ĐẠI HỌC ĐÀ NẴNG
TRƯỜNG ĐẠI HỌC BÁCH KHOA
KHOA CÔNG NGHỆ THÔNG TIN

ĐỒ ÁN TỐT NGHIỆP

NGÀNH: CÔNG NGHỆ THÔNG TIN

CHUYÊN NGÀNH: AN TOÀN THÔNG TIN

ĐỀ TÀI:

XÂY DỰNG HỆ THỐNG AI HỖ TRỢ NGƯỜI DÂN VỀ THỦ TỤC HÀNH CHÍNH

Người hướng dẫn: **PGS. TS NGUYỄN TẤN KHÔI**
Sinh viên thực hiện: **NGUYỄN NHO GIA HUY**
Mã số sinh viên: **102200134**
Lớp: **20TCLC_DT3**

Đà Nẵng, 01/2026

TÓM TẮT

Tên đề tài: Xây dựng hệ thống AI hỗ trợ người dân về thủ tục hành chính

Sinh viên thực hiện: Nguyễn Nho Gia Huy

Số thẻ SV: 102200134 Lớp: 20TCLC_DT3

Trong bối cảnh chuyển đổi số đang diễn ra mạnh mẽ tại Việt Nam, việc tiếp cận và thực hiện các thủ tục hành chính vẫn còn là thách thức đối với nhiều người dân do thông tin phân tán, khó hiểu và thường xuyên thay đổi. Đề tài **"Xây dựng hệ thống AI hỗ trợ người dân về thủ tục hành chính"** được thực hiện với mục tiêu nghiên cứu, thiết kế và triển khai một hệ thống trí tuệ nhân tạo có khả năng tư vấn, hướng dẫn và hỗ trợ người dân tra cứu thông tin thủ tục hành chính một cách nhanh chóng, chính xác và dễ hiểu.

Hệ thống được xây dựng dựa trên mô hình ngôn ngữ lớn (Large Language Model – LLM) kết hợp với kỹ thuật truy xuất tăng cường dữ liệu (Retrieval-Augmented Generation – RAG), cho phép khai thác hiệu quả kho dữ liệu thủ tục hành chính đã được chuẩn hóa. Kiến trúc hệ thống theo hướng ứng dụng web hiện đại, sử dụng giao diện thân thiện cho người dùng, đồng thời phân quyền rõ ràng cho các nhóm đối tượng như khách truy cập, người dùng đăng ký và quản trị viên.

Kết quả của đề tài cho thấy hệ thống AI có khả năng trả lời các câu hỏi liên quan đến thủ tục hành chính với độ chính xác cao, giảm thời gian tìm kiếm thông tin cho người dân và góp phần nâng cao hiệu quả cung cấp dịch vụ công. Đề tài không chỉ có ý nghĩa về mặt học thuật trong việc ứng dụng AI vào lĩnh vực hành chính công mà còn có giá trị thực tiễn, góp phần hỗ trợ quá trình cải cách hành chính và thúc đẩy chính phủ số tại Việt Nam.

ĐẠI HỌC ĐÀ NẴNG

TRƯỜNG ĐẠI HỌC BÁCH KHOA

KHOA CÔNG NGHỆ THÔNG TIN

CỘNG HÒA XÃ HÔI CHỦ NGHĨA VIỆT NAM

Độc lập - Tự do - Hạnh phúc

NHIỆM VỤ ĐỒ ÁN TỐT NGHIỆP

Họ tên sinh viên: Nguyễn Nho Gia Huy Số thẻ sinh viên: 102200134

Lớp: 20TCLC_DT3 Khoa: Công nghệ Thông tin Ngành: Công nghệ thông tin

1. *Tên đề tài đồ án:* Xây dựng hệ thống AI hỗ trợ người dân về thủ tục hành chính.
2. *Đề tài thuộc diện:* ☐ *Có ký kết thỏa thuận sở hữu trí tuệ đối với kết quả thực hiện*
3. *Các số liệu và dữ liệu ban đầu:*

…...……

…...……

…...……

4. *Nội dung các phần thuyết minh và tính toán:*

…...……

…...……

…...……

5. *Các bản vẽ, đồ thị (ghi rõ các loại và kích thước bản vẽ):*

…...……

…...……

…...……

6. *Họ tên người hướng dẫn:* ……..……
7. *Ngày giao nhiệm vụ đồ án:* *……/……./201…..*
8. *Ngày hoàn thành đồ án:* *……/……./201…..*

Đà Nẵng, ngày 21 tháng 01 năm 2026

Trưởng Bộ môn …….. **Người hướng dẫn**

LỜI NÓI ĐẦU

Trong suốt quá trình học tập tại Trường Đại học Bách Khoa - Đại học Đà Nẵng, em đã trải qua một hành trình học hỏi và phát triển không ngừng. Em xin gửi lời cảm ơn chân thành đến tất cả quý thầy cô trong Khoa Công Nghệ Thông Tin, cũng như từ các khoa khác, đã nhiệt tình giảng dạy và hướng dẫn em. Các thầy cô không chỉ truyền đạt kiến thức chuyên môn mà còn khơi dậy trong em tinh thần học tập, sự tự tin và niềm đam mê nghiên cứu, giúp em có đủ năng lực để hoàn thành đồ án tốt nghiệp của mình.

Đặc biệt, em muốn bày tỏ lòng biết ơn sâu sắc tới thầy Nguyễn Tấn Khôi, người đã hướng dẫn và hỗ trợ em hết mình trong suốt quá trình thực hiện đồ án. Từ việc giúp em xác định mục tiêu nghiên cứu, lựa chọn đề tài phù hợp đến việc cung cấp những lời khuyên quý báu, thầy luôn là người đồng hành tin cậy. Thầy đã dành nhiều thời gian gặp gỡ, thảo luận, giải đáp mọi thắc mắc và giúp tôi hoàn thiện đồ án một cách tốt nhất. Sự tận tâm và nhiệt huyết của thầy đã truyền cảm hứng lớn lao và động lực cho em vượt qua những khó khăn, thử thách.

Ngoài ra, em cũng xin gửi lời cảm ơn tới các bạn bè, đồng nghiệp đã chia sẻ kiến thức, kinh nghiệm và hỗ trợ em trong suốt quá trình học tập và nghiên cứu. Sự giúp đỡ và động viên từ mọi người đã giúp em hoàn thành nhiệm vụ một cách hiệu quả và ý nghĩa.

Dù đã cố gắng hết sức, em hiểu rằng không thể tránh khỏi những thiếu sót trong quá trình học tập và thực hiện đồ án. Em rất mong nhận được những ý kiến đóng góp quý báu từ quý thầy cô và các bạn để có thể hoàn thiện hơn kết quả của mình, cũng như rút kinh nghiệm cho công việc trong tương lai.

Một lần nữa, em xin chân thành cảm ơn!

Đà Nẵng, ngày tháng 01 năm 2026

Sinh viên thực hiện

Nguyễn Nho Gia Huy

CAM ĐOAN

Tôi xin cam đoan đồ án tốt nghiệp với đề tài "Xây dựng hệ thống AI hỗ trợ người dân về thủ tục hành chính." dưới sự hướng dẫn của PGS.TS Nguyễn Tấn Khôi là công trình của tôi thực hiện. Không sao chép bất kỳ đồ án nào có sẵn trước đây.

