

Nhóm 14

Họ và tên	Mã sinh viên	Lớp
Lê Duy An	20213815	IT13
Lý Khánh Duy	20213899	IT13
Nguyễn Quang Hồng Anh	20213648	IT13
Nguyễn Đức Long		IT13

1. Phân tích yêu cầu

- **ChatGPT:**



- Yêu cầu chính: Hệ thống xử lý ngôn ngữ tự nhiên, có khả năng trả lời câu hỏi, tạo văn bản dựa trên yêu cầu người dùng.
- Người dùng: Người sử dụng cá nhân, doanh nghiệp, nhà phát triển tích hợp AI.
- Chức năng:
 - Tương tác với người dùng bằng ngôn ngữ tự nhiên.
 - Đưa ra câu trả lời chính xác và ngữ cảnh.
 - Tích hợp vào các ứng dụng khác thông qua API.

- **Cursor:**



- Yêu cầu chính: Một trình soạn thảo mã nguồn với tính năng gợi ý mã thông minh, tự động hóa lập trình, hỗ trợ gỡ lỗi.
- Người dùng: Lập trình viên, nhà phát triển phần mềm.
- Chức năng:
 - Tự động hoàn thành mã.
 - Gợi ý cải thiện hiệu suất mã.
 - Quản lý thư viện và tích hợp với hệ thống kiểm soát phiên bản.

- **MySQL:**



- Yêu cầu chính: Hệ thống quản lý cơ sở dữ liệu quan hệ.
- Người dùng: Lập trình viên, quản trị viên cơ sở dữ liệu, doanh nghiệp.
- Chức năng:
 - Lưu trữ và truy vấn dữ liệu có cấu trúc.
 - Quản lý truy cập đồng thời.

- Bảo mật và quản lý quyền người dùng.

2. Thiết kế hệ thống

- **ChatGPT:**
 - **Thiết kế kiến trúc:** Mô hình máy học dựa trên Transformer, sử dụng GPU/TPU để xử lý truy vấn.
 - **Giao diện người dùng (UI):** Chatbox trực tiếp hoặc API cho ứng dụng khác.
 - **Tích hợp API:** RESTful API hoặc WebSocket cho tương tác thời gian thực.
- **Cursor:**
 - **Thiết kế frontend:** Sử dụng Electron hoặc ReactJS cho giao diện người dùng.
 - **Backend:** Phân tích mã dựa trên các ngôn ngữ lập trình phổ biến như Python, JavaScript, C++, và Rust.
 - **Gợi ý mã thông minh:** Tích hợp với OpenAI hoặc công cụ AI khác để đưa ra gợi ý mã.
- **MySQL:**
 - **Thiết kế cơ sở dữ liệu:** Lưu trữ dữ liệu quan hệ theo mô hình bảng, có thể mở rộng.
 - **Tối ưu hóa truy vấn:** Sử dụng chỉ số, khóa chính và khóa ngoại để tối ưu hiệu suất.
 - **Hỗ trợ lưu trữ:** Hỗ trợ lưu trữ trên nhiều hệ thống tệp khác nhau (InnoDB, MyISAM).

3. Lập trình và triển khai

- **ChatGPT:**
 - Sử dụng Python và các framework như TensorFlow, PyTorch để xây dựng và huấn luyện mô hình.
 - Triển khai trên nền tảng đám mây với Docker hoặc Kubernetes để mở rộng khả năng xử lý nhiều người dùng đồng thời.
 - Tích hợp API để cung cấp dịch vụ chatbot cho các ứng dụng web và di động.
- **Cursor:**
 - Frontend: HTML, CSS, JavaScript (ReactJS).
 - Backend: Node.js hoặc Python để xử lý gợi ý mã và kiểm tra cú pháp.

- Tích hợp các công cụ phân tích tĩnh như ESLint, Pylint để hỗ trợ việc gợi ý mã chính xác hơn.
- Triển khai như một ứng dụng desktop (Electron) hoặc plugin cho các trình soạn thảo mã hiện có (VS Code).
- **MySQL:**
 - Viết mã quản lý cơ sở dữ liệu bằng C/C++ để đảm bảo hiệu suất cao.
 - Tạo giao diện người dùng quản lý cơ sở dữ liệu qua web (PHP, Python) để dễ dàng quản trị.
 - Đóng gói dưới dạng dịch vụ (service) để dễ dàng triển khai trên các hệ điều hành khác nhau (Linux, Windows).

4. Kiểm thử phần mềm

- **ChatGPT:**
 - **Kiểm thử chức năng:** Đảm bảo chatbot hiểu đúng yêu cầu của người dùng và đưa ra phản hồi phù hợp.
 - **Kiểm thử hiệu suất:** Đo lường thời gian phản hồi và khả năng xử lý đồng thời nhiều truy vấn.
 - **Kiểm thử bảo mật:** Đảm bảo API không bị tấn công DDoS hoặc lỗ hổng bảo mật.
- **Cursor:**
 - **Kiểm thử giao diện người dùng:** Đảm bảo gợi ý mã được đưa ra đúng ngữ cảnh và giao diện thân thiện.
 - **Kiểm thử khả năng gỡ lỗi:** Đảm bảo các tính năng gỡ lỗi không gây ra lỗi hệ thống.
 - **Kiểm thử tích hợp:** Kiểm tra việc tích hợp với các công cụ phát triển khác (Git, GitHub, Docker).
- **MySQL:**
 - **Kiểm thử chức năng:** Đảm bảo các câu truy vấn SQL chạy chính xác và trả về kết quả đúng.
 - **Kiểm thử hiệu suất:** Đo lường khả năng xử lý truy vấn lớn, đồng thời kiểm tra tính ổn định dưới tải cao.
 - **Kiểm thử bảo mật:** Kiểm tra các lỗi tiêm nhiễm SQL (SQL injection), phân quyền người dùng.

5. Quản lý dự án

- **Phương pháp Agile:**

- Chia nhỏ các giai đoạn phát triển thành các **sprint** (2-4 tuần) và liên tục cải tiến.
- **Quản lý backlog:** Theo dõi các tính năng và lỗi cần xử lý qua các công cụ như Jira hoặc Trello.
- **Họp stand-up hàng ngày:** Cập nhật tiến độ công việc, kịp thời phát hiện và giải quyết vấn đề.
- **CI/CD:** Tích hợp liên tục (Continuous Integration) và triển khai liên tục (Continuous Deployment) để giảm thiểu lỗi.

- **Gantt Chart hoặc Kanban:**

- Sử dụng Gantt chart để lập kế hoạch dài hạn cho các tính năng chính.
- Sử dụng Kanban để quản lý tiến độ và ưu tiên các công việc cần hoàn thành.