**TRƯỜNG ĐẠI HỌC XÂY DỰNG HÀ NỘI**

**KHOA CÔNG NGHỆ THÔNG TIN**



**ĐỒ ÁN**

**PHÁT TRIỂN ỨNG DỤNG ĐA NỀN TẢNG**

**Đề Tài** : **APP SỬ DỤNG LLM**

**Giảng Viên** : Nguyễn Đình Quý

**Nhóm Thực Hiện** : 6

**Lớp 67CS2**

**Thành Viên**:

- Nguyễn Duy Hiếu - 0178167

- Nguyễn Sơn Nam - 0055267

-Phạm Ngọc Minh - 0117867

- Đỗ Văn Duy - 0190167

**I. Giới Thiệu Đề Tài**

Trong kỷ nguyên số, dữ liệu văn bản và ngôn ngữ tự nhiên xuất hiện ở khắp mọi nơi: từ các bài viết trên mạng xã hội, email, văn bản kinh doanh, đến tài liệu kỹ thuật, nội dung học thuật,... Việc hiểu và xử lý ngôn ngữ tự nhiên là một trong những thách thức lớn đối với máy tính trong suốt nhiều thập kỷ. Tuy nhiên, với sự phát triển nhanh chóng của lĩnh vực Trí tuệ nhân tạo (AI), đặc biệt là sự ra đời của các **Mô hình Ngôn ngữ Lớn (LLM - Large Language Models)**, khả năng xử lý ngôn ngữ của máy tính đã đạt được những bước tiến mang tính đột phá.

Các LLM như **GPT-4** , **Gemini** , **Grok**  đều đã chứng minh khả năng hiểu ngữ cảnh, trả lời các câu hỏi phức tạp, viết sáng tạo, thậm chí tham gia đối thoại một cách mạch lạc và tự nhiên. Đây là kết quả của việc huấn luyện trên hàng tỷ từ, sử dụng kiến trúc mạng neuron sâu (transformers), cho phép các mô hình này có thể sinh ra văn bản giống con người một cách đáng kinh ngạc.

Trong bối cảnh đó, nhu cầu tích hợp các LLM vào các ứng dụng thực tế đang trở nên rất cấp thiết. Từ doanh nghiệp đến giáo dục, từ y tế đến truyền thông, các tổ chức đều có nhu cầu triển khai trợ lý AI, chatbot, công cụ hỗ trợ viết lách, hoặc hệ thống hỗ trợ ra quyết định dựa trên ngôn ngữ.

Việc chọn đề tài này không chỉ bắt kịp xu hướng công nghệ mới nhất mà còn mở ra cơ hội để người nghiên cứu:

* **Tiếp cận các mô hình AI hiện đại**.
* **Rèn luyện kỹ năng triển khai ứng dụng thực tế sử dụng AI**.
* **Đóng góp vào hệ sinh thái ứng dụng thông minh tại Việt Nam** trong tương lai.

**II. Mục tiêu của đề tài**

Mục tiêu của đề tài là xây dựng một ứng dụng tích hợp mô hình ngôn ngữ lớn (LLM) để khai thác khả năng xử lý và sinh ngôn ngữ tự nhiên của các mô hình hiện đại như GPT-4 (OpenAI), Gemini (Google), và Grok (xAI). Ứng dụng được thiết kế nhằm hỗ trợ người dùng tương tác với hệ thống một cách tự nhiên, nhanh chóng và hiệu quả thông qua các chức năng như trả lời câu hỏi, viết nội dung, tóm tắt văn bản, hoặc hội thoại đa lượt với trí tuệ nhân tạo. Việc sử dụng nhiều mô hình LLM giúp người dùng có thể lựa chọn linh hoạt theo nhu cầu về độ chính xác, tốc độ phản hồi hoặc phong cách trả lời.

