Bài 1:

* Giao dịch bán hàng : TPS  
  Lý do: xử lý giao dịch hàng ngày theo thời gian thực/đều đặn, đảm bảo tính nhất quán và lưu trữ chi tiết.
* Phân tích xu hướng kinh doanh : DSS   
  Lý do: phân tích dữ liệu lịch sử, mô hình hoá, báo cáo phân tích/biểu đồ, hỗ trợ ra quyết định chiến lược/chi tiết .
* Bảng tổng quan hiệu suất hàng tháng cho CEO : EIS   
  Lý do: cung cấp dashboard tóm tắt KPI cao cấp, xu hướng chính và cảnh báo, giao diện trực quan để ban lãnh đạo nhanh chóng nắm bắt hiệu suất và ra quyết định chiến lược.
* Tổng hợp báo cáo quản trị : MIS

Lý do:Tổng hợp dữ liệu từ TPS thành báo cáo định kỳ/ngoại lệ cho quản lý .

Bài 2

* Dự án A — Waterfall  
  Lý do: yêu cầu rõ ràng, ổn định nên làm theo quy trình tuyến tính tiết kiệm chi phí thay đổi và dễ lên kế hoạch/đánh giá tiến độ.
* Dự án B — Agile   
  Lý do: ứng dụng mobile cần linh hoạt, thay đổi theo phản hồi người dùng và thị trường; Agile cho phép phát hành từng phiên bản , liên tục cải tiến, tích hợp CI/CD và thu thập phản hồi sớm.
* Dự án C — Spiral  
  Lý do: hệ thống ngân hàng điện tử có độ phức tạp cao và rủi ro — Spiral tập trung quản lý rủi ro qua các vòng lặp: phân tích rủi ro → prototyping/kiểm thử → điều chỉnh, phù hợp để giảm rủi ro trước khi triển khai rộng.

Bài 3:

* Con người  
  Vai trò: người dùng tương tác và vận hành hệ thống.  
  Ý nghĩa: quyết định yêu cầu, tinh chỉnh quy trình và xử lý ngoại lệ.
* Dữ liệu  
  Vai trò: thông tin cần thu thập, lưu trữ và xử lý.  
  Ý nghĩa: nền tảng cho xử lý giao dịch, phân tích và ra quyết định.
* Quy trình (business processes)  
  Vai trò: luồng công việc, quy tắc vận hành từ đặt đến giao hàng.  
  Ý nghĩa: đảm bảo trải nghiệm mượt mà, hiệu quả và tuân thủ SLA.
* Phần mềm  
  Vai trò: các ứng dụng và hệ thống thực thi chức năng.  
  Ý nghĩa: thực thi quy trình, giao tiếp giữa các bên, bảo mật và phân tích.
* Phần cứng & hạ tầng mạng  
  Vai trò: nền tảng vật lý/đám mây để chạy phần mềm và kết nối.  
  Ý nghĩa: đảm bảo sẵn sàng, hiệu năng, khả năng mở rộng và kết nối thời gian thực.

Bài 4:

* Planning   
  Xác định mục tiêu dự án, phạm vi, stakeholders ,ngân sách và timeline; xác định rủi ro chính và lập kế hoạch nguồn lực.
* Analysis  
  Thu thập yêu cầu chức năng & phi chức năng: cách điểm danh ,xử lý tình huống ngoại lệ, chính sách bảo mật/kiểm duyệt, báo cáo; phân tích tích hợp với SIS, thanh toán và nghiệp vụ .
* Design   
  Thiết kế kiến trúc hệ thống , mô hình dữ liệu , UI/UX của app giáo viên & sinh viên, luồng nghiệp vụ, sơ đồ bảo mật và kế hoạch triển khai/khả năng mở rộng.
* Implementation   
  Lập trình frontend , backend API, database, tích hợp xác thực , triển khai tính năng điểm danh , gửi thông báo, lưu logs; áp dụng coding standards, CI/CD và viết tài liệu kỹ thuật.
* Testing   
  Kiểm thử đơn vị, tích hợp, hệ thống; kiểm thử UAT với giáo viên/sinh viên; kiểm thử hiệu năng , bảo mật và kiểm thử tính đúng đắn nghiệp vụ .
* Deployment & Maintenance   
  Triển khai lên môi trường production , phát hành app lên store, đào tạo người dùng, thiết lập monitoring, backup và quy trình sao lưu/khôi phục; bảo trì liên tục: sửa lỗi, cập nhật bảo mật, cải tiến theo phản hồi và đảm bảo tuân thủ chính sách bảo vệ dữ liệu.

Bài 5:

Planning:

* Mục tiêu: làm app điểm danh bằng QR cho giảng viên, sinh viên, phòng đào tạo.
* Ai làm: PM, 2 dev mobile, 1 backend, 1 QA, 1 UI.
* Thời gian MVP: ~4 tuần.
* Rủi ro cần lưu ý: gian lận QR, mạng yếu, bảo mật.
* Giao nộp: kế hoạch, timeline, phân công.

