**1. Planning (Lập kế hoạch)**

Giai đoạn này tập trung vào việc xác định mục tiêu, phạm vi và tính khả thi của dự án.

* **Mục tiêu:** Xây dựng một cổng thông tin giúp người dân đăng ký tiêm chủng dễ dàng, đồng thời giúp trung tâm y tế quản lý lịch hẹn và theo dõi tình trạng tiêm chủng một cách hiệu quả, giảm tải công việc giấy tờ.
* **Phạm vi:**
  + **Bao gồm (In-scope):** Chức năng đăng ký cho người dân, chức năng xác nhận và xếp lịch của trung tâm, chức năng quản lý danh sách (đã tiêm/chưa tiêm).
  + **Không bao gồm (Out-of-scope):** Quản lý kho vaccine, thanh toán, tư vấn y tế trực tuyến.
* **Tính khả thi:**
  + **Kỹ thuật:** Công nghệ web và cơ sở dữ liệu đã sẵn có và phổ biến.
  + **Kinh tế:** Ước tính chi phí phát triển (nhân sự, máy chủ) và lợi ích (tiết kiệm thời gian, tăng hiệu quả).
  + **Pháp lý:** Phải đảm bảo tuân thủ các quy định về bảo mật thông tin y tế cá nhân.
* **Kế hoạch:** Lập kế hoạch thời gian (timeline) cho 6 giai đoạn, phân bổ nhân sự (Project Manager, Developers, Testers) và ngân sách.

**2. Analysis (Phân tích yêu cầu)**

Giai đoạn này thu thập và làm rõ các yêu cầu chi tiết của hệ thống.

* **Thu thập yêu cầu:** Phỏng vấn nhân viên trung tâm y tế và khảo sát người dân (người dùng cuối) để hiểu rõ nhu cầu.
* **Yêu cầu chức năng (Functional):**
  + **Người dân:**
    - Phải có thể tạo tài khoản/xác thực (ví dụ: qua SĐT).
    - Phải điền được biểu mẫu đăng ký (Thông tin cá nhân, tiền sử bệnh, loại vắc-xin mong muốn).
    - Phải nhận được thông báo (SMS/Email) xác nhận đã đăng ký thành công.
    - Phải nhận được lịch hẹn (ngày, giờ, địa điểm).
  + **Trung tâm y tế (Nhân viên):**
    - Phải có thể đăng nhập an toàn.
    - Phải xem được danh sách đăng ký mới.
    - Phải có chức năng "Xác nhận" hoặc "Từ chối" đăng ký (kèm lý do).
    - Phải có thể xếp lịch (chọn ngày, giờ) cho các đăng ký đã xác nhận.
    - Phải có thể cập nhật trạng thái "Đã tiêm" hoặc "Chưa tiêm" cho người dân.
    - Phải có thể lọc và tìm kiếm danh sách (theo tên, trạng thái tiêm, ngày tiêm).
* **Yêu cầu phi chức năng (Non-functional):**
  + **Bảo mật:** Dữ liệu cá nhân, y tế phải được mã hóa.
  + **Hiệu năng:** Hệ thống phải xử lý được lượng lớn người truy cập cùng lúc (ví dụ: khi mở đợt đăng ký mới).
  + **Dễ sử dụng:** Giao diện phải đơn giản, rõ ràng cho cả người dân (có thể không rành công nghệ) và nhân viên y tế.
* **Kết quả:** Tạo tài liệu đặc tả yêu cầu (SRS) và **vẽ biểu đồ Use Case** mô tả các tác nhân (Người dân, Trung tâm y tế) và các chức năng.

**3. Design (Thiết kế)**

Giai đoạn này chuyển các yêu cầu thành một bản thiết kế kỹ thuật chi tiết.

