**Kế hoạch sơ bộ**

**1. Phân Tích Yêu Cầu**

* Xác định các yêu cầu chính cho ứng dụng quiz:
  + Nhiều cấp độ câu hỏi (Easy, Medium, Hard)
  + Nhiều chủ đề câu hỏi (SQL, C#, Algorithm, v.v.)
  + Các loại câu hỏi (trắc nghiệm, tự luận, điền đáp án, chọn nhiều đáp án đúng, v.v.)
  + Giao diện người dùng hỗ trợ HTML và rich media (hình ảnh, video, âm thanh)
  + Tạo quiz với thông tin về tiêu đề, nội dung, và đáp án.

**2. Thiết Kế Cơ Sở Dữ Liệu**

* Sử dụng các bảng đã được tạo trước đó để lưu trữ thông tin quiz, câu hỏi, đáp án, người dùng, và mối quan hệ giữa chúng.
* Xác định các ràng buộc cần thiết (khóa chính, khóa ngoại).

**3. Tạo Dự Án C#**

* Tạo một dự án mới trong Visual Studio hoặc IDE tương tự:
  + Chọn **ASP.NET Core Web API** hoặc **.NET Core** cho backend.

**4. Cài Đặt Thư Viện và Công Cụ**

* Cài đặt các gói NuGet cần thiết, như:
  + **Entity Framework Core** (nếu bạn quyết định sử dụng nó cho ORM).
  + **MSSQL Server** cho kết nối với SQL Server.
  + **Moq** cho kiểm thử đơn vị (Unit Testing).

**5. Triển Khai Cơ Sở Dữ Liệu**

**6. Xây Dựng Các Mô Hình (Models)**

**7. Triển Khai Controllers**

* Tạo các Controller để xử lý các yêu cầu API:

|  |  |
| --- | --- |
| **Controller** | **Main methods** |
| QuizController | CreateQuiz, GetAllQuizzes, GetQuizById, UpdateQuiz, DeleteQuiz |
| QuestionController | CreateQuestion, GetAllQuestions, GetQuestionById, UpdateQuestion, DeleteQuestion |
| AnswerController | CreateAnswer, GetAnswersByQuestionId, UpdateAnswer, DeleteAnswer |
| UserController | RegisterUser, UpdateUser, AuthenticateUser |
| UserQuizController | StartQuiz, GetUserQuizzes, CompleteQuiz, GetQuizResults |
| UserAnswerController | SubmitAnswer, GetUserAnswersByQuiz, EvaluateAnswer |
| TopicController | CreateTopic, GetAllTopics, UpdateTopic, DeleteTopic |
| LevelController | GetAllLevels, CreateLevel, UpdateLevel, DeleteLevel |

**8. Xây Dựng Repositories và Services**

**Repositories**

Repositories chịu trách nhiệm giao tiếp với cơ sở dữ liệu cho từng thực thể riêng biệt. Mỗi repository sẽ có các phương thức CRUD để truy cập và thao tác dữ liệu trong database.

* **UserRepository**: Quản lý dữ liệu người dùng.
  + AddUser, GetUserById, GetUserByEmail, UpdateUser, DeleteUser
* **QuizRepository**: Quản lý các quiz.
  + AddQuiz, GetQuizById, GetAllQuizzes, UpdateQuiz, DeleteQuiz
* **QuestionRepository**: Quản lý câu hỏi.
  + AddQuestion, GetQuestionById, GetQuestionsByCriteria, UpdateQuestion, DeleteQuestion
* **AnswerRepository**: Quản lý câu trả lời.
  + AddAnswer, GetAnswerById, GetAnswersByQuestionId, UpdateAnswer, DeleteAnswer
* **UserQuizRepository**: Quản lý kết quả làm quiz của người dùng.
  + AddUserQuiz, GetUserQuizById, GetUserQuizzesByUserId, UpdateUserQuiz, DeleteUserQuiz
* **RoleRepository**: Quản lý vai trò của người dùng.
  + AddRole, GetRoleById, GetAllRoles, UpdateRole, DeleteRole
* **UserRoleRepository**: Quản lý quan hệ giữa người dùng và vai trò.
  + AddUserRole, GetUserRoleById, GetRolesByUserId, DeleteUserRole
* **TagRepository**: Quản lý các tag của quiz.
  + AddTag, GetTagById, GetAllTags, UpdateTag, DeleteTag
* **QuizTagRepository**: Quản lý quan hệ giữa quiz và tag.
  + AddQuizTag, GetQuizTagsByQuizId, DeleteQuizTag
* **TypeRepository**: Quản lý các type của câu hỏi.
  + AddType, GetTypeById, GetAllTypes, UpdateType, DeleteType

