**1. Xác định tác nhân và chức năng**

Hệ thống có 3 tác nhân (Actor) chính với các chức năng tương ứng:

1. **Học viên:**
   * Đăng ký tài khoản / Đăng nhập
   * Xem danh sách khóa học
   * Đăng ký tham gia khóa học
   * Vào học (xem video, tài liệu)
   * Làm bài tập / Thi
   * Xem điểm số và tiến độ học tập
2. **Giảng viên:**
   * Đăng nhập
   * Quản lý khóa học của mình
   * Tạo/Sửa/Xóa bài học, bài giảng
   * Tạo bài tập và quản lý điểm số
   * Theo dõi danh sách học viên trong lớp
3. **Admin (Quản trị viên):**
   * Đăng nhập
   * Quản lý tài khoản (tạo, khóa, phân quyền cho Học viên, Giảng viên)
   * Quản lý toàn bộ các khóa học (gán giảng viên, duyệt khóa học)
   * Xem báo cáo, thống kê (ví dụ: doanh thu, số lượng học viên mới)

**2. Phân loại hệ thống thông tin**

Hệ thống này là sự kết hợp của nhiều loại, phục vụ các cấp độ quản lý khác nhau:

* **TPS (Transaction Processing System - Hệ thống xử lý giao dịch):**
  + **Phần nào?** Các chức năng "hàng ngày" như **Học viên đăng ký**, **Giảng viên nhập điểm**, **Học viên nộp bài**.
  + **Lý do:** Đây là các hệ thống thu thập và xử lý dữ liệu thô, cơ bản, phát sinh liên tục (giao dịch) để duy trì hoạt động của trung tâm.
* **MIS (Management Information System - Hệ thống thông tin quản lý):**
  + **Phần nào?** Chức năng của **Admin (theo dõi báo cáo, thống kê)**.
  + **Lý do:** Hệ thống này tổng hợp dữ liệu từ TPS (dữ liệu đăng ký, điểm số) để tạo ra các báo cáo (ví dụ: "Báo cáo số lượng học viên tháng 10", "Thống kê tỷ lệ hoàn thành khóa học") giúp quản lý cấp trung (Admin, chủ trung tâm) ra quyết định.

**3. Đề xuất mô hình phát triển**

* **Mô hình phù hợp nhất: Agile (Linh hoạt)**
* **Giải thích lý do:**
  1. **Yêu cầu dễ thay đổi:** Đây là một nền tảng trực tuyến, yêu cầu về giao diện (UI) và trải nghiệm người dùng (UX) rất quan trọng và thường xuyên cần thay đổi dựa trên phản hồi thực tế của học viên và giảng viên.
  2. **Phát triển lặp (Iterative):** Agile cho phép nhóm phát triển xây dựng và phát hành các tính năng cốt lõi trước (ví dụ: đăng ký và học), sau đó dần dần bổ sung các tính năng khác (như diễn đàn, game hóa...) trong các vòng lặp (sprint) tiếp theo.
  3. **Giảm rủi ro:** Thay vì đợi 6 tháng để hoàn thành toàn bộ (như mô hình Thác nước), Agile cho phép nhận phản hồi sớm và điều chỉnh kịp thời, tránh việc làm ra một sản phẩm mà người dùng không muốn sửt dụng.

**4. Ba sơ đồ UML sẽ sử dụng để thiết kế**

3 sơ đồ UML quan trọng nhất cần dùng là:

1. **Sơ đồ Use Case (Use Case Diagram):**
   * **Mục đích:** Xác định rõ ràng các tác nhân (Học viên, Giảng viên, Admin) là ai và họ có thể thực hiện những chức năng gì trên hệ thống. Đây là sơ đồ đầu tiên để xác định phạm vi của dự án.
2. **Sơ đồ Lớp (Class Diagram):**
   * **Mục đích:** Thiết kế "bộ xương" của hệ thống. Nó mô tả các lớp dữ liệu chính (ví dụ: HocVien, GiangVien, KhoaHoc, BaiHoc, DiemSo) và mối quan hệ tĩnh giữa chúng (ai sở hữu cái gì, 1-nhiều...). Đây là cơ sở để thiết kế cơ sở dữ liệu.
3. **Sơ đồ Tuần tự (Sequence Diagram):**
   * **Mục đích:** Mô tả chi tiết *luồng* hoạt động của một chức năng cụ thể. Ví dụ: mô tả chính xác các bước tương tác (thứ tự các lời gọi) giữa Học viên, Giao diện, Máy chủ (Server) và Cơ sở dữ liệu (Database) khi Học viên thực hiện chức năng "Đăng ký khóa học".