**BÁO CÁO TỔNG HỢP ĐỀ TÀI**

**\*\*\***

**I. Thông tin nhóm:**

**Nhóm 2 – Lớp SE1805**

|  |  |  |
| --- | --- | --- |
| **STT** | **MSSV** | **Họ và tên** |
| 1 | SE190124 | Lê Tuấn Kiệt |
| 2 | SE192861 | Phạm Tuấn Anh |
| 3 | SE190769 | Đặng Hồng Phước |
| 4 | SE190007 | Trương Thảo Vi |
| 5 | SE184947 | Lê Khắc Nguyên Trung |

**II. Phát biểu bài toán:**

Trung tâm đào tạo **MindX** có nhu cầu phát triển một hệ thống quản lí học tập trực tuyến toàn diện nhằm phục vụ cho các hoạt động dạy và học hiện đại, quản lý học viên hiệu quả, hỗ trợ giảng viên, và tối ưu quy trình vận hành. Để hiện thực hóa điều đó, cần thiết kế và xây dựng một **cơ sở dữ liệu quan hệ** đáp ứng đầy đủ các chức năng nghiệp vụ, đảm bảo tính mở rộng, bảo mật và toàn vẹn dữ liệu.

#### **Yêu cầu chức năng nghiệp vụ chính:**

1. **Quản lý nội dung & khóa học**:

* Giảng viên có thể tạo khóa học, thêm bài học, bài tập, quiz tương ứng.
* Mỗi nội dung có thể kèm video, tài liệu, deadline và tiêu chí chấm điểm.

1. **Đăng ký & theo dõi tiến độ**:
   * + - Học viên đăng ký khóa học, truy cập bài học, làm bài tập, quiz.
       - Hệ thống theo dõi tỷ lệ hoàn thành, lưu điểm, đánh giá và cấp chứng chỉ khi đủ điều kiện.
2. **Thanh toán học phí**:

* Cho phép thanh toán qua nhiều phương thức (chuyển khoản, ví điện tử, thẻ).
* Hỗ trợ mã giảm giá, tạo và quản lý hóa đơn, ghi nhận trạng thái thanh toán.

1. **Tương tác thời gian thực**:
   * + - Học viên và giảng viên tham gia phiên học trực tuyến (live session), điểm danh.
       - Cho phép gửi tin nhắn, phản hồi, và nhận thông báo cá nhân hóa.
2. **Phân quyền & bảo mật**:
   * + - Hệ thống phân vai trò (Admin, Instructor, Student), mỗi vai trò có quyền riêng biệt.
       - Kiểm soát truy cập theo module và hành động, bảo vệ dữ liệu cá nhân.
3. **Kiểm toán & thống kê**:
   * + - Ghi nhận lịch sử truy cập và hành động người dùng (log).
       - Dashboard báo cáo số lượng học viên, khóa học, doanh thu, tiến độ học tập…

#### **2. Yêu cầu hệ thống dữ liệu:**

* Thiết kế **cơ sở dữ liệu chuẩn hóa** đến tối thiểu 3NF để đảm bảo tính toàn vẹn, không dư thừa.
* Sử dụng **khóa chính, khóa ngoại**, ràng buộc toàn vẹn để liên kết dữ liệu chính xác.
* Sử dụng các bảng trung gian để thể hiện quan hệ nhiều-nhiều như ghi danh, phân quyền, nộp bài.
* Tạo **trigger, view, function, stored procedure** hỗ trợ kiểm tra ràng buộc nghiệp vụ, thống kê, và xử lý logic tự động.

**III. Phân tích yêu cầu dữ liệu & chức năng cần có:**

1. **Các yêu cầu về dữ liệu:**

### **a. Độ chính xác và toàn vẹn dữ liệu:**

* Dữ liệu trong các bảng chính như **Users, Courses, Enrollments, Payments...** cần đảm bảo tính chính xác và nhất quán.
* Mỗi bảng sử dụng **khóa chính (primary key)** để định danh duy nhất từng bản ghi.
* **Khóa ngoại (foreign key)** và các ràng buộc toàn vẹn được áp dụng để đảm bảo các mối quan hệ giữa các bảng luôn hợp lệ.
* Các trường quan trọng như email, số tiền, mã định danh phải được kiểm tra định dạng, không để trống và đúng kiểu dữ liệu.