Đề tài hướng đến việc tích hợp các API chính thức của các mô hình nói trên vào hệ thống backend, đồng thời phát triển một giao diện trực quan, thân thiện trên frontend để người dùng dễ dàng tương tác. Ngoài ra, ứng dụng sẽ áp dụng các kỹ thuật tối ưu hóa đầu vào như prompt engineering nhằm nâng cao chất lượng phản hồi từ mô hình, cũng như hỗ trợ lưu trữ và quản lý lịch sử tương tác để cải thiện trải nghiệm người dùng. Thông qua quá trình nghiên cứu, phát triển và thử nghiệm thực tế, đề tài đặt mục tiêu đánh giá hiệu quả sử dụng LLM trong bối cảnh ứng dụng cụ thể, từ đó làm cơ sở để mở rộng và nâng cấp hệ thống trong tương lai.

**III. Chức Năng Chính Của App**

Ứng dụng được thiết kế nhằm tận dụng sức mạnh của các mô hình ngôn ngữ lớn để hỗ trợ người dùng xử lý ngôn ngữ tự nhiên một cách thông minh. Các chức năng chính bao gồm:

**1. Chọn mô hình LLM**

* Cho phép người dùng lựa chọn mô hình muốn sử dụng: GPT-4, Gemini, hoặc Grok.
* Hiển thị thông tin cơ bản về từng mô hình (độ chính xác, tốc độ phản hồi, mức độ sáng tạo,…).
* Giao diện chuyển đổi mô hình linh hoạt theo nhu cầu người dùng.

**2. Gửi câu hỏi và nhận phản hồi**

* Người dùng có thể nhập câu hỏi hoặc yêu cầu bất kỳ dưới dạng văn bản.
* Hệ thống xử lý truy vấn và trả kết quả dựa trên phản hồi từ mô hình đã chọn.
* Hỗ trợ các loại truy vấn như:
  + Câu hỏi kiến thức tổng quát hoặc chuyên sâu.
  + Viết nội dung (đoạn văn, mô tả sản phẩm, bài viết…).
  + Gợi ý ý tưởng sáng tạo.
  + Chuyển đổi văn phong hoặc tóm tắt nội dung dài.

**3. Hội thoại nhiều lượt (multi-turn chat)**

* Lưu ngữ cảnh cuộc trò chuyện để tạo ra hội thoại tự nhiên và liên tục.
* Hỗ trợ nhắn tin qua lại như một trợ lý ảo thực thụ.
* Hiển thị đoạn chat theo dạng tin nhắn người dùng và AI.

**4. Quản lý phiên trò chuyện (chat sessions)**

* Tạo phiên trò chuyện mới.
* Đổi tên, lưu lại và xoá các phiên trò chuyện cũ.
* Dễ dàng xem lại lịch sử tương tác trong từng phiên.

**5. Lưu và xem lại lịch sử trò chuyện**

* Tự động lưu nội dung chat để người dùng tra cứu lại khi cần.
* Hỗ trợ tìm kiếm theo từ khóa trong lịch sử hội thoại.
* Hiển thị thời gian và mô hình sử dụng trong từng tương tác.

**6. Giao diện người dùng trực quan và phản hồi nhanh**

* Thiết kế đơn giản, dễ sử dụng trên máy tính và thiết bị di động.
* Phản hồi tức thì sau khi người dùng gửi truy vấn.
* Giao diện thân thiện, hiện đại, hiển thị rõ ràng phản hồi từ AI.

**IV. Công Nghệ Sử Dụng**

**1. Node.js**

* Là môi trường runtime cho phép chạy mã JavaScript phía server.
* Xây dựng backend nhanh, nhẹ, xử lý nhiều kết nối đồng thời.
* Hỗ trợ phát triển ứng dụng có tính mở rộng cao và hiệu suất tốt.

**2. Express.js**

* Framework web phổ biến, tối giản, chạy trên Node.js.
* Hỗ trợ xây dựng RESTful API để giao tiếp giữa frontend và backend.
* Quản lý routing, middleware giúp xử lý các yêu cầu HTTP dễ dàng.
* Tích hợp tốt với các thư viện khác, dễ mở rộng.