* **Thiết kế kiến trúc:** Quyết định kiến trúc tổng thể, ví dụ:
  + **Frontend:** Một website (responsive) cho người dân đăng ký.
  + **Backend:** Một API service để xử lý logic.
  + **Admin Panel:** Một trang quản trị (web portal) riêng cho trung tâm y tế.
* **Thiết kế Cơ sở dữ liệu:** **Vẽ biểu đồ Quan hệ Thực thể (ERD)**. Các bảng chính bao gồm:
  + NguoiDan (Thông tin cá nhân, SĐT, ... )
  + PhieuDangKy (Ngày đăng ký, Trạng thái, ... )
  + LichTiem (Ngày giờ hẹn, Địa điểm, Nhân viên phụ trách, ... )
  + NhanVienYTe (Tài khoản đăng nhập, ... )
* **Thiết kế Giao diện (UI/UX):** Tạo wireframe (khung sườn) và mockup (thiết kế hoàn chỉnh) cho tất cả các màn hình:
  + Trang đăng ký của người dân.
  + Trang quản lý danh sách của trung tâm y tế.
* **Thiết kế chi tiết:** **Vẽ các biểu đồ Tuần tự (Sequence Diagram)** để mô tả luồng xử lý, ví dụ: luồng "Người dân nộp đơn đăng ký" (từ trình duyệt $\to$ server $\to$ database và ngược lại).

**4. Implementation (Thi công/Lập trình)**

Đây là giai đoạn các lập trình viên viết code để xây dựng hệ thống dựa trên bản thiết kế.

* **Setup môi trường:** Chuẩn bị môi trường phát triển (development), kiểm thử (testing) và triển khai (production). Cài đặt cơ sở dữ liệu, server.
* **Xây dựng CSDL:** Tạo các bảng trong cơ sở dữ liệu dựa trên thiết kế ERD.
* **Lập trình Backend:** Viết code cho các API (logic nghiệp vụ) như: xử lý đăng ký, xác thực người dùng, xếp lịch, cập nhật trạng thái tiêm.
* **Lập trình Frontend:** Xây dựng giao diện người dùng (các trang web) dựa trên thiết kế UI/UX và kết nối chúng với API của backend.
* **Tích hợp:** Đảm bảo các module frontend và backend hoạt động trơn tru với nhau.

**5. Testing (Kiểm thử)**

Giai đoạn này đảm bảo hệ thống hoạt động đúng và không có lỗi.

* **Unit Test (Kiểm thử đơn vị):** Lập trình viên tự kiểm tra các hàm/module nhỏ mà họ viết (ví dụ: hàm kiểm tra SĐT có hợp lệ không?).
* **Integration Test (Kiểm thử tích hợp):** Kiểm tra sự phối hợp giữa các module (ví dụ: Khi người dân nhấn "Đăng ký", dữ liệu có hiển thị đúng trên trang quản trị của trung tâm y tế không?).
* **System Test (Kiểm thử hệ thống):** Kiểm tra toàn bộ hệ thống dựa trên các yêu cầu chức năng (ví dụ: thử 1 luồng đầy đủ từ A-Z: đăng ký $\to$ xác nhận $\to$ xếp lịch $\to$ cập nhật đã tiêm).
* **User Acceptance Test (UAT - Kiểm thử chấp nhận):** Giao cho nhân viên y tế và một nhóm người dân dùng thử. Họ sẽ xác nhận hệ thống có đáp ứng đúng nhu cầu và dễ sử dụng hay không.
* **Security Test:** Kiểm tra các lỗ hổng bảo mật, đặc biệt là việc bảo vệ dữ liệu nhạy cảm.

**6. Deployment & Maintenance (Triển khai & Bảo trì)**

Giai đoạn cuối cùng để đưa hệ thống vào sử dụng thực tế và hỗ trợ vận hành.

* **Deployment (Triển khai):** Cài đặt phiên bản đã qua kiểm thử lên máy chủ thật (production) để người dân và trung tâm y tế bắt đầu sử dụng.
* **Đào tạo:** Hướng dẫn và đào tạo cho các nhân viên y tế cách sử dụng trang quản trị.
* **Data Migration (Chuyển đổi dữ liệu):** Nếu trung tâm y tế đang dùng Excel hoặc sổ sách, cần có kế hoạch nhập dữ liệu cũ (nếu cần) vào hệ thống mới.
* **Maintenance (Bảo trì):**
  + **Sửa lỗi:** Theo dõi và sửa các lỗi phát sinh trong quá trình vận hành thực tế.
  + **Nâng cấp:** Cập nhật các tính năng mới dựa trên phản hồi của người dùng (ví dụ: thêm chức năng gửi tin nhắn SMS nhắc lịch tiêm).
  + **Giám sát:** Đảm bảo máy chủ hoạt động ổn định, hiệu năng tốt và an toàn.