### **b. Bảo mật và phân quyền:**

* Người dùng được phân vai trò thông qua bảng **Roles**, với các quyền tương ứng được định nghĩa trong bảng **Role\_Permissions.**
* Mỗi hành động như tạo khóa học, xét duyệt bài, xem báo cáo chỉ được phép thực hiện bởi người có quyền phù hợp.
* Mật khẩu người dùng phải được mã hóa bảo mật. Các hoạt động như đăng nhập, cập nhật thông tin, xóa nội dung cần được ghi lại trong nhật ký hệ thống **(Activity\_Logs).**

### **c. Dữ liệu động theo thời gian:**

* Cần lưu trữ các dấu thời gian quan trọng như:
  + Ngày học viên đăng ký khóa học **(EnrollDate).**
  + Thời điểm nộp bài **(SubmitTime).**
  + Ngày thanh toán, nhận chứng chỉ, gửi tin nhắn, cập nhật bài học...
* Việc lưu trữ thời gian giúp phân tích hành vi người học, đo lường hiệu suất và theo dõi lịch sử hoạt động.

### **d. Khả năng mở rộng linh hoạt:**

* Cấu trúc dữ liệu phải hỗ trợ dễ dàng mở rộng:
  + Thêm loại bài học mới (video, tài liệu, quiz, khảo sát...)
  + Bổ sung các phương thức thanh toán mới hoặc chính sách giảm giá.
  + Cho phép tích hợp thêm chức năng như đánh giá khóa học, hệ thống phản hồi, kiểm tra đầu vào...
* Thiết kế hướng module, tránh ràng buộc cứng, dễ dàng bảo trì và phát triển trong tương lai.

### **e. Hiệu suất truy vấn:**

* Hệ thống cần đáp ứng hiệu quả khi xử lý: Truy vấn danh sách khóa học, thống kê học viên, kết quả học tập, lịch sử giao dịch...
* Các cột được truy vấn thường xuyên như **UserID, CourseID, Email, LessonID...** cần được đánh chỉ mục (index).
* Dữ liệu lớn cần được xử lý qua phân trang (pagination), lọc và tìm kiếm hiệu quả để đảm bảo hiệu suất và trải nghiệm người dùng.

1. **Các chức năng chính của hệ thống:**

* **Học viên:** đăng ký tài khoản, ghi danh khóa học, học bài, làm bài tập, làm quiz, nhận chứng chỉ, theo dõi tiến độ.
* **Giảng viên:** tạo khóa học, thêm bài học, chấm bài, gửi thông báo, theo dõi học viên.
* **Quản trị viên:** phân quyền người dùng, quản lý danh mục khóa học, theo dõi hoạt động hệ thống.
* **Hệ thống tự động:** tính điểm, kiểm tra hạn nộp bài, gửi thông báo, ghi log hoạt động, tạo hóa đơn và xử lý thanh toán.

**IV. Thiết kế mô hình dữ liệu (ERD & RD):**

Hệ thống cơ sở dữ liệu được xây dựng dựa trên mô hình thực thể – quan hệ (ERD) với quy mô lớn, bao gồm **hơn 30 thực thể chính**, đại diện cho các thành phần thiết yếu trong một nền tảng học trực tuyến. Các thực thể tiêu biểu như người dùng (Users), khóa học (Courses), bài học (Lessons), bài tập (Assignments), bài kiểm tra (Quizzes), chứng chỉ (Certificates), thanh toán (Payments) và diễn đàn thảo luận (DiscussionThreads). Mỗi thực thể đều có hệ thống thuộc tính được xác định rõ ràng, đảm bảo tính nhất quán về kiểu dữ liệu, đầy đủ về thông tin và có khóa chính để định danh duy nhất từng bản ghi.

Các quan hệ giữa các thực thể được xác định đầy đủ, với **hơn 40 mối liên kết** phản ánh nghiệp vụ phức tạp của hệ thống. Các quan hệ được phân loại thành các dạng **một-một, một-nhiều và nhiều-nhiều**. Đối với các quan hệ nhiều-nhiều, mô hình sử dụng các bảng trung gian để quản lý dữ liệu một cách hiệu quả và rõ ràng, chẳng hạn như bảng Enrollments thể hiện mối quan hệ giữa học viên và khóa học, bảng Submissions thể hiện việc học viên nộp bài tập, hoặc bảng RolePermissions quản lý quyền hạn của từng loại người dùng theo vai trò.

Từ sơ đồ ERD, hệ thống được chuyển đổi sang mô hình quan hệ gồm các bảng dữ liệu cụ thể. Mỗi bảng đều được định nghĩa rõ thuộc tính, khóa chính, khóa ngoại và các ràng buộc toàn vẹn. Cụ thể, khóa chính đảm bảo tính duy nhất của mỗi bản ghi, khóa ngoại tạo liên kết giữa các bảng có liên quan, và các ràng buộc toàn vẹn như UNIQUE, CHECK, DEFAULT, hay ON DELETE CASCADE giúp bảo vệ tính hợp lệ và an toàn của dữ liệu trong toàn hệ thống.