**3. MongoDB & Mongoose**

* MongoDB: Cơ sở dữ liệu NoSQL, lưu trữ dữ liệu dạng tài liệu JSON linh hoạt, phù hợp với dữ liệu phi cấu trúc như lịch sử chat.
* Mongoose: Thư viện ODM (Object Data Modeling) cho MongoDB, giúp định nghĩa schema, validate dữ liệu và thao tác với database dễ dàng trong Node.js.

**4. node-fetch**

* Thư viện HTTP client dùng để gửi request từ Node.js đến các API bên ngoài.
* Gửi các yêu cầu GET, POST,... đến OpenAI, Google AI API một cách hiệu quả.
* Hỗ trợ xử lý bất đồng bộ với Promise, phù hợp cho các thao tác gọi API mạng.

**V. Các Dịch Vụ (AI) Được Sử Dụng**

**1. GPT-4**

* Mô tả: GPT-4 là mô hình ngôn ngữ lớn đa phương thức của OpenAI, nổi bật với khả năng xử lý văn bản và hình ảnh, phù hợp cho các tác vụ như tạo nội dung, dịch thuật, và trả lời câu hỏi phức tạp.
* Tích hợp: Được tích hợp qua OpenAI API, sử dụng API key để gửi yêu cầu HTTP và nhận phản hồi. API này tính phí dựa trên số lượng token xử lý.
* Ứng dụng: Hỗ trợ các tính năng như chatbot thông minh, tạo văn bản sáng tạo, và phân tích dữ liệu văn bản/hình ảnh.
* Ưu điểm: Độ chính xác cao, khả năng xử lý đa dạng, được cập nhật thường xuyên.
* Hạn chế: Chi phí sử dụng cao, yêu cầu tối ưu hóa để giảm độ trễ trong các tác vụ thời gian thực.

**2. Gemini**

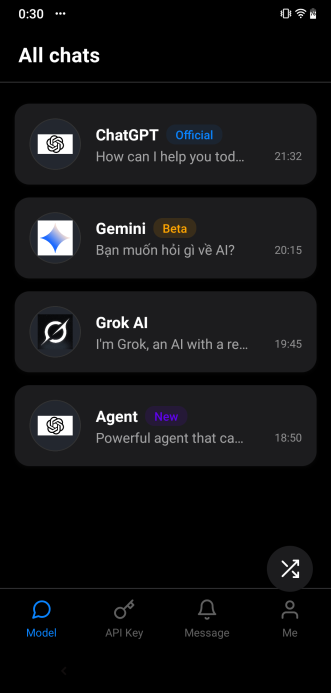
* Mô tả: Gemini là gia đình mô hình AI đa phương thức của Google, hỗ trợ xử lý văn bản, hình ảnh, và các tác vụ tìm kiếm nâng cao. Nó được thiết kế để tối ưu hóa trong hệ sinh thái Google Cloud.
* Tích hợp: Sử dụng qua Google Cloud AI Platform hoặc Vertex AI, với các API endpoint để gọi mô hình và xử lý dữ liệu.
* Ứng dụng: Phù hợp cho tìm kiếm ngữ nghĩa, phân tích hình ảnh, và các tác vụ liên quan đến dữ liệu lớn.
* Ưu điểm: Tích hợp tốt với các công cụ Google, hiệu quả cho các tác vụ phân tích dữ liệu.
* Hạn chế: Phụ thuộc vào hệ sinh thái Google Cloud, có thể phức tạp khi cấu hình ban đầu.

