 

PROGRAM TITLE: BTEC in Computing (Software Engineering)

UNIT TITLE: Software Development Lifecycles

ASSIGNMENT NUMBER: 1

ASSIGNMENT NAME: Tran Huy Hoang

SUBMISSION DATE: 31/10/2024

DATE RECEIVED: 31/10/2024

TUTORIAL LECTURER: Nguyen Van Tuyen

WORD COUNT:

STUDENT NAME: Tran Huy Hoang

STUDENT ID: BKC15018

MOBILE NUMBER: 0969893512

## **I. Mô tả vòng đời phát triển phần mềm (LO1)**

### 1. Vòng đời phát triển phần mềm (SDLC) (P1)

* **Vòng đời phát triển phần mềm (SDLC) là gì?**
  + Là quy trình tiêu chuẩn được sử dụng để phát triển phần mềm, đảm bảo phần mềm được tạo ra đúng cách, hiệu quả và phù hợp với yêu cầu của người dùng.
* **SDLC có mấy giai đoạn?**
  + Các giai đoạn:
    - Khởi đầu (Planning)
    - Phân tích yêu cầu (Requirement Analysis)
    - Thiết kế (Design)
    - Phát triển (Development)
    - Kiểm thử (Testing)
    - Triển khai (Deployment)
    - Bảo trì (Maintenance)
* **Hai mô hình vòng đời tuần tự**
  + **Tổng quan:**
    - Các mô hình tuần tự thực hiện các giai đoạn theo thứ tự nhất định, không quay lại giai đoạn trước khi hoàn tất.
  + **Mô hình có những giai đoạn nào?**
    - **Waterfall**:
      * Lần lượt đi qua các giai đoạn: Khởi đầu, Phân tích yêu cầu, Thiết kế, Phát triển, Kiểm thử, Triển khai, Bảo trì.
    - **V-Shaped**:
      * Mô hình V tương tự Waterfall nhưng mỗi giai đoạn phát triển đi kèm với một giai đoạn kiểm thử tương ứng.
  + **Mô hình có ưu điểm, nhược điểm gì?**
    - **Waterfall**:
      * Ưu điểm: Dễ hiểu, quản lý dự án dễ dàng, phù hợp với các dự án có yêu cầu rõ ràng từ đầu.
      * Nhược điểm: Không linh hoạt, khó thay đổi khi dự án đã qua các giai đoạn sau.
    - **V-Shaped**:
      * Ưu điểm: Tăng cường khả năng kiểm soát chất lượng, mỗi giai đoạn phát triển đi kèm kiểm thử.
      * Nhược điểm: Tương tự Waterfall, không linh hoạt và khó điều chỉnh nếu có thay đổi yêu cầu.
* **Hai mô hình vòng đời lặp**
  + **Tổng quan về mô hình:**
    - Mô hình lặp cho phép quay lại và cải tiến các giai đoạn trước đó, thường sử dụng khi yêu cầu phần mềm không rõ ràng hoặc có thể thay đổi trong quá trình phát triển.
  + **Mô hình có những giai đoạn nào?**
    - **Iterative**:
      * Các giai đoạn: Lặp lại nhiều lần quy trình từ Phân tích, Thiết kế, Phát triển, Kiểm thử cho đến khi hoàn thiện sản phẩm.
    - **Spiral**:
      * Các giai đoạn: Tương tự Iterative nhưng tập trung vào quản lý rủi ro, mỗi vòng lặp (spiral) có các bước: Xác định mục tiêu, Phân tích rủi ro, Phát triển và Kiểm thử, Đánh giá.
  + **Mô hình có ưu điểm, nhược điểm gì?**
    - **Iterative**:
      * Ưu điểm: Linh hoạt, dễ thay đổi và điều chỉnh sản phẩm theo yêu cầu.
      * Nhược điểm: Có thể tăng thời gian và chi phí phát triển.
    - **Spiral**:
      * Ưu điểm: Tập trung quản lý rủi ro, phù hợp với các dự án phức tạp.
      * Nhược điểm: Yêu cầu kiến thức cao, chi phí và thời gian phát triển lớn.

