**Kế hoạch Tuần 1: Phân tích Yêu cầu – Xác định Use Case và Actor**

**1. Mục tiêu Tuần 1**

* Hiểu rõ hệ thống Quản lý Đăng ký Học theo Tín chỉ.
* Xác định các Actor (người dùng, vai trò).
* Liệt kê các Use Case (chức năng hệ thống).
* Viết tài liệu yêu cầu: bao gồm yêu cầu chức năng và phi chức năng.
* Hoàn thành bản báo cáo 5–7 trang mô tả yêu cầu hệ thống.

**2. Công việc cần thực hiện**

**Bước 1: Nghiên cứu & phân tích**

Đọc và hiểu đề tài.

* Xác định các Actor chính: Sinh viên, Giảng viên, Quản trị viên, Hệ thống.
* Liệt kê các chức năng mà mỗi Actor cần thực hiện.

**Bước 2: Xác định Use Case**

* Vẽ sơ đồ Use Case UML (có thể dùng draw.io, Visio, hoặc Word).
* Mô tả chi tiết từng Use Case (ai thực hiện, hành động gì, kết quả mong đợi).

**Bước 3: Viết tài liệu yêu cầu**

* Yêu cầu chức năng (Functional Requirements).
* Yêu cầu phi chức năng (Non-Functional Requirements).
* Viết mô tả hệ thống ở mức tổng quan.

**Bước 4: Hoàn thiện báo cáo**

* Hoàn chỉnh báo cáo 5–7 trang (bao gồm: Giới thiệu, Actor, Use Case, Yêu cầu hệ thống, Kết luận).
* Thêm biểu đồ Use Case UML vào báo cáo.
* Kiểm tra chính tả, định dạng, trích dẫn.

**Bước 5: Nộp và trình bày**

* Xuất báo cáo dưới dạng Word/PDF.
* Chuẩn bị 3–5 phút trình bày ngắn (nếu cần).

**3. Phân công công việc (nếu làm theo nhóm)**

* **Đỗ Hoài Nam**: Phân tích Actor + Use Case, vẽ sơ đồ.
* **Nguyễn Văn Minh**: Viết yêu cầu chức năng & phi chức năng.
* **Đỗ Hoài Nam**: Hoàn thiện báo cáo, trình bày, định dạng.

**4. Kết quả mong đợi cuối tuần**

* Báo cáo hoàn chỉnh 5–7 trang.
* Sơ đồ Use Case UML.
* Danh sách Actor và Use Case.
* Tài liệu yêu cầu chức năng và phi chức năng.

**Kế hoạch Tuần 2 – Mô hình hóa Use Case và Kịch bản**

**Mục tiêu tuần 2**

* Trực quan hóa các Use Case chính của hệ thống bằng biểu đồ.
* Viết kịch bản chi tiết cho từng Use Case (bao gồm luồng chính và luồng phụ).
* Đảm bảo mỗi Use Case đều có **Actor**, **tiền điều kiện**, **hậu điều kiện**, **luồng chính**, **luồng phụ**.
* Hoàn thành sơ đồ Use Case cho các chức năng quan trọng.

**Nội dung công việc**

**1. Vẽ Biểu đồ Use Case tổng thể**

* Xác định các **Actor** chính:
  + Sinh viên
  + Giảng viên
  + Quản trị viên
* Xác định các Use Case chính của hệ thống (ví dụ):
  + Đăng ký môn học
  + Hủy đăng ký môn học
  + Xem danh sách môn học
  + Mở lớp học
  + Quản lý tài khoản người dùng

Kết quả: Một sơ đồ Use Case tổng quan hệ thống.

**2.Viết kịch bản chi tiết cho từng Use Case**

* **Use Case 1: Đăng ký môn học (Sinh viên)**
* **Use Case 2: Hủy đăng ký môn học (Sinh viên)**
* **Use Case 3: Mở lớp học (Giảng viên)**
* (Các Use Case khác có thể bổ sung sau: Xem lịch học, Xem kết quả, Quản lý lớp học, Quản lý người dùng…)

Mỗi Use Case gồm:

* Actor chính
* Mục tiêu
* Tiền điều kiện
* Hậu điều kiện
* Luồng chính
* Luồng phụ

**3. Vẽ biểu đồ kịch bản chi tiết cho từng Use Case**

* Mỗi Use Case sẽ có **sơ đồ hoạt động hoặc biểu đồ Use Case chi tiết** minh họa.
* Ví dụ:
  + Use Case 1: Đăng ký môn học → Luồng chọn môn học → kiểm tra điều kiện → thêm vào danh sách → thông báo kết quả.
  + Use Case 2: Hủy môn học → Chọn môn cần hủy → kiểm tra thời hạn → xác nhận → thông báo kết quả.
  + Use Case 3: Mở lớp học → Nhập thông tin lớp → kiểm tra phòng/lịch → lưu thông tin → thông báo kết quả.

