**Business Requirements Document (BRD)**

**Project name: Hệ thống quản lý phòng gym kết hợp theo dõi luyện tập và gợi ý chế độ dinh dưỡng**

**Project manager:** Đặng Hoàng Danh

**Date submitted:** 07/07/2025

**Date updated:** 10/07/2025

**2nd Date updated:** 13/07/2025

# **1. Executive Summary**

Hệ thống quản lý phòng gym là một nền tảng giúp cho các phòng gym quy mô vừa và nhỏ có thể dễ dàng quản lý hội viên, huấn luyện viên, gói tập, lịch tập, và trang thiết bị, đồng thời hỗ trợ khách hàng theo dõi luyện tập và nhận gợi ý dinh dưỡng cá nhân hóa. Hệ thống mang lại giá trị cốt lõi bằng cách đơn giản hóa quy trình quản lý, giảm thiểu sai sót khi sử dụng Excel, và nâng cao trải nghiệm khách hàng thông qua giao diện thân thiện và tính năng cá nhân hóa.

Hệ thống hướng đến các phòng gym chưa có phần mềm chuyên nghiệp, giúp họ vận hành với chi phí hợp lý. Giao diện bao gồm:

* **Admin**: Xây dựng bằng Spring Boot và Thymeleaf, cung cấp công cụ quản lý trực quan, dễ sử dụng cho quản trị viên.
* **User**: Phát triển với ReactJS, đảm bảo trải nghiệm mượt mà và tương tác cao cho khách hàng.

Hệ thống dùng **Spring Boot + REST API** để xử lý dữ liệu nhanh, dễ nâng cấp nhờ kiến trúc tách biệt từng chức năng, khả năng xử lý API hiệu quả, và cộng đồng hỗ trợ rộng lớn. **ReactJS** giúp giao diện mượt mà, dễ tùy biến theo nhu cầu người dùng. **Thymeleaf** hỗ trợ tạo giao diện quản trị nhanh, tích hợp tốt với Spring Boot. **MySQL** đảm bảo lưu trữ ổn định, chi phí thấp, dễ mở rộng khi lượng dữ liệu tăng.

# **2. Project Objectives**

## **2.1. Specific (Cụ thể)**

- Xây dựng hệ thống quản lý phòng gym hỗ trợ các chức năng như: đăng ký hội viên, phân công HLV, quản lý thiết bị, lịch tập luyện.

- Phát triển giao diện web cho phép hội viên đăng ký gói tập, ghi nhật ký bài tập và nhận chế độ dinh dưỡng phù hợp với thể trạng cá nhân.

## **2.2. Measurable (Đo lường được)**

- Cho phép ghi nhận lịch sử bài tập khi người dùng tập luyện. Trong Gym (Thể hình) có các thuật ngữ: **Name** (tên bài tập), **Set** (Số Hiệp), **Rep** (Số lần). Người dùng có thể ghi log lại các buổi tập của mình (Nếu có nhu cầu) không giới hạn số lần trên hệ thống. Nó như một bảng Excel thu nhỏ để hổ trợ việc tập luyện tốt hơn mà không cần ghi nhớ.

- Cung cấp dashboard hiển thị tổng doanh thu, số lượng hội viên đang hoạt động và số lớp học tổ chức thành công trong mỗi tuần.

- Gợi ý chế độ ăn dựa trên các chỉ số: chiều cao, cân nặng, tuổi, giới tính và mục tiêu tập luyện – đảm bảo tuân theo khuyến nghị từ nguồn tin cậy như Viện Dinh dưỡng Quốc gia hoặc tài liệu quốc tế (WHO).

## **2.3. Achievable (Khả thi)**

- Dự án sử dụng Spring Boot, Thymeleaf và ReactJS – phù hợp với năng lực kỹ thuật hiện tại, và khả năng hoàn thiện trong 10 tuần là hợp lý với quy mô tính năng.

- Các chức năng đều có phạm vi thực hiện vừa phải, không vượt quá năng lực phát triển do làm cá nhân.