### 2. Quản lý rủi ro trong các mô hình (P2)

### ****Rủi ro trong SDLC là gì?****

### **Rủi ro trong SDLC là những yếu tố có thể gây ảnh hưởng tiêu cực đến tiến trình phát triển phần mềm, bao gồm các yếu tố như thay đổi yêu cầu, lỗi kỹ thuật, chi phí vượt ngân sách, và sự chậm trễ.**

### ****Quản lý rủi ro của các mô hình là gì?****

* + Quản lý rủi ro trong SDLC là quá trình xác định, đánh giá, và đưa ra các biện pháp kiểm soát để giảm thiểu hoặc loại bỏ những rủi ro có thể xảy ra trong quá trình phát triển phần mềm.

1. **Trình bày các cách quản lý rủi ro trong một số mô hình SDLC:**
   * **Waterfall**: Quản lý rủi ro thông qua kế hoạch chi tiết từ đầu, nhưng khả năng ứng phó với rủi ro thấp.
   * **Spiral**: Mỗi vòng lặp tập trung vào việc xác định và giảm thiểu rủi ro, giúp quản lý rủi ro tốt hơn.

### 3. Ví dụ và giải thích tại sao cần lựa chọn mô hình phát triển phù hợp với dự án (M1)

1. **Đưa ra một dự án ví dụ:**
   1. Ví dụ: Phát triển một hệ thống quản lý trường học.
2. **Lựa chọn mô hình phát triển cho dự án trên:**
   1. Chọn mô hình **Iterative**.
3. **Giải thích lý do lựa chọn mô hình đó:**
   1. Vì yêu cầu dự án có thể thay đổi khi nhận phản hồi từ người dùng, Iterative cho phép điều chỉnh và cải tiến phần mềm một cách linh hoạt.

### 4. Ví dụ cụ thể về việc 1 phần mềm lớn áp dụng hiệu quả mô hình Waterfall (D1)

* **Trình bày ví dụ về một phần mềm lớn áp dụng hiệu quả mô hình Waterfall:**
  + Ví dụ: Phát triển hệ thống quản lý ngân hàng.
* **Ưu điểm, nhược điểm của việc áp dụng mô hình Waterfall vào dự án trên là gì?**
  + Ưu điểm: Dễ quản lý, phù hợp với dự án có yêu cầu rõ ràng và ít thay đổi.
  + Nhược điểm: Khó khăn trong việc điều chỉnh nếu có yêu cầu thay đổi giữa quá trình phát triển.
* **Đánh giá việc áp dụng mô hình Waterfall có hiệu quả với dự án trên không? Vì sao?**
  + Hiệu quả vì dự án có yêu cầu cố định, việc tuân thủ quy trình Waterfall đảm bảo chất lượng và tiến độ.

## II. Nghiên cứu khả thi (LO2)

* **Báo cáo khả thi (P3)**
  + **Báo cáo khả thi là gì?**

- Báo cáo khả thi là tài liệu phân tích và đánh giá tính khả thi của một dự án, giúp các bên liên quan quyết định có nên tiến hành dự án hay không.

* + **Mục đích của báo cáo khả thi là gì?**

**-** Mục đích của báo cáo khả thi là xác định liệu dự án có khả thi về mặt kỹ thuật, tài chính và thời gian hay không, từ đó hỗ trợ việc ra quyết định.

* **Làm thế nào để so sánh các giải pháp công nghệ có thể áp dụng vào dự án? (P4)**
  + Để so sánh các giải pháp công nghệ, bạn cần xem xét các yếu tố sau:
    - **Khả năng tương thích**: Giải pháp đó có thể tích hợp với hệ thống hiện tại không?
    - **Tính linh hoạt**: Giải pháp có thể dễ dàng thay đổi và mở rộng trong tương lai không
    - **Hiệu suất**: Giải pháp đó có đáp ứng được yêu cầu hiệu suất của dự án không?
    - **Chi phí**: So sánh chi phí triển khai, bảo trì và vận hành.
    - **Bảo mật**: Giải pháp có đảm bảo an toàn cho dữ liệu và hệ thống không?
    - **Kinh nghiệm và hỗ trợ**: Đánh giá từ những người dùng khác và khả năng hỗ trợ từ nhà cung cấp.
* **Các thành phần của một báo cáo khả thi (M2)**
  + Một báo cáo khả thi bao gồm các thành phần chính sau:
    - **Tóm tắt dự án**: Giới thiệu dự án, mục tiêu và phạm vi.
      * **Tác dụng**: Cung cấp cái nhìn tổng quan nhanh cho người đọc.
    - **Phân tích thị trường**: Đánh giá nhu cầu và tiềm năng thị trường.
      * **Tác dụng**: Xác định mức độ cần thiết của dự án trong thị trường.
    - **Phân tích kỹ thuật**: Khả năng triển khai về mặt kỹ thuật, công nghệ.
      * **Tác dụng**: Đảm bảo rằng dự án khả thi về mặt công nghệ.
    - **Phân tích tài chính**: Dự đoán chi phí, lợi nhuận, và các chỉ số tài chính khác.
      * **Tác dụng**: Giúp đánh giá tính khả thi về mặt tài chính.
    - **Đánh giá pháp lý**: Xem xét các quy định, chính sách pháp lý liên quan.
      * **Tác dụng**: Đảm bảo dự án tuân thủ quy định pháp luật.
    - **Kết luận và đề xuất**: Tóm tắt các kết quả phân tích và đưa ra khuyến nghị.
      * **Tác dụng**: Cung cấp quyết định về việc có nên triển khai dự án hay không.
* **Tác động của các tiêu chí khả thi (D2)**

