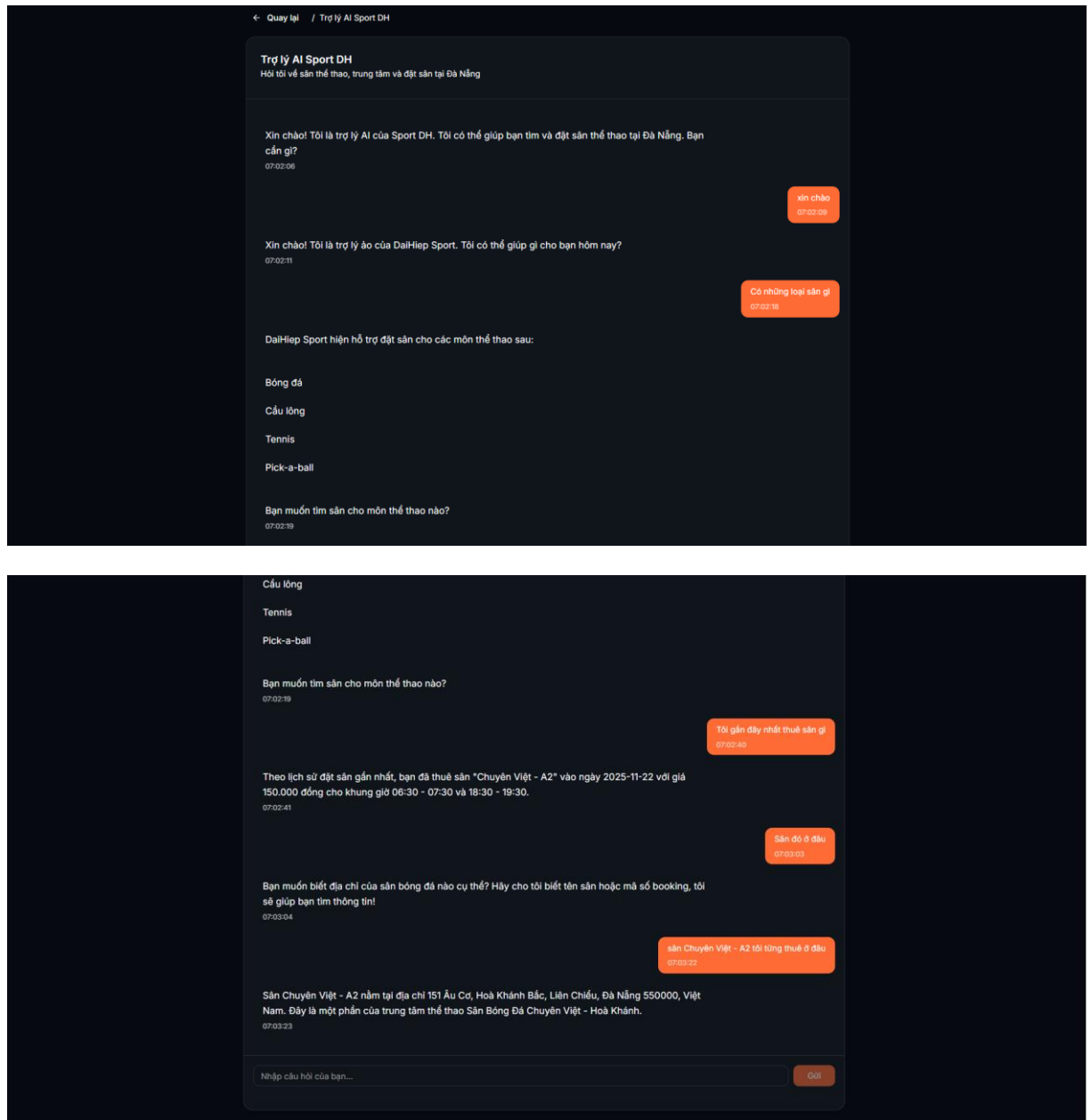
Kết quả (6086)

