Hệ thống có 3 loại người dùng

Guest

Đăng ký

Xem danh sách bài tập

Xem danh sách bài giảng

Xem câu trả lời của bài tập

Xem Danh sách các cuộc thi

Xem danh sách các tài liệu học tập

Xem các bình luận bài tập của người dùng khác

RegularUser

Tất cả các chức năng của guest

Đăng nhập

Quên mật khẩu

Nộp bài tập

Xem lịch sử nộp bài tập của cá nhân

Tham gia cuộc thi

Xem lịch sử thi

Tải tài liệu học tập

Thêm câu trả lời cho bài tập

Admin

Tạo câu hỏi

Tạo lời giải

Tạo cuộc thi

Xem thống kê

Tạo tài liệu học tập

Các danh từ có trong hệ thống

Người dùng -> tạo lớp người dùng

Quản trị viên: kế thừa từ người dùng

RegularUser: kế thừa từ người dùng

Vấn đề -> tạo lớp Problem: Nội dung, Yêu cầu, Độ khó, Chủ đề, Giới hạn thời gian, Giới hạn không gian

Test case -> Đầu vào, đầu ra

Câu trả lời của người dùng => UserAnswer : Tên lời giải, Chủ đề, Độ phức tạp thời gian, Độ phức tạp không gian, Code, Lời giải thích, Cách tiếp cận

Ngôn ngữ lập trình: tên ngôn ngữ, tên hiển thị của ngôn ngữ, tóm tắt về ngôn ngữ

Cuộc thi: Tên, thời gian, danh sách các vấn đề của cuộc thi, Danh sách người đăng ký tham gia cuộc thi

Tài liệu: tên tài liệu, đường link đến tải tài liệu

Mối quan hệ giữa các thực thể

* Một người dùng có thể nộp nhiều vấn đề, một vấn đề có thể được nộp bởi nhiều người dùng nhiều lần tại các thời điểm khác nhau
* Tạo lớp UserSubmission: thời gian, số test case vượt qua, status, người dùng, vấn đề
* Quan hệ giữa Tag và Problem là nhiều – nhiều
* Tạo lớp ProblemTags: TagId, ProblemId
* Quan hệ giữa Tag và UserAnswer : quan hệ nhiều nhiều
* Tạo lớp UserAnswerTags: TagId, UserAnswerId, CreatedDate, IsDeleted, UpdatedDate
* Quan hệ giữa UserAnswer và Problem: người dùng có thể có nhiều câu trả lời cho một thời điểm với các thông tin khác nhau (thời gian, cách làm), một câu trả lời chỉ cho một vấn đề
* Để thông tin vào trong bảng UserAnswer: Id, ProblemId, UserId, title ,….
* Quan hệ giữa cuộc thi và các Problem: một cuộc thi có thể có nhiều ngôn ngữ lập trình, một ngôn ngữ lập trình có thể nằm trong nhiều cuộc thi -> nhiều nhiều
* Tạo lớp ContestProblems để phản ánh mối quan hệ giữa các cuộc thi và các vấn đề
* Một cuộc thi có thể có nhiều người đăng ký tham gia, một người có thể đăng ký nhiều cuộc thi tại nhiều thời điểm khác nhau
* Tạo lớp ContestParticipant
* Khi người dùng tham gia cuộc thi, việc nộp bài sẽ có cách để nhận biết là bài nộp này được nộp cho một cuộc thi => sẽ tạo lớp một UserContestSubmission để lưu thông tin bài nộp của người dùng cho cuộc thi
* Lớp UserContestSubmission bao gồm: Code, UserId, ContestProblemId, số test case vượt qua, thời gian, không gian, ….
* Một Problem có thể có nhiều Solution, một Solution chỉ cho một bài tập. Một Solution có các thông tin: tên Solution, cách tiếp cận, thuật toán, ứng dụng code, Phân tích độ phức tạp, IsHidden (thêm vào cho các người dùng VIP)
* Một Solution có thể có nhiều ứng dụng code với các ngôn ngữ khác nhau -> tạo lớp SolutionImplementations với các thông tin: ProgrammingLanguage, Code, SolutionId