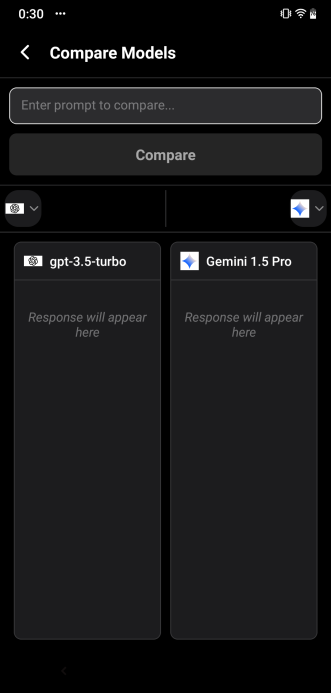
**3. Grok (xAI)**

* Mô tả: Grok, phát triển bởi xAI, là mô hình AI tập trung vào cung cấp câu trả lời trung thực, hữu ích, với góc nhìn khách quan. Grok 3, phiên bản hiện tại, hỗ trợ xử lý văn bản và tìm kiếm web thời gian thực (khi kích hoạt DeepSearch mode).
* Tích hợp: Tích hợp qua Grok API (xem chi tiết tại <https://x.ai/api>). API cho phép gửi yêu cầu và nhận phản hồi văn bản từ Grok.
* Ứng dụng: Dùng để trả lời câu hỏi, phân tích thông tin từ các nguồn công khai (như bài đăng trên X), và hỗ trợ người dùng với các câu hỏi phức tạp.
* Ưu điểm: Câu trả lời ngắn gọn, tập trung vào sự trung thực, khả năng tìm kiếm thời gian thực.
* Hạn chế: Không hỗ trợ BigBrain mode công khai; khả năng đa phương thức hạn chế hơn so với GPT-4 và Gemini.

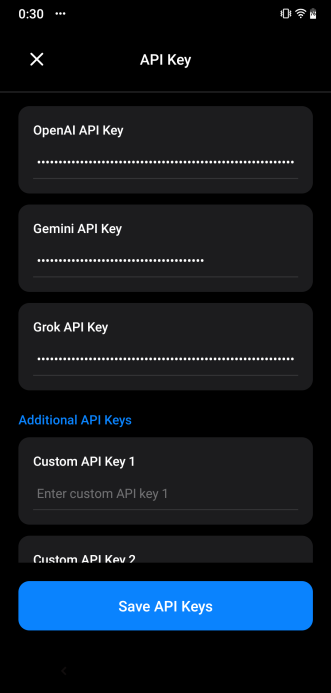
**VI. Hình Ảnh Minh Họa**

Danh sách các cuộc trò chuyện với các mô hình AI như ChatGPT, Gemini Beta, Grok AI, và Agent, thể hiện sự tích hợp đa dạng của các dịch vụ AI trong ứng dụng.

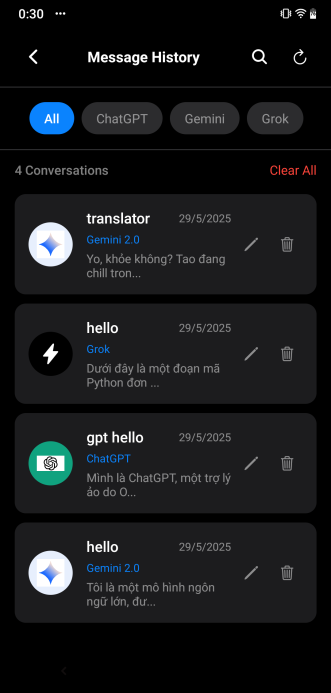


 Tính năng "Compare Models" cho phép so sánh phản hồi giữa hai mô hình AI: GPT-3.5-Turbo (OpenAI) và Gemini 1.5 Pro (Google). Người dùng có thể nhập prompt để nhận kết quả từ cả hai mô hình, minh họa khả năng tích hợp và so sánh hiệu suất của các dịch vụ AI trong ứng dụng.

