

Session1

Bài 1:

Doanh nghiệp muốn xây dựng hệ thống quản lý hoạt động kinh doanh bao gồm:

Giao dịch bán hàng sử dụng hệ thống thông tin TPS

Phân tích xu hướng kinh doanh sử dụng hệ thống thông tin DSS

Bảng tổng quan hiệu suất hàng tháng dành cho CEO sử dụng hệ thống thông tin EIS

Bài 2:

Dự án A: Phần mềm quản lý điểm cho trường cấp 2, yêu cầu rõ ràng

Mô hình phù hợp: Waterfall

*Lý do:

- Yêu cầu đã rõ ràng và ổn định, ít thay đổi trong quá trình phát triển.
- Phù hợp với quy trình phát triển tuần tự, từng giai đoạn rõ ràng: phân tích → thiết kế → lập trình → kiểm thử → triển khai.
- Dễ quản lý, dễ ước lượng thời gian và chi phí cho một dự án nhỏ, ổn định như phần mềm trường học.

Dự án B: Ứng dụng mobile đặt lịch khám bệnh, yêu cầu linh hoạt

Mô hình phù hợp: Agile

*Lý do:

- Yêu cầu thường xuyên thay đổi do người dùng và bác sĩ có phản hồi liên tục.
- Ứng dụng mobile cần phát hành nhanh, cập nhật thường xuyên, nên Agile giúp chia nhỏ dự án thành các vòng lặp ngắn (Sprint).
- Cho phép thử nghiệm sớm, điều chỉnh linh hoạt theo phản hồi của người dùng.

Dự án C: Hệ thống ngân hàng điện tử, có bảo mật và độ phức tạp cao

Mô hình phù hợp: Spiral

*Lý do:

- Dự án phức tạp, rủi ro cao (liên quan đến bảo mật, giao dịch tài chính).
- Spiral cho phép đánh giá rủi ro ở mỗi vòng lặp, kết hợp ưu điểm của Waterfall (có giai đoạn rõ ràng) và Agile (lặp lại, cải tiến liên tục).
- Giúp phát hiện sớm lỗi hoặc rủi ro bảo mật trước khi hệ thống hoàn thiện.

Bài 3:

Con người : Gồm khách hàng (đặt món), nhà hàng/quán ăn (nhận đơn & chuẩn bị món), tài xế giao hàng (nhận và giao đơn), cùng quản trị viên hệ thống (quản lý người dùng, đơn hàng, xử lý sự cố).

Dữ liệu : Gồm thông tin người dùng (tên, địa chỉ, số điện thoại), dữ liệu nhà hàng (menu, giá, vị trí, thời gian hoạt động), đơn hàng, thanh toán, đánh giá, phản hồi, và lộ trình giao hàng.

Quy trình : Quy trình gồm: khách hàng chọn món → đặt hàng → thanh toán → nhà hàng nhận đơn & chuẩn bị món → tài xế nhận đơn → giao hàng đến khách → khách xác nhận & đánh giá.

Phần mềm : Ứng dụng di động (GrabFood, ShopeeFood...), website, hệ thống quản trị nhà hàng, phần mềm định vị GPS, hệ thống thanh toán trực tuyến, và phần mềm quản lý đơn hàng.

Phần cứng : Máy chủ lưu trữ dữ liệu (server), điện thoại di động của khách hàng và tài xế, thiết bị POS hoặc máy tính tại nhà hàng, hệ thống mạng (WiFi, 4G/5G), GPS định vị.

Bài 4:

Planning : Xác định mục tiêu của hệ thống (điểm danh sinh viên nhanh, chính xác), phạm vi dự án, nguồn lực (nhân sự, thời gian, chi phí), và lập kế hoạch tiến độ thực hiện.

Analysis : Thu thập và phân tích yêu cầu từ người dùng: giảng viên, sinh viên, và phòng đào tạo. Xác định các chức năng chính như: đăng nhập, quản lý lớp học, điểm danh bằng mã QR, thống kê buổi học.