Cơ sở dữ liệu đã được chuẩn hóa tối thiểu đến **dạng chuẩn thứ ba (3NF)**. Quá trình chuẩn hóa giúp loại bỏ sự dư thừa dữ liệu, tránh trùng lặp thông tin, và tạo điều kiện thuận lợi cho việc bảo trì hệ thống. Ví dụ, thông tin về quyền người dùng được tách riêng thành bảng Roles thay vì lưu trực tiếp trong bảng Users, hay danh mục khóa học được chuẩn hóa thông qua bảng Categories kết hợp với bảng trung gian CourseCategories. Việc này không chỉ làm rõ cấu trúc dữ liệu mà còn giúp dễ dàng mở rộng về sau nếu hệ thống phát triển thêm chức năng mới.

Bên cạnh đó, hệ thống cũng được tăng cường bằng các **trigger và function** nhằm kiểm soát các quy tắc nghiệp vụ phức tạp, chẳng hạn như kiểm tra hạn nộp bài, tự động cập nhật tiến độ học, hay ghi nhận lịch sử thao tác người dùng. Các ràng buộc và logic này giúp hệ thống hoạt động ổn định, đảm bảo tính toàn vẹn dữ liệu, và hạn chế sai sót khi thao tác.

Nhìn chung, mô hình dữ liệu được thiết kế theo hướng logic, mạch lạc và có tính mở rộng cao. Đây là nền tảng quan trọng giúp hệ thống học trực tuyến vận hành hiệu quả, phục vụ được cả nhu cầu quản lý phức tạp lẫn trải nghiệm người dùng mượt mà.

**V. Sơ lược các đối tượng chính & chức năng:**

1. **Danh sách các đối tượng và thuộc tính:**

|  |  |  |
| --- | --- | --- |
| **STT** | **Đối tượng (Bảng)** | **Các thuộc tính quan trọng** |
| 1 | **Roles** | RoleID PK, RoleName, Description |
| 2 | **Permissions** | PermissionID PK, PermissionKey, Description |
| 3 | **RolePermissions** | RoleID FK, PermissionID FK |
| 4 | **Users** | UserID PK, Email\*, PasswordHash\*, FirstName, LastName, Phone, BirthDate, Gender, RoleID FK, CreatedAt, IsActive |
| 5 | **Centers** | CenterID PK, Name, Address, City, Phone, Latitude, Longitude, IsActive |
| 6 | **Categories** | CategoryID PK, Name, Description |
| 7 | **Courses** | CourseID PK, Title, Slug\*, Description, Level, Language, DurationHours, Price, StartDate, EndDate, CenterID FK, InstructorID FK, ThumbnailURL, IsActive |
| 8 | **CourseCategories** | CourseID FK, CategoryID FK |
| 9 | **CoursePrerequisites** | CourseID FK, PrereqCourseID FK |
| 10 | **Enrollments** | EnrollmentID PK, CourseID FK, StudentID FK, EnrollDate, ProgressPercent, CompletedAt, IsCompleted |
| 11 | **Lessons** | LessonID PK, CourseID FK, Title, Content, VideoURL, DurationMin, OrderIndex |
| 12 | **LessonAttachments** | AttachmentID PK, LessonID FK, FileURL, FileType, UploadedAt |
| 13 | **Assignments** | AssignmentID PK, LessonID FK, Title, Description, MaxScore, DueDate |
| 14 | **Submissions** | SubmissionID PK, AssignmentID FK, StudentID FK, SubmitDate, FileURL, Score, Feedback |
| 15 | **Quizzes** | QuizID PK, LessonID FK, Title, TimeLimitSec |
| 16 | **QuizQuestions** | QuestionID PK, QuizID FK, QuestionTxt, QuestionType, OrderIndex |
| 17 | **QuizOptions** | OptionID PK, QuestionID FK, OptionText, IsCorrect |
| 18 | **QuizAttempts** | AttemptID PK, QuizID FK, StudentID FK, StartedAt, FinishedAt, Score |
| 19 | **LiveSessions** | SessionID PK, CourseID FK, Title, StartTime, EndTime, MeetingURL, CenterID FK |
| 20 | **Attendance** | AttendanceID PK, SessionID FK, StudentID FK, Status, RecordedAt |
| 21 | **Invoices** | InvoiceID PK, StudentID FK, InvoiceDate, TotalAmount, Status |
| 22 | **InvoiceItems** | InvoiceItemID PK, InvoiceID FK, CourseID FK (NULL OK), Description, Amount |
| 23 | **Payments** | PaymentID PK, InvoiceID FK, PaidAmount, PaymentDate, PaymentMethod, TransactionRef |
| 24 | **DiscountCodes** | DiscountCode PK, Description, DiscountPercent, MaxUsage, ExpireAt |
| 25 | **InvoiceDiscounts** | InvoiceID FK, DiscountCode FK |
| 26 | **Certificates** | CertificateID PK, StudentID FK, CourseID FK, IssueDate, CertificateURL |
| 27 | **CourseReviews** | ReviewID PK, CourseID FK, StudentID FK, Rating, Comment, CreatedAt |
| 28 | **Messages** | MessageID PK, SenderID FK, ReceiverID FK, Body, SentAt, IsRead |
| 29 | **Notifications** | NotificationID PK, UserID FK, Title, Message, CreatedAt, IsRead |
| 30 | **ActivityLogs** | LogID PK, UserID FK, Action, Entity, EntityID, Timestamp, Details |