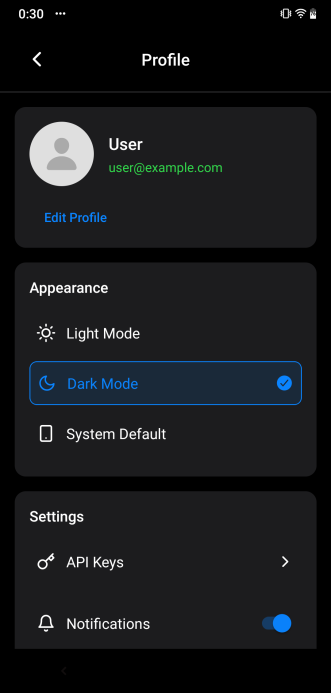
Phần cài đặt API Key cho các dịch vụ AI: OpenAI API Key, Gemini API Key, và Grok API Key, cùng với tùy chọn thêm các Custom API Key. Tính năng này cho phép người dùng nhập và lưu trữ API Key để kết nối với các mô hình AI, đảm bảo tích hợp liền mạch với các dịch vụ AI trong ứng dụng.



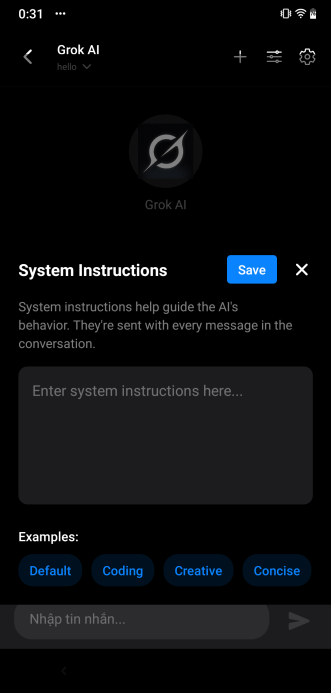
Phần "Message History" với lịch sử 4 cuộc trò chuyện từ ngày 29/05/2025. Các cuộc trò chuyện bao gồm các mô hình AI như Gemini, Grok, và ChatGPT, với các nội dung như dịch thuật, chào hỏi, và câu hỏi về lập trình, cho thấy khả năng đa dạng của ứng dụng trong tương tác với người dùng.



Phần "Profile" của người dùng với thông tin tài khoản "User ([user@example.com](mailto:user@example.com))" và các tùy chọn cá nhân hóa. Giao diện cho thấy chế độ "Dark Mode" được bật, cùng với các cài đặt bổ sung như "API Keys" và "Notifications", thể hiện khả năng tùy chỉnh trải nghiệm người dùng trong ứng dụng.



Phần "System Instructions" của Grok AI. Tính năng này cho phép người dùng nhập hướng dẫn hệ thống để định hình hành vi của Grok AI trong các cuộc trò chuyện, với các tùy chọn mẫu như Default, Coding, Creative, và Concise, hỗ trợ tùy chỉnh trải nghiệm người dùng.



Giao diện thấy tùy chọn "Select Model" với các phiên bản Grok-2 và Grok-3 Beta, thể hiện khả năng lựa chọn mô hình khác nhau để tương tác, với Grok-3 Beta được mô tả là mô hình mới nhất với khả năng suy luận nâng cao, phù hợp cho trò chuyện.



Cuộc trò chuyện với mô hình GPT của OpenAI. ChatGPT tự giới thiệu bằng tiếng Việt ("Minh là ChatGPT, được tạo bởi OpenAI..."), cung cấp thông tin về khả năng hỗ trợ (giải thích, trò chuyện, kể chuyện, v.v.) và hỏi người dùng ("Bạn có thể minh giúp gì không? 😊"), thể hiện tính năng tương tác thân thiện và đa ngôn ngữ.



VII. Agent Báo Cáo Thu Chi

**1.Giới Thiệu**

Chat Agent quản lý tài chính cá nhân là một công cụ hỗ trợ người dùng theo dõi, kiểm soát và tối ưu hóa các khoản thu chi, ngân sách và đầu tư cá nhân thông qua giao tiếp tự nhiên bằng chat. Người dùng có thể dễ dàng nhập liệu, tra cứu thông tin tài chính và nhận tư vấn ngay trong cuộc trò chuyện.