Với tài liệu tham khảo và trích dẫn từ các tài liệu có liên quan được sử dụng trong đồ án đã được nêu rõ ở phần tài liệu tham khảo. Các thông tin, số liệu được nêu trong bài báo cáo đều mang tính trung thực. Nếu sai thì tôi xin chịu hoàn toàn trách nhiệm và chịu mọi kỷ luật từ Thầy/ Cô và Nhà trường đã đề ra.

Đà Nẵng, ngày tháng 01 năm 2026

Sinh viên thực hiện

Nguyễn Nho Gia Huy

MỤC LỤC

DANH SÁCH HÌNH ẢNH

DANH SÁCH CÁC BẢNG

DANH SÁCH CÁC KÝ HIỆU, CHỮ VIẾT TẮT

Từ viết tắt	Diễn giải
AI	Artificial Intelligence
API	Application Programming Interface
REST	Representational State Transfer
CSDL	Cơ sở dữ liệu
CNTT	Công nghệ thông tin
HTTP	Hypertext Transfer Protocol
E-commerce	Electronic Commerce
SEO	Search Engine Optimization
SSR	Server-Side Rendering
CSR	Client-side Rendering
SSG	Static Site Generation
SPA	Single Page Applications
ISR	Incremental Static Regeneration
IoT	Internet of Things

MỞ ĐẦU

1. Phân tích hiện trạng

Trong những năm gần đây, công tác cải cách hành chính và chuyển đổi số tại Việt Nam đã đạt được nhiều kết quả tích cực, đặc biệt là việc triển khai các cổng dịch vụ công trực tuyến ở trung ương và địa phương. Tuy nhiên, trên thực tế, việc tiếp cận và thực hiện các thủ tục hành chính của người dân vẫn còn gặp nhiều khó khăn và hạn chế.

Thứ nhất, **thông tin về thủ tục hành chính còn phân tán và thiếu tính thống nhất**. Mỗi bộ, ngành, địa phương thường công bố thủ tục trên các cổng thông tin khác nhau, với cách trình bày và mức độ chi tiết không đồng đều. Điều này khiến người dân mất nhiều thời gian tìm kiếm, đối chiếu và dễ xảy ra nhầm lẫn.

Thứ hai, **nội dung thủ tục hành chính mang tính pháp lý cao, khó hiểu đối với người không có chuyên môn**. Các văn bản hướng dẫn thường sử dụng nhiều thuật ngữ chuyên ngành, câu chữ phức tạp, gây khó khăn cho người dân trong việc nắm bắt đúng yêu cầu, hồ sơ cần chuẩn bị và quy trình thực hiện.

Thứ ba, **việc hỗ trợ, tư vấn trực tiếp còn hạn chế**. Tại các cơ quan hành chính, số lượng cán bộ tư vấn có hạn, trong khi nhu cầu của người dân lớn, dẫn đến tình trạng quá tải, chờ đợi lâu hoặc không được giải đáp đầy đủ. Các kênh hỗ trợ trực tuyến hiện nay chủ yếu ở dạng tra cứu tĩnh (FAQ, văn bản), thiếu tính tương tác và cá nhân hóa.

Thứ tư, **mức độ ứng dụng công nghệ thông minh trong hỗ trợ người dân còn thấp**. Phần lớn các hệ thống hiện tại chưa khai thác hiệu quả trí tuệ nhân tạo để phân tích nhu cầu, hiểu câu hỏi tự nhiên của người dân và đưa ra câu trả lời phù hợp với từng trường hợp cụ thể.

Từ những hạn chế trên có thể thấy, nhu cầu xây dựng một **hệ thống AI thông minh, có khả năng tư vấn thủ tục hành chính tự động, chính xác và dễ tiếp cận** là hết sức cần thiết. Đây chính là cơ sở thực tiễn quan trọng để triển khai đề tài "Xây dựng hệ thống AI hỗ trợ người dân về thủ tục hành chính".

2. Mục tiêu hệ thống

Mục tiêu tổng quát của đề tài là **xây dựng một hệ thống AI hoàn chỉnh** có khả năng hỗ trợ người dân trong việc tra cứu, tìm hiểu và thực hiện các thủ tục hành chính thông qua giao diện trực tuyến, hoạt động hiệu quả và phù hợp với điều kiện triển khai thực tế tại Việt Nam.

Cụ thể, đề tài hướng tới các mục tiêu sau:

- Xây dựng hệ thống phần mềm cho phép người dùng đặt câu hỏi về thủ tục hành chính bằng **ngôn ngữ tự nhiên tiếng Việt**.
- Ứng dụng các kỹ thuật **xử lý ngôn ngữ tự nhiên và mô hình ngôn ngữ lớn** để hiểu ý định người dùng và sinh câu trả lời.
- Cung cấp câu trả lời chi tiết, bao gồm **trình tự thực hiện, hồ sơ cần chuẩn bị, cơ quan tiếp nhận và lệ phí** (nếu có).
- Thiết kế hệ thống theo hướng **dễ sử dụng, có khả năng mở rộng và triển khai trên nền tảng web**.
- Đánh giá hiệu quả của hệ thống thông qua thực nghiệm và kết quả sử dụng.

3. Tính năng

Đối với người quản lý:

- Quản lý người dùng và phân quyền: Thêm, sửa, xóa tài khoản; phân loại người dùng (quản trị viên, người dùng, khách vãng lai); khóa hoặc mở quyền truy cập khi cần thiết.
- Quản lý nạp tiền và số dư tài khoản: Theo dõi lịch sử nạp tiền của người dùng; kiểm tra số dư; quản lý các giao dịch nạp tiền và xử lý các vấn đề phát sinh liên quan đến thanh toán.
- Quản lý gói dịch vụ AI: Cấu hình và quản lý các mô hình AI trong hệ thống, bao gồm:

    - Mô hình GovAI (trả phí)

  - Mô hình GPT (miễn phí Thiết lập quyền sử dụng mô hình theo từng nhóm người dùng)

- Quản lý dữ liệu và nội dung thủ tục hành chính: Cập nhật, chỉnh sửa, bổ sung các thủ tục hành chính và văn bản pháp lý dùng cho mô hình GovAI.
- Giám sát và thống kê sử dụng: Theo dõi số lượt truy vấn theo từng mô hình (GovAI/GPT), mức tiêu thụ của người dùng, doanh thu từ nạp tiền; xuất báo cáo phục vụ quản lý.

Đối với người dùng:

- Nạp tiền vào tài khoản: Người dùng có thể nạp tiền vào hệ thống để sử dụng các dịch vụ AI trả phí.
- Lựa chọn mô hình AI**:**

  - Sử dụng mô hình GovAI (trả phí)**:** Nhận tư vấn chuyên sâu, chính xác hơn về thủ tục hành chính dựa trên dữ liệu chuyên ngành.
  - Sử dụng mô hình GPT (miễn phí): Tra cứu và hỏi đáp các thông tin thủ tục hành chính ở mức độ cơ bản.

- Tra cứu và hỏi đáp bằng AI**:** Đặt câu hỏi bằng ngôn ngữ tự nhiên, nhận câu trả lời rõ ràng, dễ hiểu theo từng mô hình đã chọn.
- Quản lý và theo dõi lịch sử sử dụng: Xem lịch sử câu hỏi, số lượt sử dụng, chi phí đã tiêu thụ khi dùng mô hình GovAI.
- Quản lý thông tin cá nhân**:** Cập nhật hồ sơ, thay đổi mật khẩu, theo dõi số dư tài khoản.

Đối với người khách vãng lai:

- Sử dụng mô hình GPT miễn phí**:** Khách vãng lai chỉ được phép sử dụng mô hình GPT miễn phí với số lượt truy vấn giới hạn.