Design : Thiết kế giao diện người dùng (UI), cơ sở dữ liệu (bảng SinhVien, LopHoc, DiemDanh...), kiến trúc hệ thống, và luồng xử lý dữ liệu. Chuẩn bị mockup hoặc prototype của ứng dụng

Implementation : Tiến hành lập trình các module của hệ thống: ứng dụng di động/web cho sinh viên và giảng viên, API kết nối với cơ sở dữ liệu, chức năng quét QR và ghi nhận điểm danh.

Testing : Thực hiện kiểm thử chức năng, kiểm thử bảo mật, kiểm thử hiệu năng. Phát hiện và sửa lỗi để đảm bảo hệ thống hoạt động ổn định, đúng yêu cầu.

Deployment & Maintenance : Triển khai ứng dụng lên máy chủ hoặc store (Google Play, App Store). Theo dõi, cập nhật, và bảo trì hệ thống khi có lỗi, thay đổi quy định, hoặc yêu cầu cải tiến từ nhà trường.

Bài 5 :

Giai đoạn Planning:

Mục tiêu:

- Xây dựng hệ thống giúp tự động hóa quá trình điểm danh, giảm sai sót, tiết kiệm thời gian, và thuận tiện cho việc quản lý dữ liệu điểm danh.

Công việc thực hiện:

- Xác định mục tiêu cụ thể: hỗ trợ điểm danh bằng QR code, thống kê nhanh, tránh gian lận.
- Xác định phạm vi dự án: bao gồm ứng dụng cho sinh viên, giao diện web hoặc app cho giảng viên, và dashboard cho phòng đào tạo.
- Xác định nguồn lực:

- Nhân sự: 1 trưởng nhóm, 2 lập trình viên backend, 2 frontend, 1 tester, 1 UI/UX designer.
- Thời gian: khoảng 3–4 tháng.
- Công nghệ: React Native hoặc Flutter (mobile), Node.js (backend), MySQL (database).
- Kế hoạch tiến độ: chia theo sprint (nếu dùng Agile), hoặc theo pha (nếu Waterfall).

Giai đoạn Requirement Analysis:

Mục tiêu:

Hiểu rõ hệ thống cần làm gì, ai sẽ sử dụng và luồng hoạt động của từng chức năng.

Công việc thực hiện:

- Xác định tác nhân (Actor):
 - Giảng viên
 - Sinh viên
 - Phòng đào tạo
 - Hệ thống quét mã QR
- Xác định các chức năng chính:
 - Giảng viên: đăng nhập, tạo lớp học, tạo buổi học, hiển thị mã QR, xem danh sách sinh viên đã điểm danh.
 - Sinh viên: đăng nhập, quét mã QR, xem lịch sử điểm danh của mình.
 - Phòng đào tạo: theo dõi, thống kê tình hình điểm danh theo lớp, theo môn, theo thời gian.

Giai đoạn System Design:

Mục tiêu:

Thiết kế kiến trúc tổng thể, cơ sở dữ liệu, giao diện, và luồng xử lý cho hệ thống.

Công việc thực hiện:

- Thiết kế kiến trúc hệ thống:
 - Ứng dụng mobile (cho sinh viên, giảng viên).
 - Web dashboard (cho phòng đào tạo).
 - Backend API quản lý dữ liệu.
 - Database server lưu thông tin người dùng, lớp học, buổi học, điểm danh.
- Thiết kế sơ đồ UML phù hợp:
 - Class Diagram: biểu diễn các lớp (SinhVien, GiangVien, LopHoc, BuoHoc, DiemDanh, QRCode) và mối quan hệ giữa chúng.
 - Sequence Diagram: mô tả luồng xử lý khi sinh viên điểm danh (Quét QR → Gửi yêu cầu → Server xác nhận → Lưu kết quả).
 - Activity Diagram: thể hiện quy trình điểm danh của giảng viên và sinh viên.
- Thiết kế cơ sở dữ liệu (ví dụ):
 - Bảng SinhVien: mã SV, họ tên, lớp, email.
 - Bảng GiangVien: mã GV, họ tên, khoa, email.
 - Bảng LopHoc, BuoHoc, DiemDanh, QRCode.