1. **Danh sách các chức năng:**

|  |  |  |
| --- | --- | --- |
| **Mã** | **Chức năng** | **Diễn giải & dữ liệu liên quan** |
| T01 | **Đăng ký tài khoản** | INSERT Users |
| T02 | **Đăng nhập / đổi mật khẩu** | SELECT/UPDATE Users |
| T03 | **Tạo khóa học** | INSERT Courses, Lessons, … |
| T04 | **Cập nhật khóa học** | UPDATE Courses, Lessons |
| T05 | **Đăng ký khóa học** | INSERT Enrollments |
| T06 | **Hủy đăng ký** | DELETE Enrollments |
| T07 | **Xem bài học** | SELECT Lessons, LessonAttachments |
| T08 | **Nộp bài tập** | INSERT Submissions + TRIGGER ghi ActivityLogs |
| T09 | **Chấm điểm** | UPDATE Submissions |
| T10 | **Làm bài quiz** | INSERT QuizAttempts |
| T11 | **Tính tiến độ / hoàn thành** | UPDATE Enrollments.ProgressPercent, IsCompleted |
| T12 | **Phát hành chứng chỉ** | INSERT Certificates |
| T13 | **Tạo hóa đơn** | INSERT Invoices, InvoiceItems |
| T14 | **Thanh toán** | INSERT Payments + UPDATE Invoices.Status |
| T15 | **Áp mã giảm giá** | INSERT InvoiceDiscounts |
| T16 | **Đặt lịch / điểm danh live** | INSERT LiveSessions, Attendance |
| T17 | **Gửi / nhận tin nhắn** | INSERT Messages |
| T18 | **Gửi thông báo** | INSERT Notifications |
| T19 | **Quản lý vai trò & quyền** | INSERT/DELETE RolePermissions |
| T20 | **Ghi log hệ thống** | INSERT ActivityLogs (thông qua trigger hoặc service) |

**VI. Chuẩn hóa & ràng buộc dữ liệu:**

1. **Phân tích chuẩn hóa dữ liệu:**

Hệ thống đã được chuẩn hóa ít nhất đến **dạng chuẩn thứ ba (3NF)** – là mức chuẩn phổ biến trong các hệ thống thực tế vì đảm bảo cân bằng giữa tính tối ưu và hiệu suất.

* **Dạng chuẩn 1 (1NF)** được đảm bảo bằng cách loại bỏ các thuộc tính đa trị hoặc lặp trong bảng. Mỗi thuộc tính tại mỗi bản ghi đều có giá trị nguyên tử, ví dụ như thuộc tính **userEmail, courseTitle, lessonDuration** đều chứa dữ liệu đơn, không bị lặp hoặc phân tách.
* **Dạng chuẩn 2 (2NF)** được thực hiện bằng cách tách các thuộc tính không phụ thuộc toàn phần vào khóa chính. Ví dụ, trong bảng **Enrollments**, các thuộc tính như **ProgressPercent** hoặc **IsCompleted** phụ thuộc đầy đủ vào cặp khóa chính **CourseID** và **StudentID**, chứ không phụ thuộc riêng vào từng phần.
* **Dạng chuẩn 3 (3NF)** được đảm bảo thông qua việc loại bỏ các phụ thuộc bắc cầu giữa các thuộc tính không phải khóa. Ví dụ, thông tin về loại người dùng được tách thành bảng **Roles** riêng biệt, giúp bảng **Users** chỉ lưu khóa ngoại **RoleID**, tránh việc lặp lại tên vai trò ở nhiều bản ghi khác nhau.