- Tra cứu thông tin thủ tục hành chính cơ bản**:** Nhận hướng dẫn tổng quan, không cá nhân hóa sâu.
- Xem giới thiệu và hướng dẫn hệ thống**:** Tìm hiểu về lợi ích của GovAI và các tính năng nâng cao.
- Khuyến khích đăng ký tài khoản**:** Gợi ý người dùng đăng ký để nạp tiền và sử dụng mô hình GovAI trả phí.

CHƯƠNG 1: CƠ SỞ LÝ THUYẾT

1.1 Tổng quan về xử lý ngôn ngữ tự nhiên (Natural Language Processing – NLP)

1.1.1 Khái niệm

Xử lý ngôn ngữ tự nhiên (Natural Language Processing – NLP) là một lĩnh vực thuộc trí tuệ nhân tạo (Artificial Intelligence – AI) và khoa học máy tính, tập trung vào việc nghiên cứu các phương pháp và kỹ thuật giúp máy tính có khả năng **hiểu, phân tích, diễn giải và sinh ra ngôn ngữ của con người** dưới dạng văn bản hoặc lời nói.

NLP đóng vai trò là cầu nối giữa ngôn ngữ tự nhiên của con người và ngôn ngữ máy, cho phép hệ thống máy tính tương tác với người dùng một cách tự nhiên hơn. Các bài toán phổ biến trong NLP bao gồm: phân tích cú pháp, nhận dạng thực thể, phân tích cảm xúc, tóm tắt văn bản, dịch máy và hỏi đáp tự động.

1.1.2 Ưu điểm

Việc ứng dụng NLP mang lại nhiều lợi ích nổi bật, đặc biệt trong các hệ thống tương tác giữa con người và máy tính:

- **Tương tác tự nhiên với người dùng**: Cho phép người dùng đặt câu hỏi bằng ngôn ngữ tự nhiên mà không cần kiến thức kỹ thuật.
- **Tự động hóa xử lý thông tin**: Giảm sự phụ thuộc vào con người trong việc đọc, phân loại và xử lý khối lượng lớn văn bản.
- **Khả năng mở rộng cao**: NLP có thể xử lý đồng thời hàng nghìn đến hàng triệu yêu cầu, phù hợp với các hệ thống phục vụ số lượng lớn người dùng.
- **Cải thiện trải nghiệm người dùng**: Câu trả lời linh hoạt, dễ hiểu và gần với cách giao tiếp của con người.
- **Hỗ trợ đa ngôn ngữ**: Dễ dàng mở rộng sang nhiều ngôn ngữ khác nhau, đặc biệt hữu ích trong bối cảnh toàn cầu hóa.

1.1.3 Ứng dụng

NLP hiện nay được ứng dụng rộng rãi trong nhiều lĩnh vực khác nhau, tiêu biểu như:

- **Chatbot và trợ lý ảo**: Hỗ trợ tư vấn, giải đáp thắc mắc tự động cho người dùng.
- **Tìm kiếm và hỏi đáp thông minh**: Hiểu ý định người dùng để trả về kết quả phù hợp hơn.
- **Phân tích văn bản và tài liệu**: Trích xuất thông tin, phân loại văn bản, phân tích nội dung pháp lý hoặc hành chính.
- **Dịch máy**: Chuyển đổi ngôn ngữ tự động giữa các ngôn ngữ khác nhau.
- **Hệ thống khuyến nghị và cá nhân hóa**: Phân tích nội dung và hành vi người dùng để đưa ra gợi ý phù hợp.

Trong phạm vi đề tài, NLP đóng vai trò cốt lõi trong việc giúp hệ thống AI **hiểu câu hỏi của người dân về thủ tục hành chính**, từ đó đưa ra câu trả lời chính xác, rõ ràng và dễ tiếp cận, góp phần nâng cao hiệu quả cung cấp dịch vụ công và trải nghiệm của người dân.

1.2 Các mô hình GPT của OpenAI

1.2.1 Giới thiệu

OpenAI là một phòng thí nghiệm nghiên cứu trí tuệ nhân tạo của Mỹ bao gồm tổ chức phi lợi nhuận OpenAI Incorporated và công ty con hoạt động vì lợi nhuận OpenAI Limited Partnership. OpenAI tiến hành nghiên cứu AI với mục đích đã tuyên bố là thúc đẩy và phát triển một AI thân thiện.

GPT, hay Generative Pre-trained Transformer, là một loại mô hình học sâu được phát triển bởi OpenAI. Các mô hình GPT sử dụng kiến trúc Transformer, được giới thiệu lần đầu tiên trong bài báo "Attention is All You Need" của Vaswani và cộng sự (2017). Transformer nổi bật với cơ chế Attention, cho phép mô hình tập trung vào các phần quan trọng của dữ liệu đầu vào, cải thiện khả năng học hỏi các mối quan hệ phức tạp trong ngôn ngữ. Dưới đây là các phiên bản chính của OpenAI GPT:

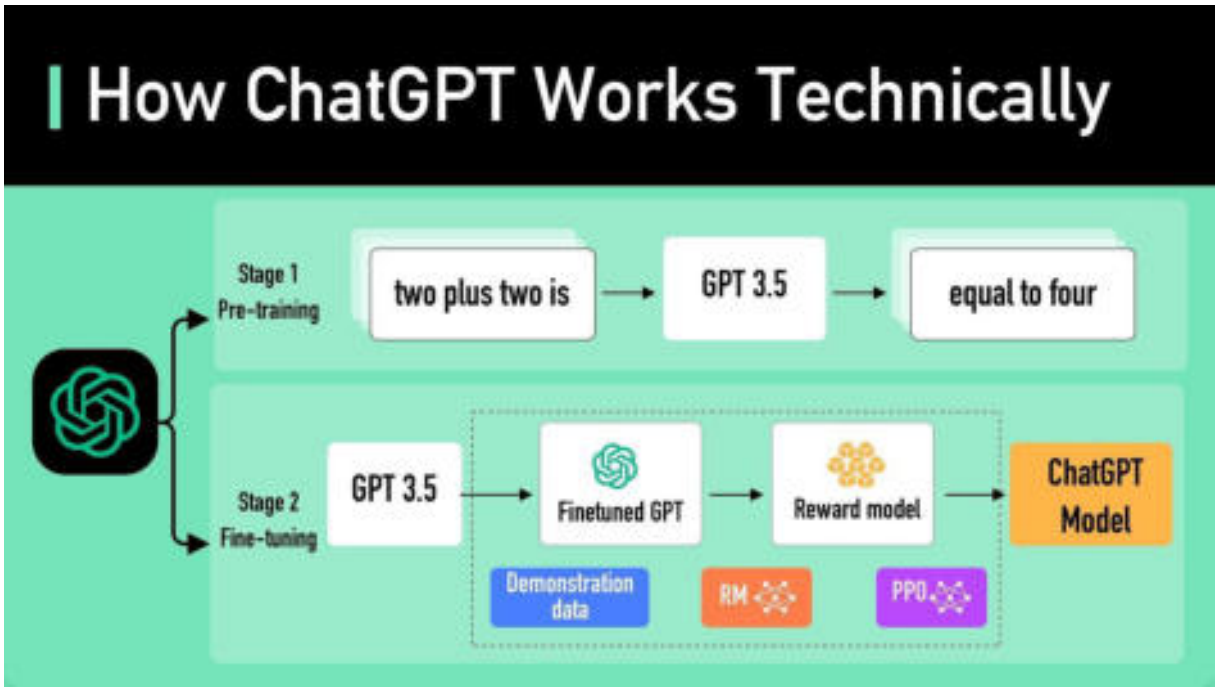
- GPT-1 (2015)
- GPT-2 (2019)

- GPT-3 (2020)
- GPT-4 (2023)
- GPT-5 (2025)

Trong đề tài này sẽ tập trung vào khai thác các ứng dụng của GPT-3, GPT-4.