## **2.4. Relevant (Phù hợp)**

- Giải quyết nhu cầu thực tế của phòng gym trong việc quản lý hội viên và hỗ trợ người dùng trong quá trình luyện tập và dinh dưỡng.

- Là một bài toán phổ biến, sát với nhu cầu thị trường – ứng dụng có tiềm năng, có thể được mở rộng và triển khai thực tế sau đồ án.

## **2.5. Time-bound (Thời hạn rõ ràng)**

- Dự án sẽ được thực hiện và hoàn thiện trong vòng 10 tuần, bao gồm các giai đoạn: phân tích yêu cầu, thiết kế, lập trình, kiểm thử và báo cáo.

- Mỗi chức năng chính sẽ được phát triển theo tiến độ cụ thể từng tuần (có chi tiết trong Project Timeline).

# **3. Background**

Tại Việt Nam, nhiều phòng gym quy mô vừa và nhỏ vẫn quản lý hoạt động bằng phương pháp thủ công như sổ sách, Excel hoặc Google Sheet. Theo KiotViet, các phương pháp này dễ dẫn đến sai sót và nhầm lẫn, gây khó khăn trong việc quản lý hội viên, huấn luyện viên, gói tập, lịch tập và trang thiết bị. Trên thị trường có một số giải pháp quản lý phòng gym như Gym Master (NextX), TimeGym, ModunSoft, SmartGYM, PosApp… Trong đó, Gym Master (phần mềm của NextX) thậm chí tự nhận đang chiếm tới 90% thị phần phần mềm quản lý fitness ở Việt Nam. Các trang web, phần mềm ra đời cho thấy xu hướng nhiều phòng tập đã tiếp cận với công nghệ thông tin hiện đại.

Tuy nhiên, các phần mềm hiện có vẫn tồn tại nhiều hạn chế, đặc biệt với phòng gym vừa và nhỏ. Theo NextX, chi phí đầu tư ban đầu và thời gian triển khai cao, cùng với giao diện phức tạp và khó tùy chỉnh, khiến nhân viên và khách hàng khó tiếp cận. Hơn nữa, hầu hết các hệ thống chỉ tập trung vào quản lý nghiệp vụ chung (thành viên, doanh thu, lịch tập), thiếu tích hợp các tính năng cá nhân hóa như gợi ý dinh dưỡng hay kế hoạch tập luyện dựa trên mục tiêu cá nhân.

Hệ thống quản lý phòng gym được đề xuất nhằm khắc phục các hạn chế này, cung cấp giải pháp chi phí thấp, dễ triển khai, với giao diện thân thiện (ReactJS cho khách hàng, Thymeleaf cho admin) và tính năng cá nhân hóa (theo dõi luyện tập, gợi ý dinh dưỡng). Sử dụng Spring Boot và MySQL, hệ thống đảm bảo tính linh hoạt, dễ bảo trì, phù hợp với nhu cầu của các phòng gym vừa và nhỏ tại Việt Nam.

**Nguồn tham khảo :**

NextX. (2024). Phần mềm quản lý Gym online cho chuỗi phòng tập gym:  
<https://nextx.vn/phan-mem-quan-ly-gym-online>

KiotViet. (2025). Chi phí mở phòng gym là bao nhiêu? Những loại chi phí cần biết khi kinh doanh phòng tập:

<https://www.kiotviet.vn/chi-phi-mo-phong-gym-la-bao-nhieu-nhung-loai-chi-phi-can-biet-khi-kinh-doanh-phong-tap>