Requirement Analysis:

* Hỏi/quan sát: gặp giảng viên, sinh viên, phòng đào tạo để lấy yêu cầu thực tế.
* Chức năng chính: tạo buổi, sinh QR, sinh viên quét, lưu điểm, báo cáo.
* Yêu cầu cần: phản hồi nhanh, an toàn (HTTPS), QR có thời hạn, ưu tiên các tính năng quan trọng trước.
* Giao nộp: danh sách tính năng (user stories) + tiêu chí chấp nhận + mockup nhanh.

System Design:

* Kiến trúc ngắn: App mobile (sv + gv) + Backend API + Database + Dashboard admin.
* Luồng chính: GV tạo buổi → backend sinh QR (có thời hạn) → SV quét → backend xác nhận → lưu điểm và cập nhật báo cáo.
* Cần có: bảng dữ liệu User/Course/Session/Attendance, API cơ bản (tạo session, lấy QR, gửi điểm, lấy báo cáo).
* Bảo mật & chống gian lận: HTTPS, JWT/SSO, QR có TTL + chữ ký, log audit.
* Giao nộp: sơ đồ kiến trúc, ERD đơn giản, spec API, wireframe.

Bài 6:

* A → Use Case diagram (Sơ đồ ca sử dụng)
* B → Class diagram (Sơ đồ lớp)
* C → Activity diagram (Sơ đồ hoạt động)
* D → Deployment diagram (Sơ đồ triển khai)
* E → Sequence diagram (Sơ đồ trình tự)

Bài 7:

1. Planning

* Xác định mục tiêu, stakeholders (dân, trung tâm y tế, admin), phạm vi MVP, đội thực hiện, timeline và rủi ro (lỗi lịch, bảo mật).
* Deliverable: project plan, timeline, phân công.

1. Analysis

* Họp hỏi nhu cầu: đăng ký, xác nhận lịch, quản lý danh sách đã/chuẩn bị tiêm, báo cáo; liệt kê yêu cầu chức năng & phi chức năng (bảo mật, tải, SLA).
* Deliverable: SRS / user stories + tiêu chí chấp nhận.

1. Design

* Thiết kế kiến trúc (app/web + backend + DB), mô hình dữ liệu (người dân, lịch, mũi tiêm, trung tâm), flow đăng ký → xác nhận → nhắc lịch.
* Deliverable: ERD, sơ đồ kiến trúc, wireframe, API spec.

1. Implementation

* Code app/website, API, DB; tích hợp gửi SMS/email, xác thực, quản trị trung tâm; xây CI/CD.
* Deliverable: bản chạy chức năng (MVP), mã nguồn, docs cài đặt.

1. Testing

* Kiểm thử chức năng, tích hợp, hiệu năng (đợt đăng ký cao điểm), bảo mật (pen-test), UAT với nhân viên trung tâm.
* Deliverable: test report, bug list, bản sửa lỗi.

1. Deployment & Maintenance

* Triển khai production, backup/restore, monitoring, đào tạo nhân viên, hỗ trợ vận hành, cập nhật vá lỗi và nâng cấp.
* Deliverable: môi trường live, hướng dẫn sử dụng, kế hoạch bảo trì.

Bài 8:

1. Tác nhân chính và chức năng

* Học viên: đăng ký khóa học, tham gia học, xem điểm.
* Giảng viên: tạo/quản lý bài học, nhập điểm.
* Admin: quản lý người dùng, phân quyền, thống kê/báo cáo.

2. Phân loại hệ thống thông tin

* TPS (Transaction Processing System): đăng ký khóa học, nhập điểm.
* MIS (Management Information System): báo cáo, thống kê cho admin.
* DSS (Decision Support System): phân tích dữ liệu học viên để hỗ trợ quyết định (nếu mở rộng).

3. Mô hình phát triển phần mềm phù hợp

Mô hình Agile:Yêu cầu có thể thay đổi, cần phát triển linh hoạt, chia nhỏ chức năng, phản hồi nhanh từ người dùng.

4. Ba sơ đồ UML sẽ sử dụng

* Use Case Diagram (mô tả chức năng và tác nhân).
* Class Diagram (mô hình hóa đối tượng và quan hệ).
* Sequence Diagram (mô tả luồng tương tác theo thời gian).

Bài 9:

1. Tác nhân và chức năng

* Khách hàng: tạo đơn, theo dõi trạng thái giao hàng
* Nhân viên vận chuyển: cập nhật trạng thái, vị trí giao
* Quản lý: theo dõi hiệu suất, khu vực giao
* CEO: xem báo cáo tổng quan, tỉ lệ giao đúng hẹn

2. Phân loại theo hệ thống thông tin

* Tạo đơn, cập nhật trạng thái → TPS
* Theo dõi hiệu suất, khu vực → MIS
* Phân tích hiệu quả, hỗ trợ quyết định → DSS
* Xem báo cáo chiến lược, KPI → EIS

3. Mô hình phát triển phần mềm

Chọn Agile (Scrum)  
Lý do: Yêu cầu thay đổi linh hoạt, nhiều bên liên quan, cần phản hồi nhanh, phát triển theo vòng lặp.

4. 4 sơ đồ UML và mục đích

* Use Case Diagram: mô tả tác nhân và chức năng
* Class Diagram: mô hình hóa dữ liệu và quan hệ
* Sequence Diagram: mô tả luồng tương tác theo thời gian
* Activity Diagram: mô tả quy trình xử lý đơn hàng