1.2.2 Nguyên lý hoạt động của một mô hình GPT

Hình ảnh dưới đây thể hiện quá trình huấn luyện GPT.



Hình 1.1 Quá trình huấn luyện mô hình GPT

1.2.2.1 Pre-training (Tiền huấn luyện)

GPT được tiền huấn luyện trên một tập dữ liệu văn bản rất lớn và đa dạng để học cách dự đoán từ tiếp theo trong một chuỗi văn bản.. Các phiên bản như GPT-2 và GPT-3 sử dụng dữ liệu từ nhiều nguồn khác nhau (WebText, Common Crawl, BooksCorpus, Wikimedia,…). Hình 1.2 minh họa tập dữ liệu của GPT.

prompt string · lengths	response string · lengths	source string · classes
Good morning I have a Wpf.DataGrid that is displaying an observable collection of a custom…	One possible solution is to use a fixed width for the GroupItem header and align the header and the…	pacovaldez/stackoverflow-questions
<h2>Hi, How can I generate a pdf with the screen visual data, or generate a pdf of the data being…	To generate a PDF with the screen visual data, you can use a library such as pdf. Here's an example:…	pacovaldez/stackoverflow-questions
<pre><code>package com.kovair.omnibus.adapter.platform; import…	The issue might be related to class loading and garbage collection. When a class loader loads a…	pacovaldez/stackoverflow-questions
I'm trying to get it so that all of the items in ListView.builder() can be displayed on the screen an…	To make the whole page scrollable, remove the `SingleChildScrollView` and wrap the entire…	pacovaldez/stackoverflow-questions
I have used a <code>ListView</code> and the parent in the <code>xml</code> is…	The issue seems to be with the layout parameters being set in the `getView()` method. The code is…	pacovaldez/stackoverflow-questions
I am calling a stored proc [MS SQL] using EF6 from a .net application The call from EF</p…	This is likely due to the fact that CHAR columns are fixed-length and padded with spaces t…	pacovaldez/stackoverflow-questions

Hình 1.2 Minh họa tập dữ liệu prompt-response

Dữ liệu sau khi thu thập được làm sạch (loại bỏ các ký tự không mong muốn như HTML Tags và các văn bản không cần thiết) và được tokenized (token hóa dữ liệu) thành các đơn vị nhỏ hơn gọi là token (thường là từ). Ví dụ:

Văn bản gốc: "Tôi thích học AI."

Sau khi tokenization: ["Tôi", "thích", "học", "AI", "."]

Mỗi token được ánh xạ thành một số nguyên (ID) dựa trên một từ điển (vocabulary) đã được học trước, từ điển này bao gồm tất cả các tokens mà mô hình có thể hiểu và xử lý.

GPT là một mô hình ngôn ngữ tự hồi quy (autoregressive language model), tức là nó dự đoán từ tiếp theo trong một chuỗi dựa trên các từ trước đó. Mô hình học cách sinh văn bản bằng cách tối ưu hóa xác suất có điều kiện của từ tiếp theo.

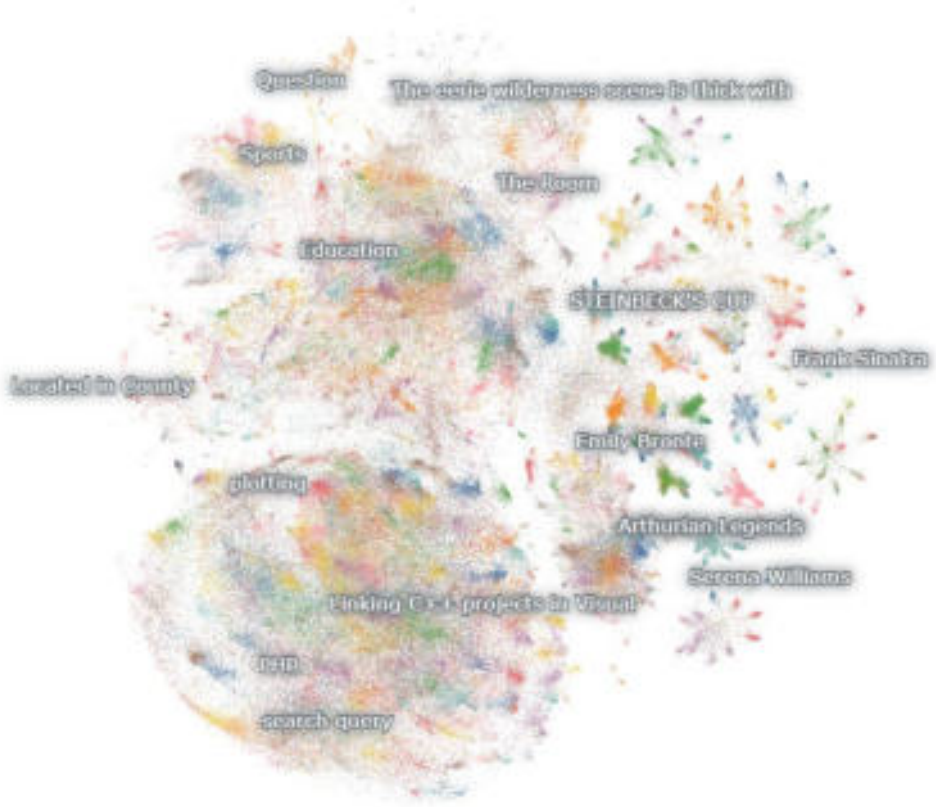
GPT sử dụng kiến trúc Transformer, cụ thể là bộ mã hóa (encoder) trong Transformer. Một số khía cạnh quan trọng của kiến trúc này bao gồm:

- **Attention Mechanism**: Cơ chế Attention cho phép mô hình tập trung vào các phần quan trọng của dữ liệu đầu vào, cải thiện khả năng học và xử lý thông tin.

- **Self-Attention**: Self-attention là một phần của Attention Mechanism, cho phép mô hình tự điều chỉnh trọng số cho từng phần tử của đầu vào dựa trên các phần tử khác trong cùng một chuỗi.

Trong quá trình pre-training, mô hình học cách biểu diễn các từ và câu dưới dạng các vector trong không gian ngữ nghĩa. Các từ có nghĩa tương tự hoặc xuất hiện trong các ngữ cảnh tương tự sẽ có các vector gần nhau trong không gian này. Điều này cho phép mô hình nắm bắt được ngữ nghĩa và mối quan hệ ngữ cảnh của các từ và cụm từ.

Dưới đây là hình ảnh phân cụm được tạo ra bằng cách áp dụng các thuật toán giảm chiều (như t-SNE hoặc UMAP) lên các vector ngữ nghĩa từ mô hình. Kết quả là các từ và cụm từ được phân cụm dựa trên ngữ nghĩa của chúng.