# **4. Scope**

### **In Scope (Thực hiện trong hệ thống)**

* **Quản lý tài khoản hội viên và huấn luyện viên (HLV): tạo, sửa, xóa thông tin tài khoản.**
  + Tạo tài khoản mới cho hội viên (tên, email, mật khẩu, số điện thoại, chiều cao, cân nặng, mục tiêu tập luyện) và HLV (tên, chuyên môn, kinh nghiệm).
  + Sửa thông tin tài khoản (cập nhật thông số cơ thể cho hội viên, chỉnh sửa thông tin chuyên môn cho HLV).
  + Xóa tài khoản không hoạt động (chỉ Admin có quyền).
  + Phân quyền rõ ràng giữa Admin, HLV, và hội viên.
* **Giao diện Admin: xây dựng bằng Spring Boot và Thymeleaf, hỗ trợ quản lý toàn diện hội viên, HLV, gói tập, lịch tập, thiết bị, và check-in.**
  + Giao diện quản lý hội viên: Xem danh sách, thêm/sửa/xóa thông tin hội viên.
  + Giao diện quản lý HLV: Xem danh sách, thêm/sửa/xóa thông tin HLV.
  + Giao diện quản lý gói tập: Tạo/sửa/xóa gói tập (gói 1 tháng, 3 tháng, VIP), gán gói cho hội viên, theo dõi trạng thái (đang hoạt động, hết hạn).
  + Giao diện quản lý lịch tập: Xem/xóa lịch tập của hội viên, hỗ trợ HLV quản lý lịch.
  + Giao diện quản lý thiết bị: Thêm/sửa/xóa thông tin thiết bị (tên, loại, tình trạng, vị trí), theo dõi số lượng và bảo trì.
  + Giao diện check-in: Nhập/quét mã check-in từ hội viên, xác nhận trạng thái gói, ghi nhận thời gian check-in/out.
* **Giao diện User: dùng với ReactJS, cung cấp trải nghiệm thân thiện cho hội viên để đăng ký gói tập, ghi nhật ký bài tập, nhận gợi ý dinh dưỡng, và tạo mã check-in.**
  + Giao diện đăng ký gói tập: Xem danh sách gói, đăng ký, theo dõi thời hạn và chi phí.
  + Giao diện ghi nhật ký bài tập: Ghi thông tin (tên bài, số hiệp, số lần), xem lịch sử dưới dạng bảng, xuất file CSV/PDF.
  + Giao diện gợi ý dinh dưỡng: Nhập thông số (chiều cao, cân nặng, tuổi, giới tính, mục tiêu), nhận thực đơn cá nhân hóa dựa trên dữ liệu tham khảo (Viện Dinh dưỡng Quốc gia, WHO), tích hợp AI cơ bản để tối ưu hóa.
  + Giao diện tạo mã check-in: Tạo mã QR hoặc mã số duy nhất để sử dụng khi check-in với nhân viên.
* **Quản lý gói tập và đăng ký gói: tạo, chỉnh sửa, hủy gói tập; hỗ trợ hội viên đăng ký và theo dõi trạng thái gói.**
  + Tạo/sửa/xóa gói tập (tên, thời gian, giá).
  + Hỗ trợ hội viên đăng ký gói qua giao diện User.
  + Theo dõi trạng thái gói (đang hoạt động, hết hạn, sắp hết hạn) và gửi thông báo nhắc nhở.
* **Quản lý lịch tập: cho phép đặt, sửa, hủy, và theo dõi lịch tập của hội viên.**
  + Đặt lịch tập cho hội viên (do HLV hoặc hệ thống gợi ý).
  + Sửa/hủy lịch tập (chỉ HLV hoặc hội viên có quyền).
  + Theo dõi tiến độ hoàn thành lịch tập (đánh dấu hoàn thành).
* **Ghi nhận nhật ký bài tập: hỗ trợ hội viên ghi lại thông tin bài tập (tên bài, số hiệp, số lần) dưới dạng bảng dữ liệu đơn giản, thay thế Excel.**
