**Kế hoạch 6 giai đoạn SDLC — Hệ thống đăng ký tiêm chủng online**

**1. Planning (Lập kế hoạch)**

* **Mục tiêu:** Xác định mục tiêu dự án, phạm vi, stakeholder, nguồn lực và kế hoạch quản lý rủi ro.
* **Việc cần làm:** thu thập stakeholder chính (trung tâm y tế, nhân viên lễ tân, bác sĩ/tiêm chủng, công dân), lập charter dự án, xác định phạm vi chức năng cốt lõi (đăng ký, xác nhận, xếp lịch, theo dõi), phân bổ đội ngũ (PO/BA/PM/devs/QA/ops), xác định yêu cầu tuân thủ dữ liệu y tế, lập kế hoạch quản lý rủi ro và phương án rollback.
* **Sản phẩm giao:** Project Charter, Stakeholder Map, Scope Statement, Risk Register, High-level roadmap, Budget & nguồn lực.
* **Người liên quan:** Trưởng trung tâm, đại diện phòng y tế, IT admin, PM, BA.
* **Tiêu chí nghiệm thu giai đoạn:** stakeholder đồng ý phạm vi và roadmap; rủi ro chính đã được ghi nhận + phương án.
* **Rủi ro & giảm thiểu:** thiếu yêu cầu quan trọng → lên lịch workshop yêu cầu; dữ liệu nhạy cảm → ghi nhận yêu cầu bảo mật ngay giai đoạn này.

**2. Requirement Analysis (Phân tích yêu cầu)**

* **Mục tiêu:** Biến mong muốn của các bên thành yêu cầu chức năng và phi chức năng rõ ràng, có thể kiểm thử.
* **Việc cần làm:** phỏng vấn chi tiết người dùng (công dân, lễ tân, cán bộ theo dõi), workshop use-case, viết user stories / yêu cầu chức năng, xác định yêu cầu phi chức năng (bảo mật, hiệu năng, SLA, backup), vẽ sơ đồ Use Case cơ bản, liệt kê tích hợp bên ngoài (cổng thanh toán, SMS, national health registry nếu có).
* **Sản phẩm giao:** SRS (Specification) hoặc Product Backlog (user stories), Acceptance Criteria cho từng chức năng, Data Dictionary, danh sách API/integ.
* **Người liên quan:** BA, PM, đại diện y tế, QA, Dev lead.
* **Tiêu chí nghiệm thu:** tất cả user stories quan trọng có Acceptance Criteria rõ; stakeholder ký duyệt SRS/Backlog.
* **Rủi ro & giảm thiểu:** yêu cầu mơ hồ → dùng prototyping / mockups để xác thực; xung đột thời gian biểu của trung tâm → ưu tiên chức năng cốt lõi.

**3. System Design (Thiết kế hệ thống)**

* **Mục tiêu:** Thiết kế kiến trúc, dữ liệu, luồng nghiệp vụ và UI để dev có thể thực hiện.
* **Việc cần làm:** chọn kiến trúc (ví dụ: client mobile/web ⇄ API gateway ⇄ microservices hoặc monolith), thiết kế ERD (các bảng: Citizen, Appointment, Center, VaccineBatch, Admin, AuditLog), xác định mô hình authentication & RBAC, thiết kế UI/UX wireframes cho các vai trò, tạo sơ đồ UML: Use Case (chi tiết), Class/ERD, Sequence (đặt lịch → xác nhận → nhắc), Deployment diagram (server, DB, SMS gateway), API spec (OpenAPI).
* **Sản phẩm giao:** Architecture Diagram, ERD, API Specs, UI wireframes/prototypes, Security design (encryption, audit, logging).
* **Người liên quan:** Solution Architect, UX/UI, Dev leads, Security officer.
* **Tiêu chí nghiệm thu:** thiết kế được duyệt, APIs có contract, mock endpoint để testing đầu.
* **Rủi ro & giảm thiểu:** thiết kế không đáp ứng tải đột biến → dự phòng autoscaling; lưu trữ PHI không an toàn → thiết kế mã hóa & logging.

**4. Implementation (Lập trình / Xây dựng)**

* **Mục tiêu:** Hiện thực các chức năng theo thiết kế, đạt tiêu chuẩn mã nguồn và tích hợp.
* **Việc cần làm:** chuẩn hoá coding standard; lập CI/CD pipeline; dev theo sprint hoặc milestone; triển khai database schema; implement authentication (2FA/OTP), tính năng đăng ký, quản lý slot, confirm/notify (SMS/email), bảng điều khiển quản trị; tích hợp với SMS/Email/payment/registry; chuẩn bị môi trường staging.
* **Sản phẩm giao:** Codebase, API servers, DB migrations, build pipeline, staging deployment, test data.
* **Người liên quan:** Dev team, DevOps, QA hỗ trợ test tự động.
* **Tiêu chí nghiệm thu:** mọi user story có Unit tests & tích hợp, code review pass, CI xanh.
* **Rủi ro & giảm thiểu:** tích hợp external thất bại → tạo mock service; lộ thông tin → kiểm tra security coding.

**5. Testing (Kiểm thử)**

* **Mục tiêu:** Đảm bảo hệ thống đáp ứng yêu cầu chức năng/phi chức năng và an toàn trước khi triển khai.
* **Việc cần làm:** viết test plan & test cases theo acceptance criteria; thực hiện: unit tests, integration tests, end-to-end tests, UAT (với đại diện trung tâm & một nhóm công dân thử), performance/load testing (điểm nghẽn xếp lịch đông), security testing (vulnerability scan, penetration test), accessibility & usability testing. Test các kịch bản đặc biệt: duplicate booking, no-show handling, change/cancel appointment, đồng bộ vaccine batch.
* **Sản phẩm giao:** Test reports, defect logs, UAT sign-off, performance report, security audit report.
* **Người liên quan:** QA, BA, đại diện y tế (UAT), Dev để fix bug.
* **Tiêu chí nghiệm thu:** UAT signed; no high-severity defects; performance within SLA; security findings remediated or accepted with mitigation.
* **Rủi ro & giảm thiểu:** lỗi nghiêm trọng gần triển khai → ưu tiên bug fix theo severity, có rollback plan.

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

* **Mục tiêu:** Triển khai an toàn xuống production, vận hành ổn định và liên tục cải tiến.
* **Việc cần làm:** triển khai theo plan (blue-green hoặc canary), cấu hình backup & disaster recovery, thiết lập monitoring/alerting (uptime, errors, queue length), lập SOP vận hành (quản lý appointment, xử lý khiếu nại), triển khai hệ thống logging & audit trail cho truy vấn PHI, tổ chức training cho nhân viên trung tâm, thu thập phản hồi người dùng, lên kế hoạch release tiếp theo và patch bảo mật định kỳ.
* **Sản phẩm giao:** Production deployment, Runbook/SOP, Backup/DR plan, Monitoring dashboard, SLA report.
* **Người liên quan:** DevOps, Ops, Support, Training team, Trung tâm y tế.
* **Tiêu chí nghiệm thu:** hệ thống hoạt động ổn định, tỷ lệ lỗi thấp, SLA được đáp ứng, người dùng cơ bản đã được đào tạo.
* **Rủi ro & giảm thiểu:** incident production → playbook & on-call; thay đổi pháp lý yêu cầu → cập nhật SRS & triển khai thay đổi nhanh.