Hình 1.3 Hình ảnh TSNE của dữ liệu huấn luyện GPT4All-J

Hình 1.4 Hình ảnh phóng to của 1.4, với khu vực hiển thị liên quan đến công việc

1.2.2.2 Fine-tuning (Tinh chỉnh)

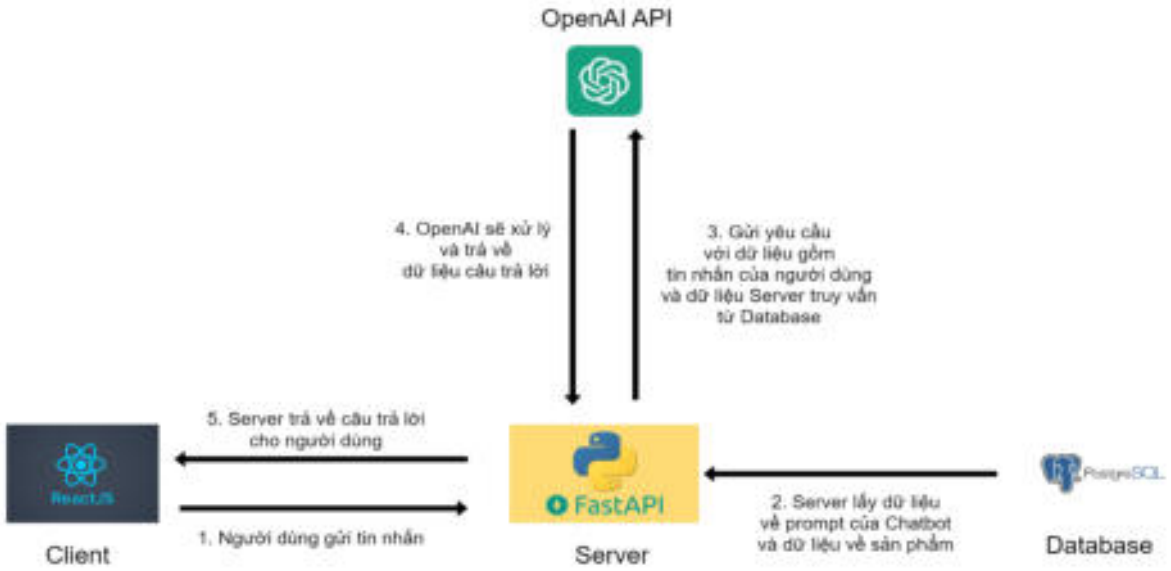
Trong bước fine-tuning, mô hình đã được tiền huấn luyện sẽ được huấn luyện lại trên một tập dữ liệu cụ thể cho một tác vụ đặc thù như phân loại văn bản, dịch ngôn ngữ, hay trả lời câu hỏi. Mục tiêu là tối ưu hóa mô hình cho tác vụ đó, dựa trên những kiến thức ngôn ngữ đã học được từ bước pre-training. Ví dụ, Chatbot Hỗ Trợ Khách Hàng: Sử dụng GPT-3 tiền huấn luyện và fine-tuning với dữ liệu hội thoại của khách hàng từ một công ty cụ thể để xây dựng một chatbot hỗ trợ khách hàng hiệu quả.

Quá trình huấn luyện lại có thể bao gồm các bước:

- Tập Dữ Liệu Cụ Thể: Thu thập và chuẩn bị một tập dữ liệu đại diện cho tác vụ cụ thể, dữ liệu này cần được dán nhãn và định dạng chính xác.
- Sử Dụng Mô Hình Pre-trained: Lấy mô hình đã được tiền huấn luyện (như GPT-2, GPT-3) làm nền tảng. Mô hình này đã học được các đặc điểm ngôn ngữ chung từ một tập dữ liệu lớn và đa dạng.

- Điều Chỉnh Trọng Số: Điều chỉnh trọng số của mô hình bằng cách huấn luyện lại trên tập dữ liệu cụ thể. Trong giai đoạn này, mô hình sẽ học cách thích nghi với đặc điểm của dữ liệu mới và tối ưu hóa hiệu suất cho tác vụ mục tiêu.
- Learning Rate: Sử dụng một learning rate nhỏ để tránh làm mất các đặc điểm đã học được từ giai đoạn tiền huấn luyện.
- Validation: Sử dụng tập kiểm tra để đánh giá hiệu suất của mô hình sau mỗi epoch huấn luyện. Điều này giúp theo dõi quá trình học và ngăn chặn overfitting.
- Hyperparameter Tuning: Điều chỉnh các siêu tham số (như learning rate, batch size) để tìm ra cấu hình tốt nhất cho mô hình.

Kết thúc quá trình fine-tuning, một mô hình GPT đã được huấn luyện và sẵn sàng sử dụng. Tuy nhiên, việc huấn luyện và vận hành một mô hình GPT đòi hỏi khả năng đáp ứng của phần cứng và hạ tầng khá cao nên trong phạm vi đề tài này sẽ chỉ phân tích mô hình và sử dụng OpenAI API để khai thác, hình vẽ bên dưới thể hiện nguyên lý hoạt động của hệ thống.



Hình 1.5 Sơ đồ nguyên lý hoạt động của hệ thống

1.3 Mô hình Qwen2.5-7B-Instruct

1.3.1 Giới thiệu

Alibaba Cloud (thuộc Tập đoàn Alibaba) là một trong những đơn vị tiên phong trong nghiên cứu và phát triển trí tuệ nhân tạo tại Trung Quốc và trên thế giới. Với mục tiêu xây dựng các mô hình AI mạnh mẽ, mở và dễ tiếp cận cho cộng đồng, Alibaba đã giới thiệu dòng mô hình ngôn ngữ lớn **Qwen (Tongyi Qianwen)**.

Qwen2.5-7B-Instruct là một mô hình ngôn ngữ lớn (Large Language Model – LLM) thuộc thế hệ Qwen 2.5, có quy mô **7 tỷ tham số**, được huấn luyện và tinh chỉnh chuyên biệt cho các tác vụ **hội thoại và làm theo chỉ dẫn (instruction-following)**. Mô hình được phát hành theo giấy phép mở, cho phép triển khai cục bộ (on-premise) và tùy biến theo nhu cầu.

So với các mô hình GPT thương mại, Qwen2.5-7B-Instruct có ưu điểm là:

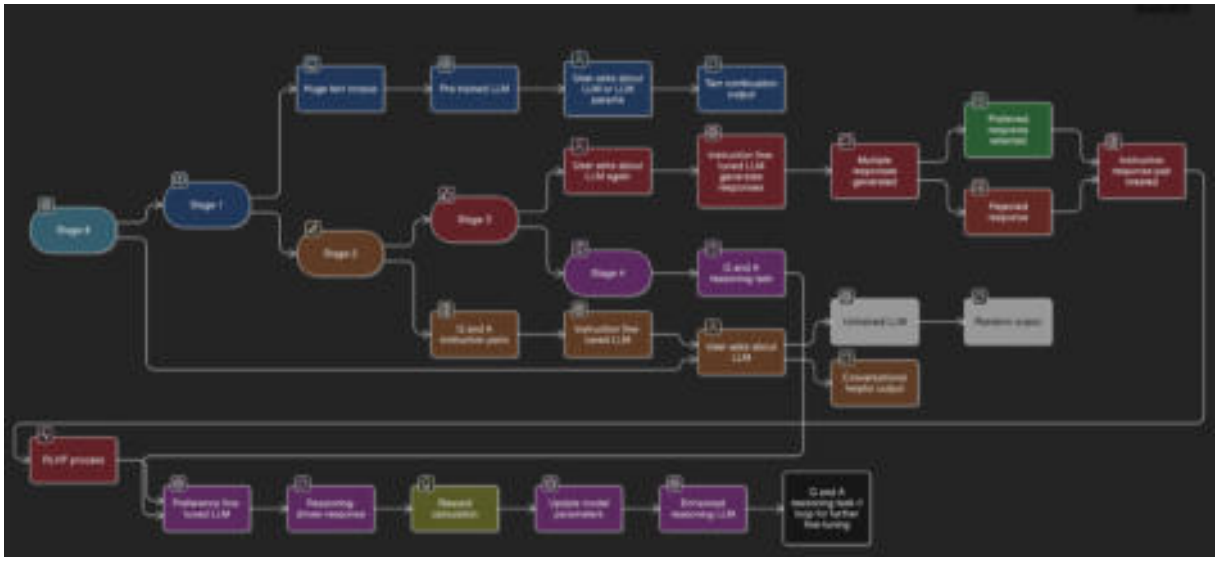
- Có thể **tự triển khai** mà không phụ thuộc hoàn toàn vào API bên thứ ba
- Dễ dàng **fine-tuning** với dữ liệu chuyên ngành
- Phù hợp với các hệ thống yêu cầu **bảo mật dữ liệu** và **kiểm soát chi phí**