1. **Một số tiêu chí đánh giá tính khả thi của dự án:**
   * **Khả thi về công nghệ**: Đánh giá liệu công nghệ hiện tại có đủ khả năng để thực hiện dự án không.
     + - **Tác động**: Công nghệ không phù hợp có thể làm trì hoãn hoặc tăng chi phí dự án.
   * **Khả thi về tài chính**: Xem xét liệu nguồn lực tài chính có đủ để hoàn thành dự án không.
     + - **Tác động**: Nếu không đủ vốn, dự án có thể gặp rủi ro tài chính, ảnh hưởng đến tiến độ và chất lượng.
   * **Khả thi về nguồn nhân lực**: Đánh giá đội ngũ nhân viên có đủ kỹ năng và kiến thức để thực hiện dự án không.
     + - **Tác động**: Thiếu nguồn nhân lực phù hợp có thể làm giảm hiệu quả triển khai.
   * **Khả thi về thời gian**: Đánh giá khả năng hoàn thành dự án trong khung thời gian dự kiến.
     + - **Tác động**: Nếu thời gian không đủ, có thể dẫn đến chất lượng không đạt yêu cầu hoặc vượt ngân sách.
2. **Ví dụ về tác động của các tiêu chí:**
   1. Nếu một dự án phần mềm thiếu nguồn nhân lực có chuyên môn cao về công nghệ mới, thời gian triển khai sẽ kéo dài và chi phí có thể đội lên do phải thuê chuyên gia hoặc đào tạo nội bộ.

### III. Thực hiện vòng đời phát triển phần mềm (LO3)

#### 1. Thực hiện một cuộc điều tra phần mềm (P5)

**a. Trình bày về dự án đã chọn:** Dự án "Hệ thống theo dõi điểm danh sinh viên" là một giải pháp công nghệ giúp tự động hóa quá trình ghi nhận và quản lý việc tham gia của sinh viên trong lớp học.

**b. Trình bày các yêu cầu chức năng của dự án:**

* Ghi nhận sự có mặt của sinh viên.
* Quản lý danh sách sinh viên và các lớp học.
* Báo cáo tình trạng điểm danh theo thời gian thực.
* Thông báo tự động đến sinh viên và giảng viên về tình trạng điểm danh.
* Phân quyền cho quản trị viên, giảng viên và sinh viên.

**c. Trình bày các yêu cầu phi chức năng của dự án:**

* **Tính bảo mật**: Hệ thống phải đảm bảo bảo mật dữ liệu cá nhân của sinh viên.
* **Tính khả dụng**: Hệ thống cần hoạt động ổn định, đáp ứng lượng lớn người dùng truy cập đồng thời.
* **Tính mở rộng**: Dễ dàng tích hợp thêm các thiết bị điểm danh khác hoặc mở rộng cho nhiều lớp học.
* **Tính tương thích**: Có thể sử dụng trên nhiều nền tảng và thiết bị khác nhau.

**d. Trình bày về tính khả thi của dự án:**

* **Tính khả thi về công nghệ**: Sử dụng các công nghệ phổ biến đã được ứng dụng rộng rãi trong thực tế.
* **Chi phí**: Tùy thuộc vào quy mô và thiết bị sử dụng, nhưng hệ thống có thể triển khai với mức chi phí hợp lý.
* **Nhân lực** Đội ngũ phát triển cần có kiến thức về lập trình phần mềm, bảo mật và tích hợp hệ thống với các phần cứng khác nhau.