Bài 6 :

Tình huống A. Mô tả chức năng người dùng có thể thực hiện trên ứng dụng học tiếng Anh sử dụng sơ đồ Use Case Diagram

Tình huống B. Mô tả lớp `NguoIDung`, `KhoaHoc`, `BaiHoc` và quan hệ giữa chúng sử dụng sơ đồ Class Diagram

Tình huống C. Mô tả luồng học viên bắt đầu → vào học → làm bài → hoàn thành sử dụng sơ đồ Activity Diagram

Tình huống D. Mô tả cách hệ thống triển khai trên các máy chủ, thiết bị sử dụng sơ đồ Deployment Diagram

Tình huống E. Mô tả thứ tự tương tác giữa học viên và hệ thống khi nộp bài sử dụng sơ đồ Sequence Diagram

Bài 7 :

1. Planning (Lập kế hoạch)

- Mục tiêu: Xác định phạm vi, mục tiêu và tính khả thi của dự án.
Việc thực hiện:
- Làm việc với trung tâm y tế để hiểu rõ nhu cầu: quản lý đăng ký tiêm, xác nhận, theo dõi.
- Xác định người dùng chính: người dân, nhân viên y tế, quản trị viên.
- Phân tích nguồn lực (thời gian, nhân lực, chi phí, công nghệ).
- Lập kế hoạch dự án: tiến độ, giai đoạn, người phụ trách, rủi ro.

2. Requirement Analysis (Phân tích yêu cầu)

- Mục tiêu: Thu thập và xác định rõ ràng các yêu cầu của hệ thống.
Việc thực hiện:
- Phỏng vấn trung tâm y tế, người dân để hiểu quy trình hiện tại.
- Xác định yêu cầu chức năng:
- Người dân: đăng ký thông tin cá nhân, chọn loại vaccine, chọn ngày tiêm.
- Trung tâm: duyệt và xác nhận lịch, cập nhật trạng thái đã/ chưa tiêm.
- Hệ thống: gửi thông báo xác nhận, thống kê báo cáo.

- Xác định yêu cầu phi chức năng: bảo mật thông tin cá nhân, tốc độ xử lý, khả năng mở rộng.

3. System Design (Thiết kế hệ thống)

- Mục tiêu: Thiết kế cấu trúc tổng thể và chi tiết của hệ thống.
Việc thực hiện:
- Thiết kế kiến trúc hệ thống (client-server, cơ sở dữ liệu, API).
- Thiết kế giao diện người dùng: trang đăng ký, quản lý lịch tiêm, thống kê.
- Thiết kế cơ sở dữ liệu: bảng NgườiDân, LịchTiêm, Vaccine, NhânViên.

4. Implementation (Triển khai lập trình)

- Mục tiêu: Viết mã và xây dựng hệ thống theo bản thiết kế.
Việc thực hiện:
- Xây dựng các module:
- Giao diện người dùng (Frontend – ReactJS hoặc JavaFX, v.v.).
- Dịch vụ xử lý (Backend – Java, Node.js,...).
- Cơ sở dữ liệu (MySQL, PostgreSQL).

5. Testing (Kiểm thử)

- Mục tiêu: Đảm bảo hệ thống hoạt động đúng yêu cầu, không lỗi.
Việc thực hiện:
- Kiểm thử chức năng: đăng ký, xác nhận, theo dõi danh sách.
- Kiểm thử giao diện: hiển thị rõ ràng, dễ dùng.
- Kiểm thử hiệu năng và bảo mật: dữ liệu người dân được bảo vệ, hệ thống chạy ổn định.
- Sửa lỗi và tối ưu.

6. Deployment & Maintenance (Triển khai và bảo trì)

- Mục tiêu: Đưa hệ thống vào sử dụng thực tế và đảm bảo hoạt động lâu dài.