Trong phạm vi đề tài, mô hình Qwen2.5-7B-Instruct được lựa chọn để xây dựng **GovAI** – mô hình AI chuyên biệt hỗ trợ người dân về thủ tục hành chính.

Hình 1.6 Logo Qwen2.5

1.3.2 Nguyên lý hoạt động của mô hình Qwen2.5-7B-Instruct

Tương tự các mô hình GPT, Qwen2.5-7B-Instruct được xây dựng dựa trên **kiến trúc Transformer**, sử dụng cơ chế **Self-Attention** để học và biểu diễn ngữ nghĩa ngôn ngữ tự nhiên. Quá trình huấn luyện của mô hình gồm hai giai đoạn chính: **Pre-training** và **Fine-tuning (Instruction tuning)**.



Hình 1.7 Nguyên lý hoạt động của mô hình Qwen2.5-7B-Instruct

1.3.2.1 Pre-training (Tiền huấn luyện)

Trong giai đoạn tiền huấn luyện, mô hình Qwen được huấn luyện trên một tập dữ liệu văn bản rất lớn và đa dạng, bao gồm:

- Dữ liệu web
- Sách điện tử
- Tài liệu học thuật
- Dữ liệu hội thoại
- Mã nguồn và tài liệu kỹ thuật

Mục tiêu của quá trình pre-training là giúp mô hình học cách **dự đoán token tiếp theo** dựa trên ngữ cảnh trước đó (autoregressive language modeling).

Quy trình xử lý dữ liệu bao gồm:

- **Làm sạch dữ liệu**: Loại bỏ HTML tags, nội dung trùng lặp, dữ liệu nhiễu
- **Tokenization**: Chuyển văn bản thành các token
- **Embedding**: Ánh xạ token thành các vector số trong không gian ngữ nghĩa

Ví dụ:

  - Văn bản gốc: "Người dân cần chuẩn bị hồ sơ đăng ký hộ khẩu."
  - Sau khi tokenization: ["Người", "dân", "cần", "chuẩn", "bị", "hồ", "sơ", "đăng", "ký", "hộ", "khẩu", "."]

Mỗi token được ánh xạ thành một **ID** trong từ điển (vocabulary). Trong quá trình huấn luyện, mô hình học cách biểu diễn các token dưới dạng vector sao cho các từ/cụm từ có ý nghĩa tương đồng sẽ nằm gần nhau trong không gian vector.

Các thành phần quan trọng trong kiến trúc Transformer của Qwen bao gồm:

- **Self-Attention**: Cho phép mô hình xem xét mối quan hệ giữa các token trong cùng một chuỗi
- **Multi-Head Attention**: Học nhiều kiểu quan hệ ngữ nghĩa song song
- **Feed-Forward Neural Network**: Xử lý phi tuyến để tăng khả năng biểu diễn

Nhờ đó, mô hình có thể nắm bắt được ngữ cảnh dài và các quan hệ phức tạp trong ngôn ngữ.

1.3.2.2 Fine-tuning / Instruction Tuning (Tinh chỉnh theo chỉ dẫn)

Sau khi tiền huấn luyện, Qwen2.5-7B-Instruct tiếp tục được **fine-tuning** trên các tập dữ liệu **instruction-response** (dữ liệu dạng: yêu cầu – câu trả lời) nhằm nâng cao khả năng:

- Hiểu yêu cầu của người dùng
- Trả lời đúng mục tiêu
- Tuân thủ chỉ dẫn

Quá trình fine-tuning bao gồm:

- **Tập dữ liệu chuyên biệt**: Các cặp câu hỏi – trả lời, hội thoại, dữ liệu nghiệp vụ
- **Sử dụng mô hình pre-trained** làm nền tảng
- **Điều chỉnh trọng số** với learning rate nhỏ để tránh mất kiến thức tổng quát
- **Đánh giá và kiểm tra (validation)** nhằm giảm overfitting

Trong đề tài này, Qwen2.5-7B-Instruct được tinh chỉnh (hoặc kết hợp RAG) với **dữ liệu thủ tục hành chính Việt Nam**, giúp mô hình:

- Hiểu đúng ngữ cảnh hành chính
- Trả lời theo văn phong chuẩn mực
- Hạn chế thông tin sai lệch

Do yêu cầu về phần cứng lớn khi huấn luyện toàn bộ mô hình, hệ thống chủ yếu **khai thác mô hình Qwen2.5-7B-Instruct thông qua fine-tuning nhẹ (LoRA) và Retrieval-Augmented Generation (RAG)** để tối ưu chi phí và hiệu năng.

1.3.3 Vai trò của Qwen2.5-7B-Instruct trong hệ thống GovAI

Trong hệ thống được xây dựng:

- **Qwen2.5-7B-Instruct** đóng vai trò là **mô hình AI trả phí (GovAI)**
- Được sử dụng cho các truy vấn yêu cầu độ chính xác cao về thủ tục hành chính
- Kết hợp với cơ sở dữ liệu và RAG để đảm bảo thông tin có nguồn rõ ràng

Việc sử dụng mô hình Qwen mang lại các lợi ích:

- Chủ động kiểm soát dữ liệu và bảo mật
- Giảm phụ thuộc vào API bên ngoài
- Phù hợp triển khai trong các hệ thống dịch vụ công

1.4 Tổng quan về Flask Framework

1.4.1 Giới thiệu

Flask là một **framework web nhẹ (microframework)** dành cho ngôn ngữ lập trình Python, được thiết kế với mục tiêu đơn giản, linh hoạt và dễ mở rộng. Flask cung cấp các chức năng cốt lõi để xây dựng ứng dụng web và API, đồng thời cho phép lập trình viên chủ động lựa chọn và tích hợp các thư viện mở rộng theo nhu cầu.

Flask hỗ trợ các chức năng cơ bản như: xử lý yêu cầu HTTP, định tuyến (routing), quản lý phiên làm việc (session), xử lý lỗi và trả về dữ liệu dưới nhiều định dạng khác nhau (HTML, JSON, XML,…). Mặc dù là một microframework, Flask vẫn có khả năng xây dựng các hệ thống lớn thông qua hệ sinh thái extension phong phú.

Flask tương thích tốt với các công nghệ hiện đại như:

- RESTful API
- WebSockets
- CORS
- OAuth2, JWT
- Docker và các nền tảng triển khai đám mây

Ngoài ra, Flask hỗ trợ kết nối với nhiều hệ quản trị cơ sở dữ liệu như: SQLite, MySQL, PostgreSQL, MongoDB thông qua các thư viện trung gian (ORM) như SQLAlchemy.

