* **IQuizGame**: Là trung tâm điều khiển, gọi các phương thức để bắt đầu, gửi câu trả lời, điều hướng, và kết thúc trò chơi. Nó tương tác với IQuizState để quản lý trạng thái.
* **IQuizState**: Lưu trữ trạng thái trò chơi, bao gồm danh sách câu hỏi (questions), câu trả lời của người dùng (userAnswers), và các thông tin khác (score, startTime, v.v.).
* **IQuestion**: Định nghĩa cấu trúc và hành vi của mỗi câu hỏi, bao gồm options (lựa chọn), correctAnswer (câu trả lời đúng), và handleAnswer() (xử lý câu trả lời).
* **IOption**: Biểu diễn các lựa chọn cho câu hỏi (nếu có), ví dụ: các đáp án trong câu trắc nghiệm.
* **IAnswer**: Lưu trữ câu trả lời của người dùng, với value thuộc kiểu AnswerValue, và được lưu trong IQuizState.userAnswers.

### **Ví dụ minh họa**

Giả sử trò chơi có 3 câu hỏi:

1. Câu trắc nghiệm: "OOP là gì?" (IMultipleChoiceQuestion).
2. Câu điền chỗ trống: "console.log() là \_\_\_" (IFillInTheBlankQuestion).
3. Câu hotspot: "Chọn thuật toán thích hợp cho giải thuật sau" (IHotspotQuestion).

* **Bắt đầu**: IQuizGame.startQuiz() khởi tạo IQuizState, đặt currentQuestionIndex = 0.
* **Câu 1 (Trắc nghiệm)**:
  + Hiển thị câu hỏi và các lựa chọn (IOption[]): ["Paris", "London"].
  + Người dùng chọn "Paris", hệ thống tạo IAnswer: { questionId: "q1", value: { type: 'text', value: "Paris" } }.
  + handleAnswer() kiểm tra, thấy đúng, tăng score và lưu vào userAnswers.
* **Chuyển tiếp**: Người dùng gọi nextQuestion(), currentQuestionIndex tăng lên 1.
* **Câu 2 (Điền chỗ trống)**:
  + Hiển thị ô nhập liệu, người dùng nhập "Jupiter".
  + Tạo IAnswer: { questionId: "q2", value: { type: 'text', value: "Jupiter" } }.
  + handleAnswer() kiểm tra, đúng, tăng score.
* **Câu 3 (Hotspot)**:
  + Hiển thị bản đồ, người dùng nhấp vào tọa độ { x: 100, y: 200 }.
  + Tạo IAnswer: { questionId: "q3", value: { type: 'coordinates', value: { x: 100, y: 200 } } }.
  + handleAnswer() kiểm tra, đúng, tăng score.
* **Kết thúc**: Người dùng gọi endQuiz(), tính điểm cuối (3/3), hiển thị kết quả.