Việc thực hiện:

- Triển khai hệ thống lên máy chủ hoặc nền tảng đám mây.
- Hướng dẫn nhân viên y tế và người dân sử dụng.
- Thu thập phản hồi, cập nhật tính năng mới (ví dụ: thêm loại vaccine, tự động nhắc lịch).
- Theo dõi lỗi, bảo trì định kỳ.

Bài 8 :

1:Xác định tác nhân chính và chức năng tương ứng

- Học viên
 - Đăng ký tài khoản.
 - Đăng ký khóa học.
 - Truy cập bài học (video, tài liệu).
 - Tham gia bài kiểm tra / bài tập.
 - Xem điểm số và tiến độ học tập.
 - Gửi phản hồi / hỏi đáp với giảng viên.
- Giảng viên
 - Tạo khóa học và bài học.
 - Cập nhật tài liệu, video, đề kiểm tra.
 - Chấm điểm và phản hồi bài tập.
 - Theo dõi tiến độ và kết quả học viên.
 - Quản lý lịch dạy, thông báo cho học viên.
- Quản trị viên (Admin)

- Quản lý người dùng (tạo, sửa, xóa, phân quyền).
- Quản lý khóa học (phê duyệt, chỉnh sửa, xóa).
- Xem và xuất báo cáo thống kê (đăng ký, tiến độ, doanh thu).
- Giám sát hoạt động hệ thống, sao lưu dữ liệu và xử lý sự cố.
- Cấu hình các thông số hệ thống (cổng thanh toán, email, phân quyền).

2: Phân loại hệ thống thông tin phù hợp

- TPS (Transaction Processing System)
 - Hệ thống xử lý các giao dịch hàng ngày như: đăng ký khóa học, thanh toán, lưu điểm và lưu tiến độ học viên.
- MIS (Management Information System)
 - Hệ thống tạo báo cáo, thống kê cho người quản lý: báo cáo số học viên, tỷ lệ hoàn thành khóa, báo cáo doanh thu.
- DSS (Decision Support System) — nếu cần
 - Hệ thống phân tích nâng cao để đưa ra gợi ý: đề xuất khóa học phù hợp, phân tích hiệu quả giảng dạy.
- Kết luận ngắn:
 - Nền tảng này chủ yếu là kết hợp TPS + MIS; có thể mở rộng thêm DSS cho phân tích nâng cao.

3: Đề xuất mô hình phát triển phần mềm (và lý do)

- Mô hình đề xuất: Agile (phát triển lặp — Iterative / Scrum)
- Lý do:
 - Ứng dụng có nhiều module (đăng ký, học liệu, chấm điểm, báo cáo) dễ chia nhỏ thành các Sprint.

- Yêu cầu có thể thay đổi (tính năng mới, chỉnh UI) — Agile cho phép điều chỉnh nhanh theo phản hồi.
- Có thể phát hành sớm bản MVP để người dùng thử, thu thập phản hồi và cải tiến dần.
- Tăng tính cộng tác giữa đội dev và bên trung tâm (product owner), giảm rủi ro phát triển sai hướng.
- Ghi chú: Nếu dự án yêu cầu rất cứng về pháp lý, an ninh và thay đổi ít, có thể cân nhắc Waterfall; nhưng thông thường Agile phù hợp hơn.

4: Ba sơ đồ UML sẽ sử dụng khi đi sâu thiết kế

- Use Case Diagram
 - Mô tả các chức năng chính của hệ thống và tương tác giữa các tác nhân (Học viên, Giảng viên, Admin).
- Class Diagram
 - Mô tả cấu trúc dữ liệu: các lớp chính (NguoiDung/HocVien/GiangVien, KhoaHoc, BaiHoc, BaiTap, Diem, DonHang/ThanhToan) và quan hệ giữa chúng.
- Sequence Diagram
 - Mô tả thứ tự tương tác giữa các đối tượng trong quy trình cụ thể, ví dụ: quá trình "Học viên đăng ký khóa học và thanh toán" hoặc "Giảng viên chấm bài".