ID	Người dùng	Trung tâm	Sân	Khung giờ	Ngày	Trạng thái	Giá
7804	-	Sân bóng đá Chuyên Việt - Hải Châu	HC - A1	06:30 - 07:30	2025-11-23	PENDING	150.000đ
7805	-	Sân bóng đá Chuyên Việt - Hải Châu	HC - A1	07:30 - 08:30	2025-11-23	PENDING	150.000đ
7806	-	Sân bóng đá Chuyên Việt - Hải Châu	HC - A1	08:30 - 09:30	2025-11-23	PENDING	150.000đ
7807	-	Sân bóng đá Chuyên Việt - Hải Châu	HC - A1	09:30 - 10:30	2025-11-23	PENDING	150.000đ
7808	-	Sân bóng đá Chuyên Việt - Hải Châu	HC - A1	10:30 - 11:30	2025-11-23	PENDING	150.000đ
7809	-	Sân bóng đá Chuyên Việt - Hải Châu	HC - A1	11:30 - 12:30	2025-11-23	PENDING	150.000đ
7810	-	Sân bóng đá Chuyên Việt - Hải Châu	HC - A1	12:30 - 13:30	2025-11-23	PENDING	150.000đ
7811	-	Sân bóng đá Chuyên Việt - Hải Châu	HC - A1	13:30 - 14:30	2025-11-23	PENDING	150.000đ



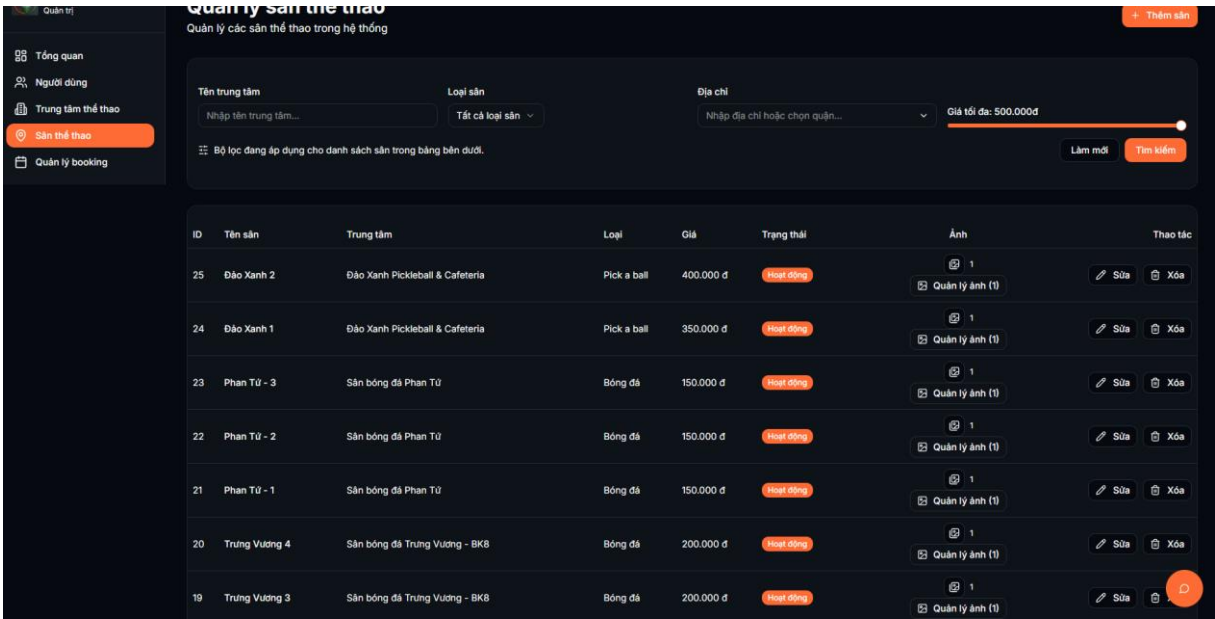
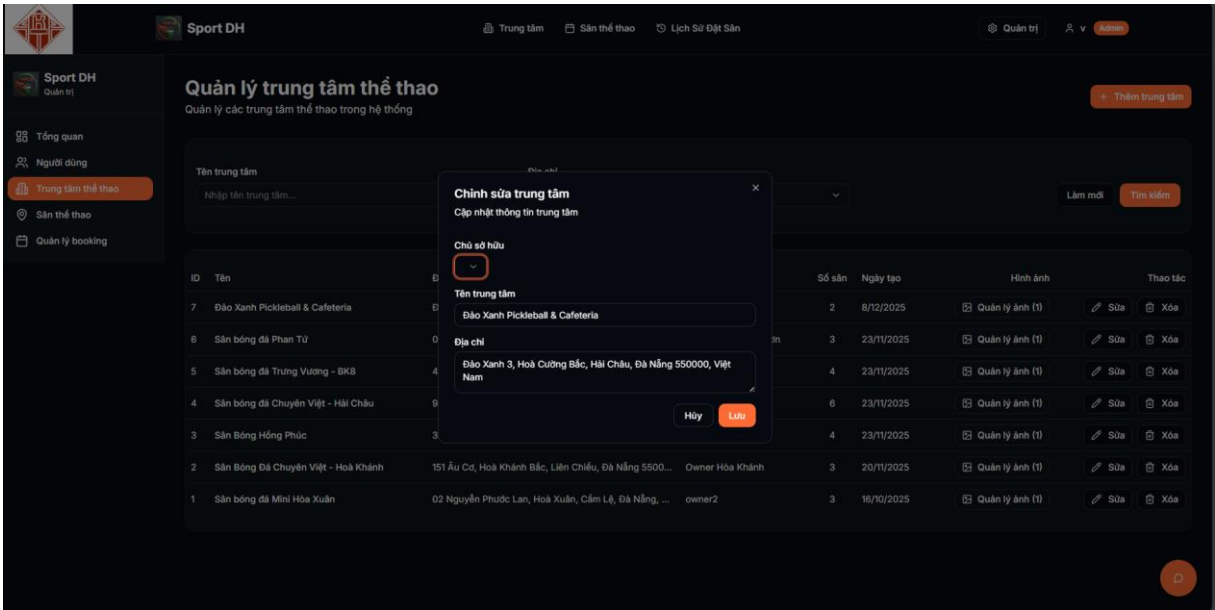
Hình 3.50 Giao diện quản lý thống kê booking