#### 2. Sử dụng các công cụ/ kỹ thuật phân tích phần mềm để thực hiện điều tra phần mềm (P6)

**a. Phần mềm có các bên liên quan nào?**

* **Giảng viên**: Người sử dụng hệ thống để kiểm tra sự có mặt của sinh viên.
* **Sinh viên**: Người được điểm danh và theo dõi tình trạng tham gia lớp học.
* **Quản trị viên**: Người quản lý hệ thống, cập nhật dữ liệu và báo cáo.
* **Bộ phận quản lý nhà trường**: Sử dụng dữ liệu điểm danh để theo dõi và đánh giá sinh viên.

**b. Sơ đồ luồng dữ liệu (DFD)**

**i. Sơ đồ luồng dữ liệu là gì?** Sơ đồ luồng dữ liệu (DFD) là công cụ để biểu diễn luồng thông tin trong một hệ thống phần mềm. Nó mô tả quá trình xử lý và trao đổi dữ liệu giữa các thực thể, hệ thống con và cơ sở dữ liệu.

**ii. Sơ đồ luồng dữ liệu có các mức nào?**

* **Mức 0**: Biểu diễn tổng quan hệ thống, mô tả các thực thể chính và các luồng dữ liệu chính.
* **Mức 1**: Chi tiết hóa các quy trình chính trong hệ thống từ mức 0, cho thấy các bước xử lý dữ liệu cụ thể.

**iii. Trình bày sơ đồ DFD mức 0 của phần mềm:** Sơ đồ DFD mức 0 mô tả tổng quan hệ thống với các thực thể chính như Giảng viên, Sinh viên, Quản trị viên, và cơ sở dữ liệu trung tâm. Các luồng dữ liệu sẽ bao gồm thông tin điểm danh, báo cáo và thông báo.

**iv. Trình bày sơ đồ DFD mức 1 của phần mềm:** Sơ đồ DFD mức 1 sẽ chi tiết hơn về các quá trình xử lý dữ liệu như:

* Quy trình nhận diện và xác thực sinh viên.
* Quy trình ghi nhận điểm danh.
* Quy trình báo cáo và gửi thông báo.

**c. Sơ đồ ERD của phần mềm:** Sơ đồ ERD (Entity Relationship Diagram) mô tả các thực thể trong hệ thống và mối quan hệ giữa chúng. Trong hệ thống này, các thực thể chính có thể bao gồm: Sinh viên, Lớp học, Buổi học, Giảng viên, và các mối quan hệ giữa chúng như "tham gia", "dạy", "điểm danh".

#### 3. Theo dõi các yêu cầu phần mềm (M3)

**a. Có các giải pháp nào để quản lý các phiên bản của phần mềm?**

* **Git**: Sử dụng hệ thống quản lý phiên bản phân tán để theo dõi các thay đổi của mã nguồn.
* **Subversion (SVN)**: Một giải pháp quản lý phiên bản tập trung, giúp kiểm soát các phiên bản khác nhau của phần mềm.

**b. Trình bày Schedule Baseline của dự án:** Schedule Baseline là lịch trình dự kiến ban đầu của dự án, bao gồm các mốc thời gian quan trọng như:

* Giai đoạn phân tích yêu cầu.
* Giai đoạn thiết kế hệ thống.
* Giai đoạn phát triển và kiểm thử.
* Giai đoạn triển khai và bảo trì.

#### 4. Trình bày về 2 cách tiếp cận cải thiện chất lượng phần mềm (M4)

**a. Tổng quan về cách tiếp cận:**

1. **Agile**: Cách tiếp cận phát triển phần mềm linh hoạt, ưu tiên phản hồi nhanh chóng và cải tiến liên tục qua các vòng lặp nhỏ.
2. **DevOps**: Kết hợp giữa phát triển phần mềm và vận hành, giúp tối ưu hóa quá trình triển khai và bảo trì phần mềm.

**b. Cách tiếp cận đó có ưu điểm, nhược điểm gì?**

* **Agile**:
  + Ưu điểm: Phản hồi nhanh chóng với yêu cầu thay đổi, tăng cường sự tham gia của khách hàng.
  + Nhược điểm: Có thể gây khó khăn trong việc quản lý đối với các dự án lớn và phức tạp.
* **DevOps**:
  + Ưu điểm: Tăng cường sự phối hợp giữa phát triển và vận hành, giảm thời gian triển khai.
  + Nhược điểm: Yêu cầu kỹ năng đa dạng từ đội ngũ và sự tích hợp phức tạp giữa các công cụ.