**Phân công công việc**

**Đỗ Hoài Nam**

* Vẽ **sơ đồ Use Case tổng thể** cho hệ thống.
* Viết kịch bản chi tiết cho:
  + Use Case 1: Đăng ký môn học
  + Use Case 2: Hủy đăng ký môn học

**Nguyễn Văn Minh**

* Vẽ sơ đồ chi tiết cho từng Use Case (1, 2, 3).
* Viết kịch bản chi tiết cho:
  + Use Case 3: Mở lớp học
  + Các Use Case phụ (nếu kịp tiến độ: Xem lịch học, Xem kết quả học tập).

**Kết quả mong đợi sau Tuần 2**

* Hoàn thành **1 sơ đồ Use Case tổng quan** của hệ thống.
* Hoàn thành **ít nhất 3 kịch bản chi tiết Use Case** (Đăng ký môn, Hủy môn, Mở lớp học).
* Có sơ đồ minh họa (vẽ bằng Draw.io) cho từng Use Case chính.
* Tài liệu tuần 2 được tập hợp thành báo cáo nhóm.

**Kế hoạch chi tiết Tuần 3 – Thiết kế Lớp và Tạo Cơ sở Code**

**Mục tiêu tuần 3**

* Xây dựng **biểu đồ lớp UML** để mô tả cấu trúc dữ liệu và đối tượng chính.
* Xác định **thuộc tính (attributes)**, **phương thức (methods)** và **mối quan hệ** giữa các lớp.
* Khởi tạo **các file lớp cơ sở** (chỉ khai báo tên lớp, thuộc tính và phương thức rỗng) để chuẩn bị cho việc phát triển chức năng trong các tuần tiếp theo.

**Nội dung công việc**

**1. Thiết kế Biểu đồ Lớp UML**

* **Xác định các lớp chính**:
  + SinhVien
  + GiangVien
  + MonHoc
  + LopHoc (hoặc LopHocPhan)
  + KetQuaHocTap
* **Xác định thuộc tính của mỗi lớp** (ví dụ: SinhVien có mã số, họ tên, ngày sinh; MonHoc có mã môn, số tín chỉ; LopHoc có thời gian, phòng học, sĩ số tối đa…).
* **Xác định phương thức chính** (ví dụ: SinhVien có đăng ký môn, hủy môn; GiangVien có mở lớp, cập nhật điểm; LopHoc có thêm/xóa sinh viên; KetQuaHocTap có tính điểm trung bình…).
* **Xác định mối quan hệ giữa các lớp**:
  + SinhVien ↔ LopHoc: quan hệ nhiều-nhiều.
  + LopHoc ↔ MonHoc: quan hệ một-nhiều.
  + GiangVien ↔ LopHoc: quan hệ một-nhiều.
  + SinhVien ↔ KetQuaHocTap ↔ MonHoc: dùng để lưu kết quả học tập.

Kết quả: Biểu đồ lớp UML hoàn chỉnh.

**2. Khởi tạo các file lớp cơ sở**

* Tạo file cho từng lớp chính (ví dụ: SinhVien, GiangVien, MonHoc, LopHoc, KetQuaHocTap).
* Khai báo các thuộc tính và phương thức (chưa cần viết logic).
* Đảm bảo thống nhất về cách đặt tên và mối quan hệ giữa các file.

**3. Chuẩn bị tài liệu mô tả**

* Mô tả chi tiết từng lớp: mục đích, thuộc tính, phương thức.
* Giải thích mối quan hệ giữa các lớp trong hệ thống.
* Đưa biểu đồ lớp UML và phần mô tả vào báo cáo.

**Phân công công việc**

**Đỗ Hoài Nam**

* Vẽ biểu đồ lớp UML tổng thể.
* Mô tả chi tiết lớp SinhVien và MonHoc.
* Chuẩn bị file lớp cơ sở cho SinhVien và MonHoc.

**Nguyễn Văn Minh**

* Mô tả chi tiết lớp GiangVien, LopHoc, KetQuaHocTap.
* Khởi tạo file lớp cơ sở tương ứng.
* Tích hợp các mối quan hệ giữa các lớp vào biểu đồ và tài liệu.

**Kết quả mong đợi cuối tuần**

* **Biểu đồ lớp UML đầy đủ** mô tả toàn bộ hệ thống.
* **5 file lớp cơ sở** cho các đối tượng chính.
* **Tài liệu mô tả chi tiết lớp và mối quan hệ**, dùng làm nền cho phát triển hệ thống ở tuần tiếp theo.