  + Ghi thông tin bài tập (tên bài, số hiệp, số lần, thời gian, calo tiêu thụ).
  + Hiển thị lịch sử dưới dạng bảng dữ liệu.
  + Xuất file CSV/PDF thay thế Excel.
* **Gợi ý dinh dưỡng cá nhân hóa: cung cấp thực đơn dựa trên chiều cao, cân nặng, tuổi, giới tính, và mục tiêu tập luyện, sử dụng dữ liệu tham khảo từ Viện Dinh dưỡng Quốc gia, WHO. Có dùng AI để hoàn thiện hơn. Ví dụ Box Chat.**
  + Nhập thông số cá nhân (chiều cao, cân nặng, tuổi, giới tính, mục tiêu).
  + Tính toán nhu cầu calo và macro (protein, carb, fat) dựa trên dữ liệu Viện Dinh dưỡng Quốc gia và WHO.
  + Tích hợp AI cơ bản (quy tắc đơn giản hoặc thuật toán nhẹ) để tối ưu hóa thực đơn, hiển thị qua giao diện Box Chat (chatbot đơn giản).
  + Cho phép chỉnh sửa và lưu thực đơn cá nhân.
* **Quản lý thiết bị tập luyện: theo dõi tình trạng, số lượng, và vị trí thiết bị trong phòng gym.**
  + Thêm/sửa/xóa thông tin thiết bị (tên, loại, tình trạng, vị trí).
  + Theo dõi số lượng thiết bị sẵn có và tình trạng (hoạt động, bảo trì, hỏng).
  + Ghi nhận lịch bảo trì và chi phí bảo trì (nếu có).
* **Dashboard thống kê: hiển thị tổng doanh thu, số lượng hội viên hoạt động, số lớp học tổ chức thành công hàng tuần, và số lần check-in, dành cho Admin và HLV.**
  + Hiển thị tổng doanh thu từ gói tập và lớp học (biểu đồ cột hoặc đường).
  + Thống kê số lượng hội viên hoạt động (biểu đồ tròn).
  + Thống kê số lớp học tổ chức thành công hàng tuần (bảng hoặc biểu đồ).
  + Thống kê số lần check-in hàng ngày/tuần (biểu đồ hoặc bảng).
* **Out of Scope (Không thực hiện trong phạm vi đồ án)**
* Thanh toán online thật (VNPay, Momo…): loại bỏ do yêu cầu tích hợp phức tạp có thể vượt thời gian 10 tuần của đồ án.
* Tích hợp máy check-in vật lý: không thực hiện vì cần phần cứng chuyên dụng và chi phí phát triển cao.
* Mobile App: tập trung vào nền tảng web để đảm bảo hoàn thành trong 10 tuần.
* **Assumptions (Giả định)**
* Phòng gym có kết nối internet ổn định để truy cập hệ thống.
* Người dùng (admin, HLV, hội viên) có kỹ năng cơ bản về sử dụng web và nhập liệu.
* Dữ liệu dinh dưỡng tham khảo (từ Viện Dinh dưỡng Quốc gia hoặc WHO) được tích hợp sẵn vào cơ sở dữ liệu.
* Dữ liệu ban đầu (hội viên, thiết bị, gói tập) được tích hợp vào cơ sở dữ liệu.
* **Constraints (Ràng buộc)**
* Hệ thống chỉ hỗ trợ tiếng Việt.
* Số lượng người dùng đồng thời tối đa là 500/ngày, phù hợp với phòng gym quy mô vừa và nhỏ.
* Lưu trữ nhật ký bài tập giới hạn ở mức cơ bản (không hỗ trợ dùng AI phân tích nâng cao. Tuy nhiên PT sẽ thấy và đưa ra nhận xét).
* Hệ thống triển khai trên nền tảng web, chưa hỗ trợ ứng dụng di động.