**Services**

Services chịu trách nhiệm thực hiện logic nghiệp vụ bằng cách phối hợp các repository. Các service này đảm bảo rằng logic nghiệp vụ không bị trùng lặp trong các controller và dễ bảo trì.

**1. QuestionService**

* **Chức năng**: Quản lý các câu hỏi trong hệ thống.
* **Các method chính**:
  + CreateQuestion(Question question): Thêm câu hỏi mới vào cơ sở dữ liệu.
  + GetQuestionById(int id): Lấy câu hỏi theo ID.
  + GetQuestionsByLevel(string level): Lấy danh sách câu hỏi theo mức độ (Beginner, Intermediate, Advanced, v.v.).
  + GetQuestionsByTopic(string topic): Lấy danh sách câu hỏi theo chủ đề (ví dụ: Toán, Lịch sử, Khoa học,...).
  + GetQuestionTypes(): Lấy danh sách loại câu hỏi (như Multiple Choice, True/False,...).
  + UpdateQuestion(Question question): Cập nhật thông tin của một câu hỏi.
  + DeleteQuestion(int questionId): Xóa câu hỏi khỏi cơ sở dữ liệu.

**2. AnswerService**

* **Chức năng**: Quản lý các câu trả lời cho từng câu hỏi.
* **Các method chính**:
  + AddAnswer(Answer answer): Thêm câu trả lời mới cho một câu hỏi.
  + GetAnswerById(int id): Lấy câu trả lời theo ID.
  + GetAnswersByQuestionId(int questionId): Lấy danh sách câu trả lời cho một câu hỏi cụ thể.
  + GetCorrectAnswersForQuestion(int questionId): Lấy danh sách các câu trả lời đúng cho một câu hỏi.
  + UpdateAnswer(Answer answer): Cập nhật nội dung của câu trả lời.
  + DeleteAnswer(int answerId): Xóa câu trả lời.

**3. UserService**

* **Chức năng**: Quản lý thông tin người dùng, bao gồm tài khoản và quyền hạn của họ.
* **Các method chính**:
  + CreateUser(User user): Tạo người dùng mới trong hệ thống.
  + GetUserById(int id): Lấy thông tin người dùng theo ID.
  + GetUsers(): Lấy danh sách toàn bộ người dùng.
  + UpdateUser(User user): Cập nhật thông tin người dùng.
  + DeleteUser(int userId): Xóa người dùng khỏi hệ thống.
  + AuthenticateUser(string username, string password): Xác thực đăng nhập của người dùng.
  + GetUserRoles(int userId): Lấy danh sách các quyền hạn của người dùng (admin, người tạo quiz, người tham gia,...).

**4. QuizService**

* **Chức năng**: Quản lý các hoạt động liên quan đến quiz như tạo quiz, lưu kết quả, và tính điểm.
* **Các method chính**:
  + CreateQuiz(Quiz quiz): Tạo một quiz mới với danh sách câu hỏi.
  + GetQuizById(int quizId): Lấy thông tin quiz theo ID.
  + AddQuestionToQuiz(int quizId, int questionId): Thêm câu hỏi vào quiz.
  + RemoveQuestionFromQuiz(int quizId, int questionId): Xóa câu hỏi khỏi quiz.
  + StartQuiz(int quizId, int userId): Bắt đầu một quiz cho người dùng.
  + SubmitAnswer(int quizId, int questionId, Answer answer): Lưu câu trả lời của người dùng cho một câu hỏi trong quiz.
  + EndQuiz(int quizId, int userId): Kết thúc quiz và tính điểm.
  + GetQuizResults(int quizId, int userId): Lấy kết quả quiz của một người dùng.