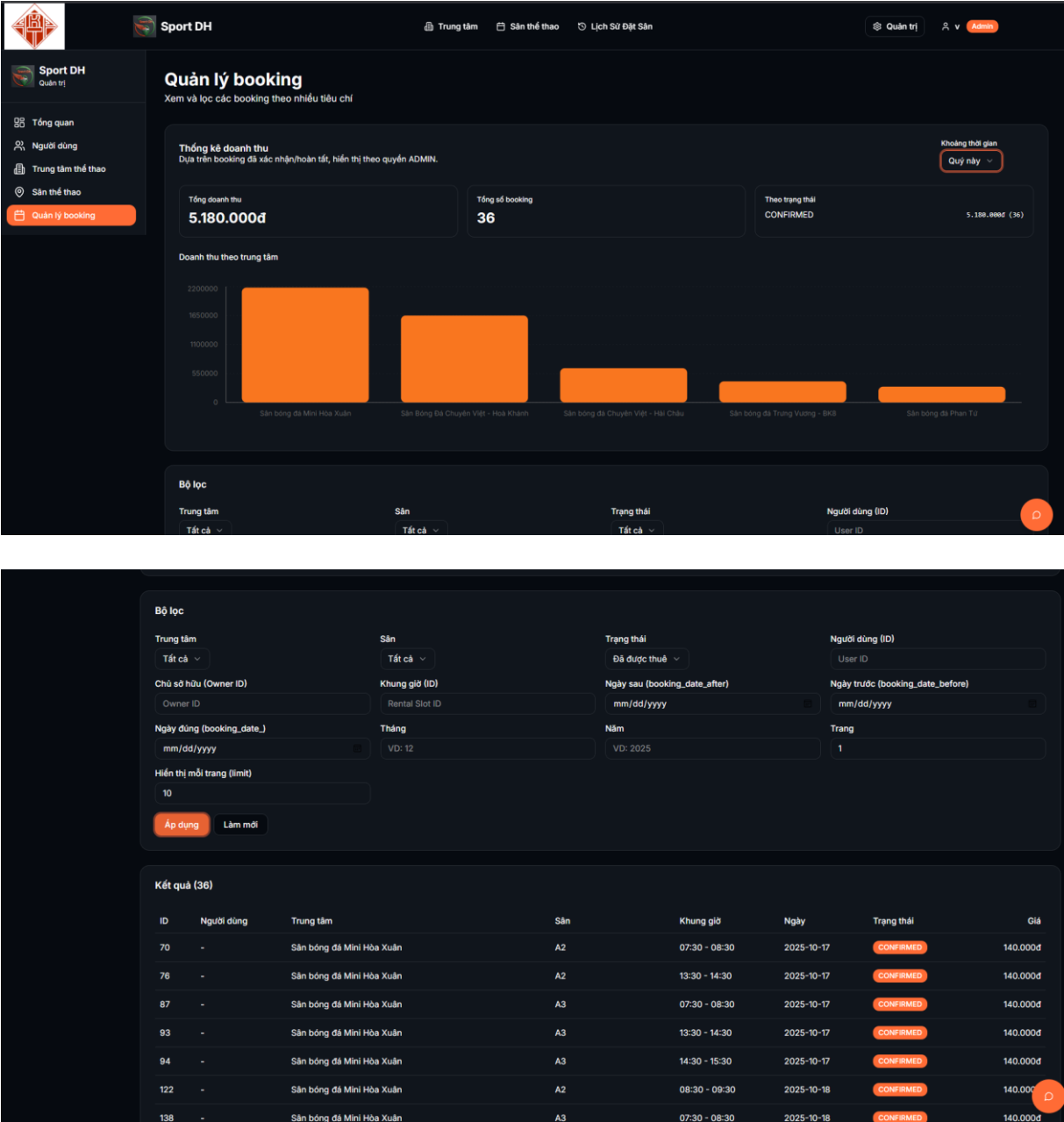
3.3.14. Giao diện ChatBot AI



Hình 3.29 Giao diện ChatBot

3.3.15. Một vài giao diện quản trị cho ADMIN





Hình 3.30 Một số giao diện cho ADMIN

Chương 4: CÀI ĐẶT VÀ GIẢI QUYẾT CÁC VẤN ĐỀ VỀ KỸ THUẬT

4.1. Tích hợp Chatbot AI với mô hình Llama

4.1.1. Vấn đề

Người dùng cần hỗ trợ nhanh chóng khi tìm kiếm sân và đặt sân. Việc tích hợp chatbot AI giúp: Trả lời câu hỏi tự nhiên về sân, giá cả, vị trí. Gợi ý sân phù hợp dựa trên nhu cầu. Hướng dẫn quy trình đặt sân. Giảm tải cho bộ phận hỗ trợ khách hàng.

4.1.2. Cách giải quyết

Tích hợp mô hình ngôn ngữ lớn Llama thông qua API:

1. Backend xử lý: Tạo endpoint nhận tin nhắn từ người dung. Kết hợp thông tin từ database với prompt gửi đến Llama. Xử lý response từ Llama và format lại . Lưu lịch sử chat (tùy chọn).
2. Context-aware responses: Cung cấp thông tin về các sân hiện có. Cho phép chatbot query database khi cần. Kết hợp dữ liệu thực tế với khả năng ngôn ngữ của AI.
3. Frontend integration Tạo UI chat thân thiện. Hiển thị typing indicator. Hỗ trợ markdown trong response. Quick replies cho câu hỏi phổ biến

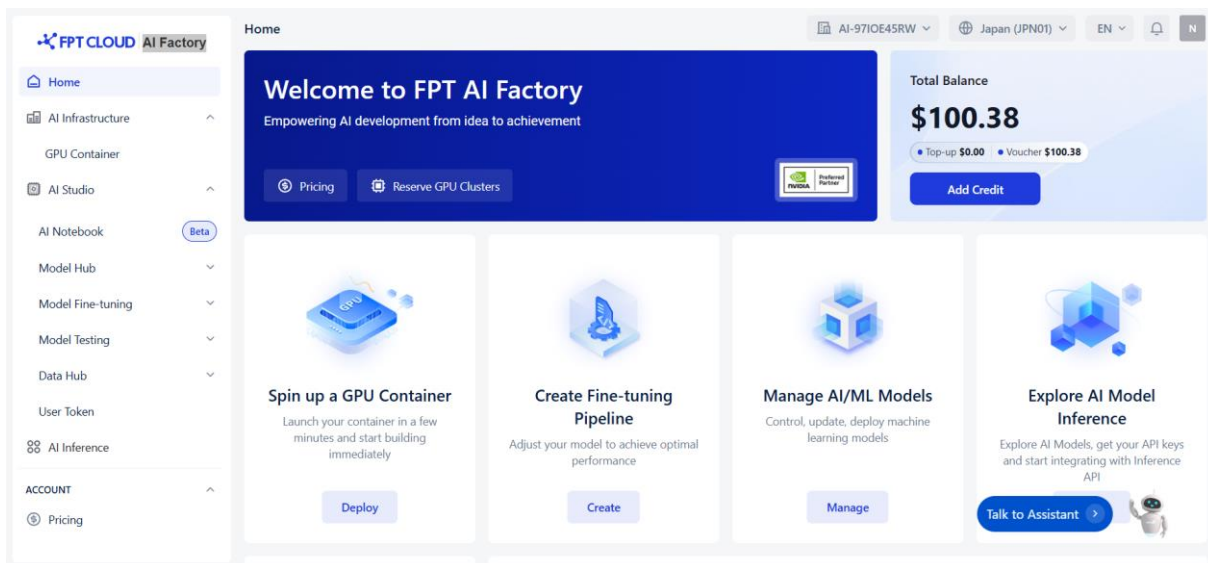
4.2. Giới thiệu nền tảng FPT AI Cloud

4.2.1. Giới thiệu

FPT AI Cloud là nền tảng điện toán đám mây do FPT phát triển, tập trung vào AI-as-a-Service. Nó cung cấp các dịch vụ trí tuệ nhân tạo sẵn dùng như:

- Nhận dạng tiếng nói (Speech-to-Text, Text-to-Speech)
- Nhận dạng ký tự (OCR)
- Chatbot và NLP tiếng Việt
- Xử lý hình ảnh, video
- Hạ tầng GPU và AI Cloud để train/inference mô hình

Mục tiêu: giúp doanh nghiệp tích hợp AI nhanh, không cần tự xây dựng hạ tầng, tối ưu chi phí và triển khai ở quy mô lớn. Nền tảng hỗ trợ tốt cho hệ thống nội bộ, chatbot, call center, tự động hóa quy trình và các ứng dụng AI doanh nghiệp khác. Hiện có chương trình tặng 100\$ credit cho người mới.



Hình 4.1 FPT AI Cloud

4.2.2. Cách sử dụng

FPT CLOUD AI Factory

Home

AI Infrastructure

GPU Container

AI Studio

AI Notebook

Model Hub

Model Fine-tuning

Model Testing

Data Hub

User Token

AI Inference

ACCOUNT

Pricing

Billing

Product Licenses

Pricing

AI-97IOE45RW

Japan (JPN01)

Total Balance: \$ 100.38

EN

N

GPU Container

AI Studio

AI Inference

USD

Serverless Endpoint

Dedicated Endpoint

Get your API Keys

Large Language Model

Model name	Context length	Max output token	Input price	Input Unit	Output price	Output Unit	View detail
DeepSeek-R1	164k	164k	\$0.649	Million tokens	\$2.739	Million tokens	
DeepSeek-V3	164k	164k	\$0.594	Million tokens	\$1.936	Million tokens	
GLM-4.5	128k	8k	\$0.66	Million tokens	\$2.2	Million tokens	
gpt-oss-120b	128K	128K	\$0.143	Million tokens			
gpt-oss-20b	128K	128K	\$0.0495	Million tokens	\$0.198	Million tokens	

The screenshot displays the FPT AI Marketplace interface, which is used for managing and utilizing AI models. The top navigation bar includes links for 'Môi trường thử nghiệm' (Test Environment), 'Tinh chỉnh mô hình' (Model Tuning), 'Triển khai mô hình' (Model Deployment), 'Bảng giá' (Pricing), and 'Hướng dẫn sử dụng' (Usage Guide). The user's account information shows 'Tiếng Việt' (Vietnamese), a balance of '\$100.38', and a 'Nạp tiền' (Recharge) button.

The main content area is divided into two panels: 'Ví dụ đầu vào' (Input Example) and 'Ví dụ đầu ra' (Output Example). The input example shows a cURL command and a JSON payload for a chat completion request. The output example shows the corresponding JSON response, including details like 'id', 'created', 'model', 'object', 'system_fingerprint', 'choices', and 'usage'.

Below the examples, the 'Thống kê sử dụng' (Usage Statistics) section is visible. It includes filters for 'API Key', 'Danh mục' (Category), 'Loại triển khai' (Deployment Type), and 'Loại sử dụng' (Usage Type). The table below shows the usage statistics for the selected API key.

STT	Tên API Key	Tên mô hình	API Key	Trạng thái	Tổng số yêu cầu	Token đầu vào	Token đầu ra	Tổng số token
1	TEST JP	Llama-3.3-70B-Instruct	sk-***PIA	Active	51	73,877	4,221	78,098
Tổng					51	73,877	4,221	78,098

The table also includes a 'Dữ liệu cập nhật lúc: 09/12/2025 07:26:35' (Data updated at: 09/12/2025 07:26:35) and a 'Tổng số bản ghi: 1' (Total number of records: 1) summary.

The bottom section of the interface features a dark blue banner with the FPT AI Marketplace logo and a list of links for 'Khám phá' (Discover) and 'Liên kết nhanh' (Quick Links). The 'Khám phá' section includes links to 'FPT Cloud' and 'AI Factory'. The 'Liên kết nhanh' section includes links to 'Chính sách bảo mật' (Privacy Policy), 'Điều khoản sử dụng' (Terms of Use), 'Tài liệu' (Documentation), 'Kiến thức' (Knowledge), and 'Hướng dẫn sử dụng' (Usage Guide).