Chuẩn hóa như vậy giúp cơ sở dữ liệu giảm thiểu sự trùng lặp thông tin (như tên danh mục, tên giảng viên), từ đó tiết kiệm dung lượng lưu trữ, dễ bảo trì và tránh sai sót trong quá trình cập nhật dữ liệu.

### **Phân tích các ràng buộc dữ liệu:**

Để đảm bảo tính nhất quán và toàn vẹn của dữ liệu trong hệ thống, nhiều **ràng buộc** đã được áp dụng trên các bảng:

* **Khóa chính (PRIMARY KEY):** Giúp định danh duy nhất cho mỗi bản ghi, ví dụ **UserID** trong bảng **Users, CourseID** trong bảng **Courses,** hay **EnrollmentID** trong bảng **Enrollments.**
* **Khóa ngoại (FOREIGN KEY):** Thiết lập mối quan hệ giữa các bảng, giúp ràng buộc tính hợp lệ của dữ liệu liên kết. Ví dụ, **UserID** trong bảng **Enrollments** phải tồn tại trong bảng **Users, CourseID** trong bảng **Lessons** phải khớp với **CourseID** trong bảng **Courses.**
* **Ràng buộc UNIQUE:** Áp dụng cho các thuộc tính như email người dùng, mã khóa học, tên đăng nhập để đảm bảo không bị trùng lặp.
* **Ràng buộc CHECK:** Được sử dụng để giới hạn giá trị hợp lệ của thuộc tính. Ví dụ: giới tính chỉ cho phép giá trị ‘Male’, ‘Female’, hoặc ‘Other’; điểm bài tập phải nằm trong khoảng từ 0 đến 100.
* **Ràng buộc DEFAULT:** Cung cấp giá trị mặc định cho các trường như **RegisterDate, CourseStatus, ProgressPercent**, hỗ trợ quá trình nhập liệu và đảm bảo tính đầy đủ của bản ghi.
* **Ràng buộc NOT NULL:** Áp dụng cho những trường thông tin bắt buộc như tên khóa học, tên người dùng, email... đảm bảo không để trống dữ liệu quan trọng.
* **Hành vi ON DELETE / ON UPDATE:** Trong các ràng buộc khóa ngoại, hệ thống sử dụng các hành vi như ON DELETE CASCADE hoặc SET NULL để xử lý khi bản ghi gốc bị xóa hoặc cập nhật. Điều này tránh gây lỗi tham chiếu và duy trì sự ổn định cho các liên kết dữ liệu.

Ngoài ra, hệ thống còn áp dụng **trigger** và **function** như một hình thức kiểm tra và xử lý nghiệp vụ chuyên sâu, ví dụ như tự động cập nhật tiến độ học tập, tự động cấp chứng chỉ khi hoàn thành khóa học, hoặc ghi nhật ký thao tác của người dùng.

**VI. Xử lý dữ liệu bằng Stored Procedures, Functions, Triggers:**

* **Các Stored Procedure** như sp\_EnrollStudent, sp\_GetCompletedStudentsByCourse, sp\_CountStudentsPerCourse giúp tự động hóa nghiệp vụ ghi danh, thống kê học viên và tiến độ.
* **Các hàm (Function)** như fn\_FullName, fn\_CourseDuration, fn\_AvgAssignmentScore hỗ trợ việc tính toán điểm số, thời lượng học, hiển thị dữ liệu người dùng.
* **Trigger** đảm nhiệm các hành động nghiệp vụ như tự động ghi log khi ghi danh (trg\_LogEnrollment), chặn nộp bài trễ (trg\_CheckAssignmentDeadline), cập nhật tiến độ học tự động (trg\_UpdateProgressAfterLesson).

**VII. Tổng kết:**

Hệ thống đã đáp ứng đầy đủ các yêu cầu nghiệp vụ cho một nền tảng học trực tuyến hiện đại. Cơ sở dữ liệu được tổ chức bài bản, có tính mở rộng và tích hợp cao:

* Đáp ứng đầy đủ luồng nghiệp vụ thực tế từ giảng viên – học viên – quản trị.
* Chuẩn hóa dữ liệu đảm bảo hiệu suất và độ tin cậy.
* Sẵn sàng triển khai thực tế và phát triển thêm giao diện Web/App.