# **5. Requirements**

## **5.1 Business Requirements**

* **Quản lý hiệu quả hoạt động phòng gym**: Tự động hóa quy trình quản lý hội viên, huấn luyện viên (HLV), gói tập, lịch tập, và thiết bị, giảm thời gian xử lý thủ công so với sử dụng Excel.
* **Tối ưu hóa trải nghiệm người dùng**: Cung cấp giao diện web thân thiện (ReactJS) để hội viên đăng ký gói tập, ghi nhật ký bài tập, và nhận gợi ý dinh dưỡng cá nhân hóa dựa trên chiều cao, cân nặng, tuổi, giới tính, và mục tiêu tập luyện.
* **Hỗ trợ quản trị bằng số liệu**: Cung cấp báo cáo doanh thu hàng tháng theo gói tập và số lượng hội viên, cùng dashboard hiển thị số liệu về hội viên hoạt động, lớp học, và tình trạng thiết bị.

## **5.2 Functional Requirements**

* **Đăng nhập/đăng ký/phân quyền**:
  + Hỗ trợ đăng nhập qua email và mật khẩu, phân quyền theo 3 vai trò: Admin (quản lý toàn bộ), HLV (quản lý lịch tập và theo dõi hội viên), User (truy cập tính năng cá nhân).
  + Trường hợp ngoại lệ: Nếu nhập sai mật khẩu 5 lần, tài khoản bị khóa tạm thời trong 15 phút.
* **CRUD người dùng, gói tập, thiết bị, HLV**:
  + Admin có thể thêm, sửa, xóa thông tin hội viên/HLV (bao gồm tên, email, số điện thoại, ngày sinh), gói tập (tên, giá, thời hạn), và thiết bị (tên, số lượng, tình trạng, vị trí).
  + Trường hợp ngoại lệ: Không cho phép xóa hội viên/HLV nếu họ đang có lịch tập hoặc gói tập hoạt động.
* **Đặt lịch tập cá nhân và nhóm**:
  + Hội viên có thể đặt, sửa, hủy lịch tập cá nhân hoặc nhóm qua giao diện user. Admin/HLV có thể quản lý lịch tập và phân công HLV.
  + Trường hợp ngoại lệ: Nếu lịch tập bị hủy, hệ thống gửi thông báo qua email cho hội viên liên quan.
* **Theo dõi macro và lịch sử tập luyện**:
  + Hội viên có thể nhập dữ liệu dinh dưỡng (calo, protein, carb, fat) và ghi nhật ký bài tập (tên bài, số hiệp, số lần), xem lịch sử trong 30 ngày qua giao diện user.
  + Gợi ý dinh dưỡng dựa trên dữ liệu nhập và tham khảo từ Viện Dinh dưỡng Quốc gia hoặc WHO.
  + Trường hợp ngoại lệ: Nếu dữ liệu nhập không hợp lệ (ví dụ: calo âm), hệ thống hiển thị thông báo lỗi.
* **Dashboard**:
  + Dashboard admin: Hiển thị doanh thu hàng tuần/tháng, số lượng hội viên mới, tỷ lệ tham gia lớp tập, và tình trạng thiết bị.
  + Dashboard HLV: Hiển thị lịch tập cá nhân/nhóm và tiến độ tập luyện của hội viên (dựa trên nhật ký bài tập).
  + Dashboard hội viên: Hiển thị lịch tập, tiến độ tập luyện, và gợi ý dinh dưỡng cá nhân hóa.

## **5.3 Non-functional Requirements**

* **Bảo mật thông tin cá nhân**:
  + Sử dụng mã hóa cho dữ liệu nhạy cảm (email, số điện thoại, ngày sinh) trong cơ sở dữ liệu.
* **Phản hồi nhanh**:
  + Thời gian phản hồi API dưới 2 giây cho 95% yêu cầu với tối đa 500 người dùng đồng thời.
* **Dễ bảo trì và mở rộng**:
  + Mã nguồn tuân theo mô hình MVC, sử dụng Spring Boot cho backend, cho phép thêm module mới (ví dụ: quản lý sự kiện phòng gym) mà không cần thay đổi cấu trúc chính.
* **Hiệu suất**:
  + Hệ thống hỗ trợ tối đa 500 người dùng đồng thời mà không bị chậm trễ (>95% yêu cầu hoàn thành trong 2 giây).
* **Khả năng tương thích**:
  + Hệ thống hoạt động trên các trình duyệt phổ biến (Chrome, Firefox, Edge)