Hình 4.2: Sử dụng credit free để sử dụng mô hình AI tân tiến của FPT AI Cloud

KẾT LUẬN

I. KẾT QUẢ ĐẠT ĐƯỢC

Về công nghệ

- Đã hiểu rõ và thành thạo cách sử dụng Django REST Framework cho backend và NextJS cho frontend, đây là những công nghệ được sử dụng rộng rãi trong phát triển ứng dụng web hiện đại.
- Biết cách thiết kế database quan hệ phức tạp, tối ưu hóa truy vấn và đảm bảo tính toàn vẹn dữ liệu thông qua các ràng buộc và transaction.
- Thành công trong việc tích hợp mô hình ngôn ngữ lớn Llama để tạo chatbot thông minh, mang lại giá trị thực tiễn cho người dùng.

Về chương trình

- Hệ thống đã được xây dựng với đầy đủ các module cho cả ba vai trò người dùng (User, Owner, Admin), đáp ứng tất cả yêu cầu ban đầu.
- Sử dụng NextJS kết hợp Tailwind CSS tạo ra giao diện responsive, đẹp mắt và dễ sử dụng trên mọi thiết bị.
- Chủ sân có đầy đủ công cụ để quản lý trung tâm, sân thể thao và theo dõi doanh thu chi tiết thông qua dashboard trực quan với biểu đồ.
- Tích hợp thành công chatbot thông minh có khả năng hiểu ngôn ngữ tự nhiên, trả lời câu hỏi và hướng dẫn người dùng một cách tự động.
- Thiết kế quy trình 3 bước đơn giản, trực quan giúp người dùng dễ dàng tìm kiếm và đặt sân chỉ trong vài phút.

Về giá trị thực tiễn

- Giải quyết vấn đề thực tế: Hệ thống giải quyết được bài toán quản lý và đặt sân thể thao tại Đà Nẵng, mang lại lợi ích cho cả người dùng, chủ sân và quản trị viên.
- Tiết kiệm thời gian: Người dùng không cần gọi điện hoặc đến trực tiếp để đặt sân, mọi thao tác đều thực hiện trực tuyến nhanh chóng.
- Tăng hiệu quả kinh doanh: Chủ sân có thể quản lý nhiều trung tâm, theo dõi doanh thu và tối ưu hóa việc khai thác sân một cách khoa học.
- Trải nghiệm người dùng tốt: Giao diện thân thiện, chatbot hỗ trợ và quy trình đặt sân đơn giản tạo trải nghiệm tích cực cho người dùng.

II. HẠN CHẾ

- Chức năng Admin chưa thực sự hoàn chỉnh.

- Chức năng AI chat chưa thực sự mạnh mẽ cho nhiều tình huống thực tế.
- Thiếu chức năng thông báo cho người dùng.

III. HƯỚNG PHÁT TRIỂN

Trong tương lai, em cam kết nỗ lực không ngừng để khắc phục những hạn chế hiện tại và mở rộng các tính năng mới phù hợp với nhu cầu của tuyển dụng hiện nay. Em xin trân trọng mọi ý kiến đóng góp từ phía giáo viên và đồng học để tích lũy thêm kinh nghiệm và kiến thức, từ đó có thể tiếp tục nghiên cứu và hoàn thiện nội dung của dự án này. Mong rằng, sự hỗ trợ và ý kiến đánh giá từ mọi người sẽ giúp em phát triển mạnh mẽ hơn.

TÀI LIỆU THAM KHẢO

- [1] Django REST Framework Documentation - <https://www.django-rest-framework.org/>
- [2] Django Official Documentation - <https://docs.djangoproject.com/>
- [3] PostgreSQL Documentation - <https://www.postgresql.org/docs/> (hoặc MySQL nếu dùng MySQL)
- [4] Next.js Official Documentation - <https://nextjs.org/docs>
- [5] React Official Documentation - <https://react.dev/>
- [6] Tailwind CSS Documentation - <https://tailwindcss.com/docs>
- [7] Llama Model Documentation - <https://llama.meta.com/docs/>
- [8] Hugging Face Transformers - <https://huggingface.co/docs/transformers/>
- [9] Redux Toolkit - <https://redux-toolkit.js.org/>
- [10] MDN Web Docs - <https://developer.mozilla.org/>