1.4.2 Ưu điểm

Flask được sử dụng rộng rãi trong các hệ thống backend nhờ những ưu điểm nổi bật sau:

- **Nhẹ và đơn giản**:
  Flask có cấu trúc tối giản, dễ học, dễ sử dụng, phù hợp cho cả người mới bắt đầu và các dự án nghiên cứu.
- **Tính linh hoạt cao**:
  Không áp đặt kiến trúc cứng nhắc, cho phép lập trình viên tự do thiết kế cấu trúc dự án và lựa chọn thư viện phù hợp.
- **Dễ mở rộng**:
  Hệ sinh thái extension phong phú như Flask-RESTful, Flask-JWT-Extended, Flask-SQLAlchemy giúp mở rộng chức năng dễ dàng.
- **Phù hợp xây dựng API**:
  Flask được sử dụng phổ biến để xây dựng RESTful API cho các hệ thống web và ứng dụng di động.
- **Cộng đồng lớn và ổn định**:
  Flask có cộng đồng người dùng rộng rãi, tài liệu phong phú và được sử dụng trong nhiều dự án thực tế.
- **Dễ tích hợp với AI/ML**:
  Flask rất phù hợp để triển khai các mô hình AI/ML dưới dạng API, đặc biệt trong các hệ thống nghiên cứu và thử nghiệm.

1.4.3 Ứng dụng

Flask được ứng dụng trong nhiều lĩnh vực khác nhau, tiêu biểu như:

- **Phát triển RESTful API**:
  Xây dựng backend cho website, ứng dụng di động và các hệ thống dịch vụ.

- **Triển khai mô hình AI/ML**:
  Đóng gói các mô hình học máy, học sâu thành API để phục vụ suy luận (inference).
- **Ứng dụng web quy mô nhỏ và vừa**:
  Phù hợp cho các hệ thống có kiến trúc đơn giản hoặc nghiên cứu học thuật.
- **Hệ thống chatbot và AI service**:
  Flask thường được dùng làm backend cho chatbot, hệ thống hỏi đáp và trợ lý ảo.
- **Tích hợp với Docker và Cloud**:
  Dễ dàng container hóa và triển khai trên các nền tảng đám mây.

Trong phạm vi đề tài **"Xây dựng hệ thống AI hỗ trợ người dân về thủ tục hành chính"**, Flask được lựa chọn để xây dựng **backend API** do tính linh hoạt, dễ tích hợp với mô hình AI và phù hợp cho việc triển khai nhanh các dịch vụ hỏi đáp dựa trên trí tuệ nhân tạo.

Hình 1.8 Logo Flask Framework.

1.5 Tổng quan về PostgreSQL Database

1.5.1 Giới thiệu

PostgreSQL là một hệ quản trị cơ sở dữ liệu quan hệ (RDBMS) mạnh mẽ và mã nguồn mở, được phát triển và duy trì bởi một cộng đồng các nhà phát triển toàn cầu. PostgreSQL được biết đến với sự ổn định, tính toàn vẹn, và tuân thủ các chuẩn SQL, hỗ trợ một loạt các tính năng tiên tiến mà các hệ quản trị cơ sở dữ liệu hiện đại cần có.

PostgreSQL có thể được sử dụng trong nhiều loại ứng dụng khác nhau, từ các hệ thống xử lý giao dịch trực tuyến (OLTP) đến các ứng dụng phân tích dữ liệu.

1.5.2 Các tính năng chính

- Tuân thủ ACID: PostgreSQL tuân thủ các nguyên tắc ACID (Atomicity, Consistency, Isolation, Durability), đảm bảo tính toàn vẹn của dữ liệu.
- Hỗ trợ JSON: PostgreSQL cung cấp hỗ trợ mạnh mẽ cho dữ liệu JSON, cho phép lưu trữ và truy vấn dữ liệu dạng tài liệu.
- Mở rộng dễ dàng: Hệ quản trị cơ sở dữ liệu này có khả năng mở rộng thông qua các module và extensions, như PostGIS cho dữ liệu không gian, hoặc các module lập chỉ mục toàn văn.
- Replication và High Availability: PostgreSQL hỗ trợ nhiều cơ chế nhân bản (replication) và các giải pháp sẵn sàng cao (high availability), như Streaming Replication và Logical Replication.
- Hệ thống kiểm soát truy cập mạnh mẽ: PostgreSQL có một hệ thống kiểm soát truy cập chi tiết và mạnh mẽ, cho phép quản trị viên kiểm soát chặt chẽ quyền truy cập của người dùng và vai trò.
- Hỗ trợ các loại dữ liệu đa dạng: PostgreSQL hỗ trợ nhiều loại dữ liệu, bao gồm số, chuỗi, boolean, mảng, các loại dữ liệu địa lý và nhiều loại khác.

1.5.3 Ưu điểm

- Mã nguồn mở: PostgreSQL là mã nguồn mở và miễn phí, cung cấp đầy đủ tính năng mà không cần phí bản quyền.

- Tính mở rộng cao: Với khả năng mở rộng và tích hợp dễ dàng với các hệ thống khác, PostgreSQL phù hợp cho cả các ứng dụng nhỏ lẫn các hệ thống doanh nghiệp lớn.
- Cộng đồng hỗ trợ lớn: PostgreSQL có một cộng đồng phát triển và người dùng lớn mạnh, cung cấp nhiều tài liệu, công cụ và hỗ trợ kỹ thuật.
- Tính năng mạnh mẽ: Với các tính năng tiên tiến như CTEs, bảng tạm thời, khóa ngoại, và các cơ chế lập chỉ mục phong phú, PostgreSQL đáp ứng tốt các yêu cầu phức tạp của ứng dụng hiện đại.
- Hiệu suất cao: PostgreSQL cung cấp hiệu suất cao trong xử lý truy vấn và khả năng tối ưu hóa mạnh mẽ, đảm bảo hiệu quả trong xử lý dữ liệu lớn.

1.5.4 Ứng dụng

- Web Development: PostgreSQL được sử dụng rộng rãi trong phát triển web, nhờ vào khả năng xử lý dữ liệu mạnh mẽ và hỗ trợ tốt cho các frameworks phổ biến như Django, Ruby on Rails, và Spring Boot.
- Data Warehousing: Với các tính năng như hỗ trợ OLAP và khả năng xử lý truy vấn phức tạp, PostgreSQL là một lựa chọn tốt cho các hệ thống kho dữ liệu.
- Ứng dụng doanh nghiệp: PostgreSQL được sử dụng trong nhiều ứng dụng doanh nghiệp yêu cầu tính ổn định, bảo mật và khả năng mở rộng cao.
- Ứng dụng GIS: Với extension PostGIS, PostgreSQL trở thành một hệ quản trị cơ sở dữ liệu mạnh mẽ cho các ứng dụng GIS, hỗ trợ lưu trữ và truy vấn dữ liệu địa lý.

❖ Logo PostgreSQL như hình 1.9:

Hình 1.9 Logo PostgreSQL

1.6 Tổng quan về ReactJS

1.6.1 Giới thiệu

ReactJS là một thư viện JavaScript mã nguồn mở, phát triển bởi Facebook và cộng đồng các nhà phát triển trên toàn thế giới. Được giới thiệu lần đầu tiên vào năm 2011 và trở thành mã nguồn mở vào năm 2013, ReactJS nhanh chóng trở thành một trong những thư viện phổ biến nhất để phát triển các ứng dụng web động.