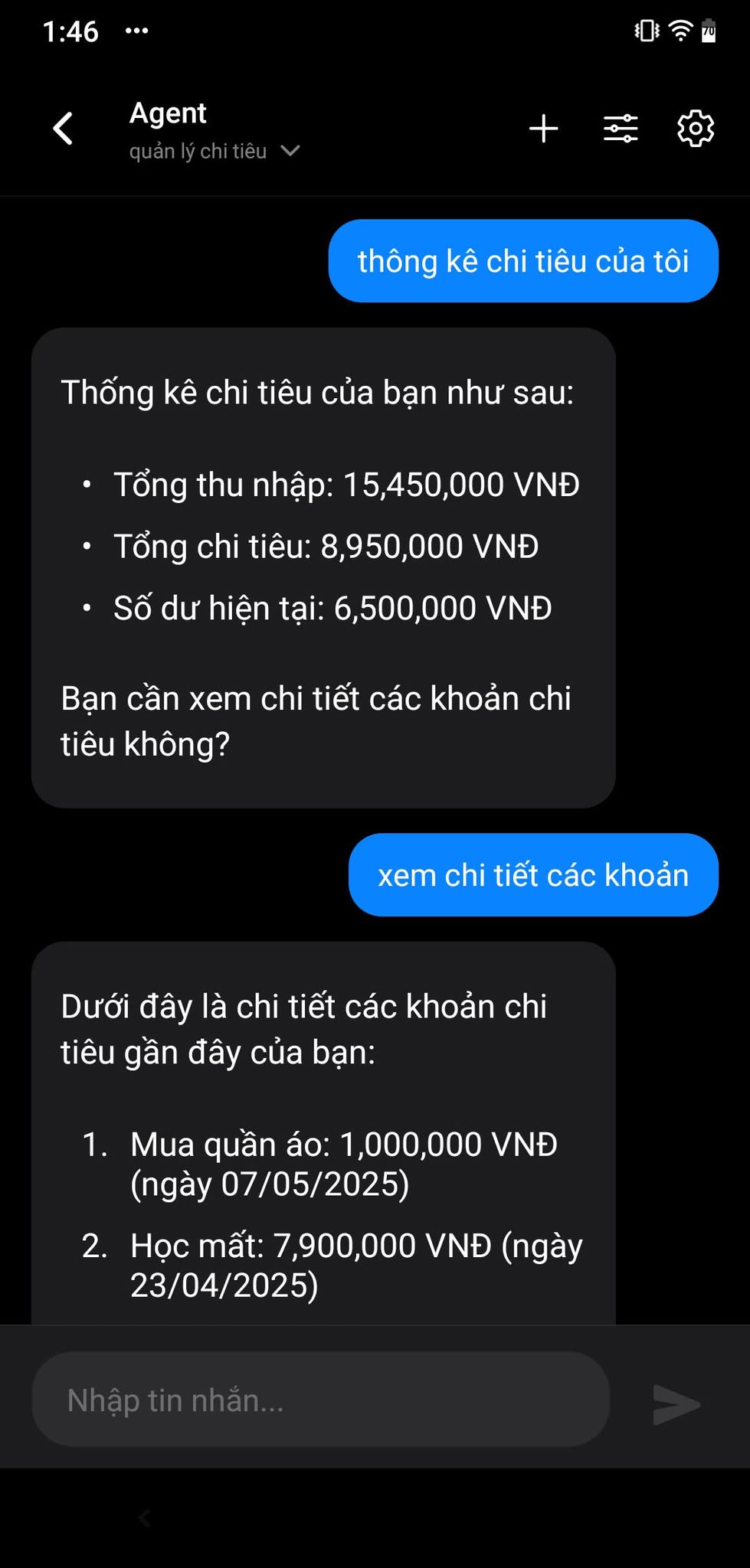
**2.Chức Năng Chính**

* Ghi nhận thu nhập và chi tiêu: Nhập nhanh các khoản thu chi hằng ngày bằng cách chat, ví dụ: “Tôi đã tiêu 200.000 đồng cho ăn uống hôm nay” hoặc “Tôi nhận lương 10 triệu đồng.”
* Phân loại tự động: Chat Agent phân tích và phân loại các khoản chi vào các nhóm như ăn uống, giải trí, học tập, đi lại,...
* Tổng hợp báo cáo: Tạo báo cáo chi tiết theo ngày, tuần, tháng giúp người dùng dễ dàng theo dõi tình hình tài chính.
* Quản lý ngân sách: Đặt và theo dõi ngân sách cho từng mục tiêu, nhận cảnh báo khi sắp vượt ngân sách.
* Tư vấn tài chính cá nhân: Đưa ra lời khuyên dựa trên thói quen chi tiêu, giúp người dùng tiết kiệm và đầu tư hiệu quả.
* Lịch sử giao dịch: Lưu trữ toàn bộ các giao dịch giúp người dùng dễ dàng tra cứu, đối chiếu.

**3.Lợi Ích**

* Giúp người dùng dễ dàng quản lý tài chính cá nhân mà không cần thao tác phức tạp.
* Tăng khả năng kiểm soát chi tiêu và tối ưu hóa ngân sách.
* Cung cấp lời khuyên tài chính cá nhân dựa trên dữ liệu thực tế và thói quen của người dùng.

4. Ví Dụ Minh Họa

KẾT LUẬN

Đề tài "Phát triển ứng dụng đa nền tảng sử dụng LLM" đã được nhóm triển khai bám sát với mục tiêu ban đầu: xây dựng một ứng dụng tích hợp các mô hình ngôn ngữ lớn hiện đại như GPT-4, Gemini và Grok. Ứng dụng đáp ứng nhu cầu xử lý ngôn ngữ tự nhiên của người dùng thông qua nhiều tính