# **6. Stakeholders**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Tên | Vai trò | Mô tả nhu cầu | Tần suất sử dụng | Mức độ ưu tiên |
| Admin | Quản trị hệ thống | Quản lý toàn bộ hoạt động phòng gym: thêm/sửa/xóa thông tin hội viên, HLV, gói tập, thiết bị; tạo và quản lý lịch tập; xem báo cáo doanh thu hàng tháng, số lượng hội viên hoạt động, tỷ lệ tham gia lớp tập, và tình trạng thiết bị qua dashboard. | Hàng ngày, 4-6 giờ  (Quản lý thường xuyên, báo cáo định kì.) | Cao |
| HLV | Nhân viên | Theo dõi và hỗ trợ hội viên: phân công lịch tập cá nhân/nhóm, ghi nhận buổi tập và xem, nhận xét tiến độ tập luyện của hội viên thông qua log. | Hàng ngày, 2-4 giờ | Trung bình |
| User | Khách hàng | Đăng ký/hủy gói tập, đặt/sửa/hủy lịch tập, ghi nhật ký bài tập (tên bài, số hiệp, số lần), nhập dữ liệu dinh dưỡng (calo, protein, carb, fat), và nhận gợi ý dinh dưỡng cá nhân hóa. | Hàng tuần, 1-2 giờ mỗi ngày | Cao |
| **Admin và User** được xếp ưu tiên cao vì đó là các đối tượng chính sử dụng hệ thống (quản lý và khách hàng).  **HLV** ưu tiên Trung bình vì vai trò hỗ trợ, phụ thuộc vào Admin và User. | | |

# **7. Timeline (10 tuần), Làm cá nhân.**

|  |  |
| --- | --- |
| **Tuần** | **Công việc** |
| 1 | |  | | --- | |  |  |  | | --- | | Viết BRD, xác định chức năng, lập kế hoạch tổng thể. | |
| 2 | Khảo sát, phân tích hệ thống hiện có. Viết SRS, vẽ sơ đồ Use Case, … (Các sơ đồ cần thiết). Thiết kế luôn DB nếu xong SRS |
| 3 | Cấu hình Code Backend (Rest API) |
| 4 | Code API (gói tập, tài chính, thiết bị) |
| 5 | Giao diện Admin (Thymeleaf) |
| 6 | Giao diện PT (Thymeleaf), bắt đầu React JS |
| 7 | Giao diện User (React: hồ sơ, log), test hệ thống |
| 8 | Gợi ý chế độ ăn/lịch tập, bắt đầu viết báo cáo |
| 9 | Kiểm thử toàn bộ, demo với GV |
| 10 | Demo và nộp đồ án |

# **8. Risks & Mitigation**

| **Rủi ro** | **Mức độ nghiêm trọng** | **Giải pháp** |
| --- | --- | --- |
| Quá tải tính năng | Cao | - Ưu tiên chức năng cốt lõi: quản lý hội viên, HLV, gói tập, lịch tập trước. - Chia dự án thành hai giai đoạn: giai đoạn 1 (chức năng cơ bản), giai đoạn 2 (dinh dưỡng, dashboard) nếu còn thời gian. - Sử dụng tài liệu BRD/SRS để giới hạn phạm vi |
| Không đủ thời gian code | Cao | - Code giao diện admin (Thymeleaf) trước sau đó là giao diện user (ReactJS). - Sử dụng các thư viện có sẵn (Axios cho ReactJS, Spring Security cho phân quyền) để giảm thời gian phát triển. - Lập kế hoạch chi tiết theo tuần và theo dõi tiến độ hàng ngày. |
| Lỗi kỹ thuật (ReactJS/Spring Boot) | Trung bình | - Chia nhỏ module (đăng nhập, CRUD, lịch tập, dinh dưỡng, dashboard)  - Tham khảo tài liệu (ReactJS, Spring Boot)  - Sử dụng công cụ debug (Postman cho API, Chrome DevTools cho frontend). |
| Thiếu kinh nghiệm với ReactJS/Spring Boot | Trung bình | - Tham khảo tài liệu chính thức trên Google và học hỏi trên Youtube - Tham gia cộng đồng (Stack Overflow, GitHub). |
| Dữ liệu người dùng bị lỗi hoặc mất mát | Cao | - Sao lưu cơ sở dữ liệu hàng ngày. |

