## STT: 2

## Họ và Tên: Lê Trọng Đức Anh

## Đơn vị: Viettel Solutions

DÀN Ý BÁO CÁO DỰ ÁN – AI Coding Tutor

1. Giới thiệu
   1. Lí do chọn đề tài:

Việc học lập trình thường đòi hỏi khả năng tư duy logic, hiểu sâu về cú pháp, và kỹ năng giải quyết vấn đề. Tuy nhiên, nhiều người học – đặc biệt là người mới bắt đầu – thường cảm thấy choáng ngợp khi tiếp cận với khối lượng kiến thức lớn, thiếu định hướng, và gặp khó khăn khi không có người hướng dẫn trực tiếp. Trong khi đó, sự phát triển mạnh mẽ của các mô hình ngôn ngữ lớn (LLM) đã mở ra cơ hội mới cho việc cá nhân hóa giáo dục. Vì vậy, tôi lựa chọn xây dựng một hệ thống **AI Coding Tutor** – một trợ lý học lập trình thông minh – nhằm giúp người học tiếp cận kiến thức một cách chủ động, hiệu quả và dễ dàng hơn thông qua đối thoại tự nhiên với AI.

* 1. Mục tiêu hệ thống:

Hệ thống hướng đến việc cung cấp cho người học một trải nghiệm học tập tương tác và linh hoạt. Cụ thể, hệ thống sẽ hiển thị dưới dạng chatbot tương tác với người dùng nơi mà họ có thể hỏi bất cứ chủ đề nào liên quan đến coding (tạo bài luyện tập, sửa lỗi và review code). Ngoài ra hệ thống còn hỗ trợ một workspace và terminal để người dùng có thể thực hành trực tiếp những bài luyện tập được tạo ra và AI tutor hoàn toàn có thể truy xuất vào workspace và terminal nó để hỗ trợ va đánh giá người dùng

* 1. Đối tượng sử dụng:
* Người mới bắt đầu học lập trình (beginner)
* Sinh viên ngành CNTT muốn ôn tập kiến thức
* Người đi làm cần học nhanh các khái niệm hoặc công nghệ mới
* Các nền tảng giáo dục muốn tích hợp gia sư AI vào hệ thống học tập
  1. Ý nghĩa thực tiễn:

Dự án **AI Coding Tutor** mang ý nghĩa thiết thực trong bối cảnh nhu cầu học lập trình ngày càng cao nhưng lại thiếu hụt đội ngũ giảng viên, trợ giảng chất lượng. Hệ thống giúp mở rộng khả năng tiếp cận giáo dục chất lượng cao cho mọi đối tượng, giảm chi phí, tăng tính chủ động trong học tập và hỗ trợ cá nhân hóa quá trình học. Bên cạnh đó, việc ứng dụng LLM vào giáo dục thực hành là một hướng đi tiềm năng cho tương lai của công nghệ giáo dục (EdTech).

1. Tổng quan hệ thống:

Hệ thống **AI Coding Tutor** là một nền tảng học lập trình cá nhân hóa sử dụng trí tuệ nhân tạo (AI), cụ thể là các mô hình ngôn ngữ lớn (LLM), nhằm hỗ trợ người học tiếp cận và nắm vững các khái niệm lập trình thông qua các bài giảng tự động được sinh theo yêu cầu. Hệ thống bao gồm các chức năng chính như sau:

* **Sinh bài tập**: hệ thống sẽ tạo ra những bài tập liên quan đến nội dung yêu cầu của người học
* **Hỏi đáp trong lúc làm bài tập**: nếu người dùng có khó khăn trong lúc làm bài, họ có thể tương tác với AI tutor để nhận hỗ trợ
* **Đánh giá bài làm**: hệ thống sẽ đánh giá bài làm của học viên và đưa ra nhận xét
* **Sửa lỗi code**: hệ thống sẽ hỗ trợ sửa lại lỗi trong quá trình làm bài của học viên

1. Phân tích yêu cầu:
   1. Yêu cầu chức năng:

Hệ thống **AI Coding Tutor** cần đáp ứng các chức năng cơ bản sau để đảm bảo hỗ trợ tốt quá trình học lập trình của người dùng:

* + 1. Phân loại yêu cầu người dùng: ta cần phân loại yêu cầu người dùng để có thể phản hồi đúng với mong muốn của người học. Hiện tại có một số yêu cầu được hỗ trợ bởi hệ thống như sau:
* Sinh bài tập
* Đưa ra gợi ý và giải đáp thắc mắc của người học liên quan đến bài tập
* Đánh giá bài làm và đưa ra nhận xét
* Sửa lỗi code cho người dùng
* Những câu hỏi khác: ta sẽ trả lời dưới dạng QA thông thường
  + 1. Sinh bài tập: Từ thông tin yêu cầu của người học, ta sẽ tạo ra bài tập với định dạng như sau:
* Mô tả bài tập
* Input
* Output
* Ví dụ
  + 1. Hỏi đáp trong lúc làm bài tập: trong quá trình làm bài, nếu như có điều gì thắc mắc thì người dùng có thể hỏi AI tutor (ví dụ cấu trúc của dòng for trong python). Lưu ý rằng AI tutor chỉ đưa ra gợi ý chứ không giải bài cho học viên.
    2. Đánh giá bài làm: AI tutor sẽ đánh giá code của người dùng bằng cách truy cập code và đưa ra nhận xét
    3. Sửa lỗi code: trong quá trình code của học viên có thể sẽ có nhiều lỗi và nhiều thứ cần phải cải thiện (cách đặt tên biến, ghi chú hàm,…). AI tutor sẽ có thể truy cập vào code người dùng và sửa lỗi lại cho họ.
    4. Những câu hỏi khác không nằm trong luồng sinh bài tập và hỏi đáp (ví dụ: “Ai là người tạo ra Python?”) sẽ được xử lí thông thường dưới dạng QA
  1. Yêu cầu phi chức năng:
     1. Giao diện thân thiện: Giao diện người dùng cần trực quan, dễ sử dụng và phù hợp với đối tượng học sinh – sinh viên, đặc biệt là những người mới học lập trình.
     2. Phản hồi nhanh: Hệ thống cần xử lý yêu cầu của người dùng một cách nhanh chóng, bao gồm thời gian sinh bài học, thời gian trả lời từ AI chatbot và phản hồi kết quả.
     3. Khả năng mở rộng: Hệ thống cần được thiết kế với khả năng mở rộng, có thể phục vụ đồng thời nhiều người dùng và dễ dàng tích hợp thêm các tính năng mới như: bài học theo cấp độ, tổng hợp tiến trình học, hệ thống gợi ý thông minh, v.v.

1. Thiết kế hệ thống:
   1. Sơ đồ luồng người dùng (User Flow Diagram):

A diagram of a diagram

AI-generated content may be incorrect.

*Luồng hoạt động của người dùng*

* 1. Sơ đồ luồng hệ thống (System Flow Diagram):

A diagram of a flowchart

AI-generated content may be incorrect.

*Sơ đồ hệ thống*

* 1. Thiết kế kiến trúc:
     1. Frontend (Giao diện người dùng):
* Giao diện web
* Gồm các thành phần như: Giao diện chat, workspace để code, terminal
* Giao tiếp với backend qua API.
  + 1. Backend (Server-side logic):
* Phân loại các yêu cầu người dùng.
* Thực hiện logic theo sơ đồ hệ thống.
* Giao tiếp với các thành phần AI (OpenAI API, …).
* Gửi kết quả qua API cho Frontend
  + 1. Model APIs (AI Services):

Sử dụng các model AI hiện đại như GPT-4o hay LlaMA (sẽ thử nghiệm và đánh giá hiệu suất)

1. Công nghệ sử dụng:

* LLM: GPT-4, Claude, RAG (sẽ thử nghiệm và đưa ra đánh giá)
* Backend: FastAPI
* Frontend: React
* Chatbot: Langchain
* Database: MongoDB

1. Hướng phát triển tiếp theo:

* Cá nhân hóa lộ trình học: Tự động điều chỉnh độ khó và nội dung bài học dựa trên lịch sử học tập và kết quả đánh giá của người dùng.
* Phân tích hành vi học tập: Sử dụng AI để theo dõi và phân tích quá trình học, từ đó đưa ra gợi ý học tập, thời điểm cần ôn lại hoặc chuyển chủ đề mới.
* Bổ sung bài tập lập trình thực hành: thêm nhiều dạng bài tập để giúp quá trình học trở nên phong phú hơn (multiple choices, flashcard,…)