**TEXT2SQL**

**CÁCH THỨC THỰC HIỆN**

# Xử lý câu hỏi người dùng

* **Người dùng nhập câu hỏi bằng ngôn ngữ tự nhiên (ví dụ: "Xem lịch học của tôi").**
* **LLM phân tích câu hỏi, kết hợp với vai trò (sinh viên/giảng viên) và ID người dùng để tạo truy vấn SQL phù hợp.**
* **Truy vấn SQL được kiểm tra và điều chỉnh để đảm bảo tuân thủ quyền truy cập.**

# Phân tích truy vấn SQL bằng LLM

* LLM nhận truy vấn SQL và phân tích để xác định:
* Thao tác: Loại lệnh SQL (SELECT, UPDATE, INSERT, DELETE).
* Bảng: Các bảng được truy cập (ví dụ: Students, Classes, Schedules).
* Cột: Các cột được chọn hoặc sửa đổi (ví dụ: FullName, Email, StudentID).
* LLM hiểu cú pháp SQL, nhận diện từ khóa (SELECT, FROM, JOIN, v.v.) và trả về thông tin dưới dạng cấu trúc (thường là JSON) chứa danh sách bảng, cột và thao tác.
* Ví dụ: Với truy vấn SELECT FullName FROM Students WHERE StudentID = 1, LLM xác định:
* Thao tác: SELECT.
* Bảng: ["Students"].
* Cột: {"Students": ["FullName"]}.
* Nếu LLM gặp lỗi, chương trình dùng phương pháp regex đơn giản (tìm từ khóa như FROM, SELECT) làm dự phòng, nhưng regex kém chính xác hơn.

# Kiểm tra quyền truy cập

* So sánh thông tin từ LLM (bảng, cột, thao tác) với cấu hình quyền trong file JSON.
* Kiểm tra xem:
* Người dùng có được phép truy cập các bảng và cột đã nêu không.
* Thao tác (SELECT, UPDATE, v.v.) có được phép không.
* Thêm điều kiện lọc từ cấu hình JSON (ví dụ: StudentID = {student\_id} cho sinh viên) vào truy vấn để giới hạn dữ liệu.
* Nếu truy vấn vi phạm quyền (ví dụ: truy cập bảng không được phép), từ chối thực thi và báo lỗi.

# Thực thi và trả kết quả

* Truy vấn SQL đã được kiểm tra và điều chỉnh được chạy trên cơ sở dữ liệu.
* Nếu là SELECT, kết quả trả về dưới dạng bảng dữ liệu.
* Nếu là UPDATE/INSERT/DELETE, trả về thông báo trạng thái (thành công hoặc lỗi).
* Kết quả được hiển thị cho người dùng (bảng hoặc thông báo).

**LUỒNG HOẠT ĐỘNG**

**A diagram of a flowchart

AI-generated content may be incorrect.**