# **9. Success Criteria (Tiêu chí đánh giá thành công)**

* Hoàn thành chức năng: Hệ thống triển khai đầy đủ 90% các chức năng trong phạm vi (In Scope): quản lý tài khoản (hội viên, HLV), gói tập, lịch tập, thiết bị, nhật ký bài tập, gợi ý dinh dưỡng, và dashboard, đúng theo tài liệu SRS (tuần 3) và hoạt động ổn định trong demo cuối kỳ (tuần 10).
* Trải nghiệm người dùng: Nhận được phản hồi tích cực từ người dùng thử nghiệm trong demo nội bộ về tính dễ sử dụng và giao diện thân thiện (ReactJS, Thymeleaf).
* Báo cáo tài chính: Dashboard của Admin hiển thị chính xác doanh thu hàng tuần/tháng, số lượng hội viên hoạt động, và tỷ lệ tham gia lớp tập.
* Bảo mật: Dữ liệu nhạy cảm (email, số điện thoại, ngày sinh) được mã hóa.

# **10. Glossary (Thuật ngữ)**

|  |  |
| --- | --- |
| **Thuật ngữ** | **Định nghĩa** |
| **Macro dinh dưỡng** | Các chất dinh dưỡng chính (calo, protein, carbohydrate, fat) cần thiết cho cơ thể, được theo dõi và gợi ý dựa trên chiều cao, cân nặng, tuổi, giới tính, và mục tiêu tập luyện của hội viên. |
| **REST API** | Giao thức lập trình ứng dụng (Application Programming Interface) sử dụng các phương thức HTTP (GET, POST, PUT, DELETE) để trao đổi dữ liệu giữa backend (Spring Boot) và frontend (ReactJS, Thymeleaf). |
| **Dashboard** | Giao diện hiển thị số liệu trực quan, cung cấp thông tin về doanh thu, số lượng hội viên hoạt động, tỷ lệ tham gia lớp tập (Admin), lịch tập và tiến độ hội viên (HLV), hoặc tiến trình tập luyện và dinh dưỡng (User). |
| **CRUD** | Viết tắt của Create, Read, Update, Delete, dùng để chỉ các thao tác quản lý dữ liệu (hội viên, HLV, gói tập, thiết bị) trong hệ thống. |
| **Thymeleaf** | Thư viện Java dùng để xây dựng giao diện admin động, tích hợp với Spring Boot để hiển thị dữ liệu quản lý. |
| **ReactJS** | Thư viện JavaScript dùng để xây dựng giao diện user thân thiện, tương tác cao, và responsive. |
| **MySQL** | Hệ quản trị cơ sở dữ liệu quan hệ, dùng để lưu trữ thông tin hội viên, HLV, gói tập, lịch tập, thiết bị, và nhật ký bài tập. |
| **MVC** | Mô hình thiết kế phần mềm (Model-View-Controller), tổ chức mã nguồn để dễ bảo trì và mở rộng, được áp dụng trong backend Spring Boot. |

# **11. Reference (Tài liệu tham khảo)**

1. Spring Boot (2024). Spring Boot Documentation: [https://spring.io/projects/spring-boot](https://spring.io/projects/spring-boot%20%20)

2. ReactJS (2024). ReactJS Official Docs: [https://reactjs.org/docs/getting-started.html](https://reactjs.org/docs/getting-started.html%20%20)

9. Postman (2024). API Testing & Documentation Tool: [https://learning.postman.com/docs/](https://learning.postman.com/docs/%20%20)

10. MySQL (2024). MySQL 8.0 Documentation. <https://dev.mysql.com/doc/>

# **12. Approval**

|  |  |
| --- | --- |
| **Người phê duyệt** | **Vai trò** |
| ThS. Võ Việt Khoa | Giảng viên Hướng Dẫn |
| Đặng Hoàng Danh | Sinh viên |