#### 5. Đánh giá hiệu quả của quá trình thực hiện điều tra hệ thống trong việc cải thiện chất lượng phần mềm (D3)

**a. Quá trình điều tra hệ thống là gì?** Quá trình điều tra hệ thống bao gồm việc thu thập, phân tích và đánh giá các yêu cầu phần mềm từ các bên liên quan, nhằm đảm bảo phần mềm được phát triển đáp ứng các nhu cầu và mong đợi.

**b. Đánh giá hiệu quả của quá trình thực hiện điều tra hệ thống trong việc cải thiện chất lượng phần mềm:**

* **Chất lượng phần mềm**: Điều tra hệ thống giúp phát hiện sớm các yêu cầu quan trọng và tránh lỗi phát sinh do hiểu sai.
* **Hiệu suất hệ thống**: Xác định các yêu cầu tối ưu hóa hiệu suất từ đầu giúp hệ thống hoạt động mượt mà và nhanh chóng.
* **Trải nghiệm người dùng**: Việc điều tra yêu cầu của người dùng giúp cải thiện giao diện và tính năng, mang lại trải nghiệm tốt hơn.
* **Hiệu suất dự án**: Điều tra rõ ràng giúp dự án diễn ra theo đúng tiến độ và giảm thiểu sự chậm trễ do phát sinh yêu cầu mới.

IV. Các kỹ thuật thiết kế hành vi phần mềm (LO4)

1. Thảo luận về thiết kế hành vi phần mềm (P7)

a. Phần mềm đã đáp ứng các yêu cầu người dùng hay chưa? Phần mềm theo dõi điểm danh sinh viên cần đáp ứng các yêu cầu như điểm danh tự động, báo cáo nhanh chóng, thông báo kịp thời và tính bảo mật cao. Qua quá trình điều tra và phân tích, phần mềm đáp ứng được các yêu cầu cơ bản của người dùng:

Giảng viên có thể quản lý danh sách sinh viên và theo dõi tình trạng tham gia lớp học dễ dàng.

Sinh viên được nhận thông báo về tình trạng điểm danh và có thể xem lịch sử điểm danh cá nhân.

b. Phần mềm có các chức năng gì?

Điểm danh tự động: Hệ thống có thể điểm danh qua nhận diện khuôn mặt hoặc quét thẻ sinh viên.

Quản lý lớp học: Giảng viên có thể tạo lớp học mới, cập nhật danh sách sinh viên, và theo dõi điểm danh.

Thông báo tự động: Hệ thống gửi thông báo về tình trạng điểm danh đến sinh viên qua email hoặc ứng dụng.

Báo cáo: Hệ thống cung cấp các báo cáo chi tiết về tỉ lệ tham gia của sinh viên theo từng buổi học, lớp học.

Hình ảnh minh họa cho các chức năng:

Giao diện trang điểm danh: Gồm danh sách sinh viên và trạng thái "Có mặt" hoặc "Vắng mặt".

Giao diện báo cáo: Đồ thị thống kê tỷ lệ sinh viên tham gia qua các buổi học.

c. Phần mềm được sử dụng các công cụ nào để phát triển?

Ngôn ngữ lập trình: TypeScript, PHP.

Công cụ phát triển: Visual Studio Code, XAMPP.

Cơ sở dữ liệu: MariaDB.

Thư viện/Framework: Tailwind, Laravel, Prettier, Vite.

Node Package Manager: Bun.

2. Phân tích các công cụ, hành vi phần mềm (M5)

a. Trình bày 2 phương pháp đặc tả hành vi phần mềm và minh họa bằng ví dụ:

Use Case (Trường hợp sử dụng): Phương pháp này mô tả cách phần mềm tương tác với người dùng và các bên liên quan. Ví dụ: "Giảng viên điểm danh sinh viên" là một Use Case, trong đó giảng viên sẽ chọn lớp học và hệ thống sẽ ghi nhận tình trạng có mặt của sinh viên.

State Diagram (Sơ đồ trạng thái): Phương pháp này mô tả trạng thái của hệ thống hoặc một thành phần phần mềm thay đổi như thế nào trong quá trình hoạt động. Ví dụ: Sinh viên có thể ở các trạng thái như "Chờ điểm danh", "Đã điểm danh", hoặc "Vắng mặt".