năng quan trọng như lựa chọn mô hình, hỏi đáp, hội thoại đa lượt, quản lý lịch sử trò chuyện và so sánh phản hồi giữa các mô hình. Nhóm đã áp dụng các công nghệ phổ biến và phù hợp như Node.js, Express.js, MongoDB, cùng với các thư viện hỗ trợ, để đảm bảo hệ thống hoạt động ổn định, linh hoạt và dễ mở rộng.

Quá trình thực hiện đề tài cho thấy việc tích hợp LLM vào ứng dụng thực tế giúp nâng cao đáng kể trải nghiệm người dùng, cho phép họ tương tác với AI một cách tự nhiên, hiệu quả, đồng thời tận dụng tối đa sức mạnh của từng mô hình cho các nhu cầu khác nhau. Việc thiết kế giao diện trực quan, dễ sử dụng, cùng khả năng tùy chỉnh và cá nhân hóa cũng là điểm mạnh giúp ứng dụng tiếp cận nhiều đối tượng người dùng.

Ngoài các tính năng chính liên quan đến LLM, đề tài còn mở rộng với chức năng quản lý tài chính cá nhân thông qua Agent chat, minh chứng cho khả năng ứng dụng đa dạng của công nghệ AI vào đời sống thực tiễn.

Tuy nhiên, trong quá trình phát triển và thử nghiệm, nhóm cũng ghi nhận một số hạn chế: chi phí sử dụng API cao (đặc biệt với GPT-4), phụ thuộc vào dịch vụ bên thứ ba, và tiềm ẩn rủi ro về bảo mật dữ liệu cá nhân nếu triển khai ở quy mô lớn.

Nhìn chung, đề tài đã hoàn thành các mục tiêu đặt ra, xây dựng được sản phẩm ứng dụng thực tiễn, tạo nền tảng để tiếp tục nghiên cứu, phát triển và mở rộng các chức năng thông minh hơn trong tương lai.