**6. TopicService**

* **Chức năng**: Quản lý các chủ đề và gán các câu hỏi vào các chủ đề khác nhau.
* **Các method chính**:
  + CreateTopic(Topic topic): Thêm chủ đề mới.
  + GetTopicById(int id): Lấy thông tin chủ đề theo ID.
  + GetAllTopics(): Lấy danh sách toàn bộ chủ đề.
  + UpdateTopic(Topic topic): Cập nhật thông tin của một chủ đề.
  + DeleteTopic(int topicId): Xóa một chủ đề.
  + GetQuestionsByTopic(int topicId): Lấy các câu hỏi thuộc một chủ đề cụ thể.

**7. LevelService**

* **Chức năng**: Quản lý các mức độ (Beginner, Intermediate, Advanced) và lấy các câu hỏi theo mức độ.
* **Các method chính**:
  + CreateLevel(Level level): Thêm mức độ mới.
  + GetLevelById(int id): Lấy thông tin mức độ theo ID.
  + GetAllLevels(): Lấy danh sách toàn bộ mức độ.
  + UpdateLevel(Level level): Cập nhật thông tin của mức độ.
  + DeleteLevel(int levelId): Xóa một mức độ.
  + GetQuestionsByLevel(int levelId): Lấy các câu hỏi thuộc một mức độ cụ thể.

**8. UserQuizService**

* **Chức năng**: Quản lý các thông tin quiz mà người dùng đã tham gia và lưu tiến trình của người dùng trong quiz.
* **Các method chính**:
  + StartUserQuiz(int userId, int quizId): Khởi tạo thông tin tham gia quiz của người dùng.
  + UpdateUserProgress(int userId, int quizId, int questionId, Answer answer): Cập nhật tiến trình làm quiz.
  + EndUserQuiz(int userId, int quizId): Kết thúc quiz và lưu kết quả cho người dùng.
  + GetUserQuizResults(int userId, int quizId): Lấy kết quả quiz của người dùng.
  + GetUserQuizzes(int userId): Lấy danh sách các quiz mà người dùng đã tham gia.
  + CalculateScore(int quizId, int userId): Tính điểm của người dùng cho quiz sau khi kết thúc.
  + GetUserScores(int userId): Lấy danh sách điểm số của một người dùng qua các quiz.
  + GetLeaderboard(): Lấy bảng xếp hạng những người có điểm cao nhất.
  + AddScore(Score score): Thêm điểm số cho một quiz của người dùng.
  + UpdateScore(Score score): Cập nhật điểm số.
  + DeleteScore(int scoreId): Xóa điểm số khỏi bảng.

**9. Triển Khai Kiểm Thử Đơn Vị (Unit Testing)**

* Tạo các bài kiểm thử đơn vị cho từng phần của ứng dụng:
  + Kiểm thử các phương thức trong QuizService, UserService, v.v.
  + Sử dụng Moq để kiểm thử các phương thức tương tác với cơ sở dữ liệu.

**10. Tài Liệu và Hướng Dẫn Sử Dụng**

* Viết tài liệu hướng dẫn cho người dùng cuối để sử dụng ứng dụng quiz.
* Tạo hướng dẫn cho nhà phát triển để bảo trì và phát triển thêm tính năng mới.

**11. Triển Khai Ứng Dụng**

* Triển khai ứng dụng lên server hoặc cloud service (Azure, AWS, Heroku, v.v.).
* Đảm bảo rằng ứng dụng có thể truy cập từ xa và hoạt động như mong đợi.

**12. Bảo Trì và Cải Tiến**

* Theo dõi ứng dụng sau khi triển khai để phát hiện và sửa lỗi.
* Thu thập phản hồi từ người dùng để cải tiến và phát triển tính năng mới.

**Lưu Ý**

* **Quản lý phiên bản:** Sử dụng Git để quản lý mã nguồn và theo dõi các thay đổi.
* **Bảo mật:** Cân nhắc các biện pháp bảo mật cho API và thông tin người dùng.