3. Máy trạng thái hữu hạn (FSM) và máy trạng thái mở rộng (M6)

a. Máy trạng thái hữu hạn (FSM) trong lĩnh vực phần mềm là gì? Máy trạng thái hữu hạn (FSM) là mô hình tính toán được sử dụng để biểu diễn và kiểm soát luồng trạng thái của hệ thống phần mềm. FSM bao gồm các trạng thái cụ thể và các sự kiện kích hoạt chuyển đổi giữa các trạng thái đó.

b. Máy trạng thái mở rộng (EFSM) trong lĩnh vực phần mềm là gì? Máy trạng thái mở rộng (EFSM) là phiên bản nâng cao của FSM, cho phép có thêm các điều kiện và biến số liên quan trong quá trình chuyển đổi trạng thái. EFSM linh hoạt hơn FSM trong việc mô tả các hệ thống phức tạp.

c. Trình bày ví dụ ứng dụng FSM và EFSM trong lĩnh vực phần mềm:

Ví dụ FSM: Hệ thống điểm danh có thể có trạng thái "Chờ điểm danh", "Điểm danh thành công", và "Điểm danh thất bại". Khi sinh viên quét thẻ hoặc khuôn mặt, hệ thống sẽ chuyển từ "Chờ điểm danh" sang trạng thái khác dựa trên kết quả.

Ví dụ EFSM: Trong EFSM, nếu có thêm điều kiện thời gian, hệ thống chỉ cho phép điểm danh thành công nếu sinh viên quét thẻ trong khoảng thời gian quy định. Điều kiện này giúp tăng độ phức tạp của trạng thái.

4. Phát triển phần mềm theo hướng dữ liệu có thể cải thiện độ tin cậy và hiệu quả của phần mềm như thế nào (D4)

a. Phần mềm theo hướng dữ liệu là gì? Phần mềm theo hướng dữ liệu là phần mềm được thiết kế và xây dựng dựa trên việc xử lý và phân tích dữ liệu. Nó tối ưu hóa việc lưu trữ, truy xuất và phân tích dữ liệu để phục vụ các yêu cầu của người dùng.

b. Phần mềm theo hướng dữ liệu cải thiện độ tin cậy và hiệu quả của phần mềm như thế nào?

Độ tin cậy:

Tỷ lệ lỗi thấp: Phần mềm theo hướng dữ liệu có khả năng giảm thiểu lỗi nhờ việc xử lý dữ liệu chặt chẽ và phát hiện lỗi nhanh chóng.

Thời gian hoạt động (Uptime): Dữ liệu được xử lý và lưu trữ hiệu quả, giúp phần mềm hoạt động liên tục mà không bị gián đoạn.

Độ chính xác: Phần mềm có khả năng cung cấp dữ liệu chính xác và đáng tin cậy trong việc điểm danh và báo cáo.

Hiệu quả:

Hiệu suất xử lý: Xử lý dữ liệu nhanh chóng và chính xác giúp tăng cường khả năng phản hồi của hệ thống.

Khả năng mở rộng: Dễ dàng mở rộng để quản lý lượng dữ liệu lớn khi có nhiều sinh viên và lớp học hơn.

c. Phần mềm dự án đã triển khai bên trên đã đáp ứng được các tiêu chí phát triển phần mềm theo hướng dữ liệu chưa?

Đạt được:

Hệ thống đã đáp ứng tốt khả năng xử lý dữ liệu điểm danh và cung cấp báo cáo chính xác về sự có mặt của sinh viên.

Hiệu suất ổn định, hỗ trợ xử lý nhiều lớp học cùng lúc và có khả năng mở rộng khi cần.

Cải tiến: Có thể cải tiến bằng cách tối ưu hóa quy trình lưu trữ dữ liệu trên các hệ thống điện toán đám mây để đảm bảo thời gian truy cập dữ liệu nhanh hơn, và tích hợp thêm các mô hình phân tích dữ liệu để phát hiện xu hướng tham gia lớp học của sinh viên.

|  |
| --- |
| **Summative Feedback:**  You achieved:  **Strength:**  **Weakness:**  **How to improve:**  Grade:  Date:  Assessor: |
| **Internal verification:**  **You achieved:**  **Strength:**  •  **Weakness:**  •  **How to improve:**  •  **Grade:**  **Date:**  **Assessor:** |