Hiện nay, ReactJS đã trở thành một trong những thư viện phát triển web phổ biến nhất, được sử dụng rộng rãi bởi các công ty lớn và nhỏ trên toàn thế giới. Facebook cũng tiếp tục đầu tư phát triển và nâng cấp ReactJS để đáp ứng nhu cầu của cộng đồng phát triển.

1.6.2 Các tính năng chính

- Components: ReactJS cho phép phát triển ứng dụng web theo mô hình component. Các component là các phần tử UI độc lập có thể được tái sử dụng trong nhiều phần khác nhau của ứng dụng.

- Virtual DOM: ReactJS sử dụng Virtual DOM để tối ưu hóa hiệu suất của ứng dụng. Virtual DOM là một bản sao của DOM được lưu trữ trong bộ nhớ và được cập nhật một cách nhanh chóng khi có thay đổi, giúp tăng tốc độ và hiệu suất của ứng dụng.
- JSX: JSX là một ngôn ngữ lập trình phân biệt được sử dụng trong ReactJS để mô tả các thành phần UI. JSX kết hợp HTML và JavaScript, giúp cho việc viết mã dễ hiểu và dễ bảo trì hơn.
- State và Props: ReactJS cho phép quản lý trạng thái của các thành phần UI thông qua State và Props. State là trạng thái của một thành phần được quản lý bởi nó chính, trong khi Props là các giá trị được truyền vào từ bên ngoài để tùy chỉnh hoặc điều khiển hành vi của một thành phần.
- Hỗ trợ đa nền tảng: ReactJS không chỉ được sử dụng để phát triển ứng dụng web, mà còn được sử dụng để phát triển ứng dụng di động với React Native. Sử dụng React Native, các nhà phát triển có thể xây dựng ứng dụng di động cho cả iOS và Android sử dụng cùng một mã nguồn.

1.6.3 Ưu điểm

- Hiệu suất cao: ReactJS sử dụng Virtual DOM để tối ưu hóa hiệu suất của ứng dụng. Virtual DOM cho phép ReactJS cập nhật các thay đổi trên trang web một cách nhanh chóng và hiệu quả hơn so với cách truyền thống, giúp tăng tốc độ và hiệu suất của ứng dụng.
- Tái sử dụng: ReactJS cho phép tái sử dụng các thành phần UI, giúp giảm thiểu thời gian và chi phí phát triển. Các thành phần UI có thể được sử dụng lại trong nhiều phần khác nhau của ứng dụng, giúp tăng tính linh hoạt và khả năng mở rộng của ứng dụng.
- Dễ dàng quản lý trạng thái: ReactJS giúp quản lý trạng thái của ứng dụng một cách dễ dàng. Sử dụng State và Props, ReactJS cho phép các nhà phát triển quản lý trạng thái của các thành phần UI một cách chính xác và dễ dàng.
- Hỗ trợ tốt cho SEO: ReactJS cho phép các nhà phát triển xây dựng ứng dụng web với khả năng tương thích tốt với SEO. Với sự hỗ trợ của các thư viện

như React Helmet, ReactJS cho phép các nhà phát triển tùy chỉnh và quản lý các phần tử meta và title cho từng trang web.

- Hỗ trợ đa nền tảng: ReactJS không chỉ được sử dụng để phát triển các ứng dụng web, mà còn được sử dụng để phát triển các ứng dụng di động với React Native. Sử dụng React Native, các nhà phát triển có thể xây dựng ứng dụng di động cho cả iOS và Android sử dụng cùng một mã nguồn.
- Cập nhật liên tục: ReactJS luôn cải tiến với các phiên bản mới, đảm bảo bạn luôn có truy cập vào tối ưu hóa hiệu suất và tính linh hoạt trong phát triển ứng dụng web.

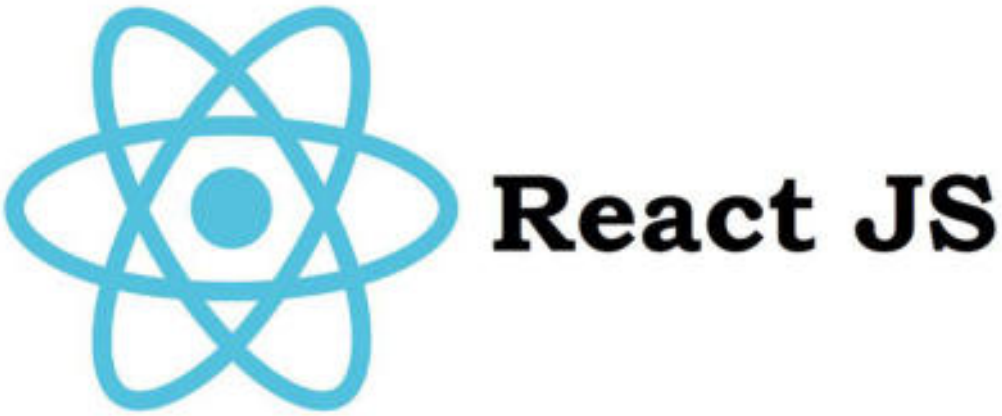
1.6.4 Ứng dụng

- Phát triển web: ReactJS là một lựa chọn phổ biến cho việc phát triển các ứng dụng web hiện đại nhờ vào sự linh hoạt và khả năng tái sử dụng các thành phần (components), giúp việc xây dựng và quản lý giao diện người dùng trở nên dễ dàng hơn.
- Trang tĩnh và động: ReactJS cho phép phát triển các trang web tĩnh với hiệu suất cao và thời gian tải nhanh, cũng như các trang web động với khả năng cập nhật dữ liệu theo thời gian thực thông qua việc sử dụng State và Props.
- Thương mại điện tử (E-commerce): Với khả năng tối ưu hóa hiệu suất và trải nghiệm người dùng, ReactJS là lựa chọn tuyệt vời để xây dựng các cửa hàng trực tuyến, giúp cải thiện trải nghiệm mua sắm và tăng tỷ lệ chuyển đổi. Các thư viện và công cụ đi kèm như React Router, Redux, và Apollo Client giúp quản lý luồng dữ liệu và điều hướng trong các ứng dụng thương mại điện tử.
- Single Page Applications (SPA): ReactJS hỗ trợ Client-Side Rendering (CSR), cho phép xây dựng các ứng dụng trang đơn (SPA) với khả năng tương tác cao và trải nghiệm người dùng mượt mà. Các SPA được xây dựng bằng ReactJS thường có hiệu suất tốt và cảm giác sử dụng như các ứng dụng di động.
- API và dịch vụ backend: Mặc dù ReactJS chủ yếu được sử dụng để xây dựng giao diện người dùng, nhưng nó cũng có thể tích hợp với các API và dịch vụ

backend thông qua các thư viện như Axios hoặc Fetch. Điều này cho phép phát triển các ứng dụng hoàn chỉnh với khả năng giao tiếp dữ liệu hiệu quả giữa frontend và backend.

- Ứng dụng di động: ReactJS là nền tảng của React Native, một framework cho phép phát triển ứng dụng di động cho cả iOS và Android từ một mã nguồn chung. Điều này giúp các nhà phát triển sử dụng cùng một ngôn ngữ và công cụ để xây dựng cả ứng dụng web và di động.
- Ứng dụng IoT và công nghệ mới: ReactJS có thể được sử dụng để xây dựng giao diện người dùng cho các ứng dụng IoT, xử lý dữ liệu thời gian thực và tích hợp với các dịch vụ đám mây và thiết bị IoT. Với khả năng linh hoạt và hiệu suất cao, ReactJS là công cụ mạnh mẽ để phát triển các ứng dụng công nghệ mới.

❖ Logo ReactJS như hình 1.10:



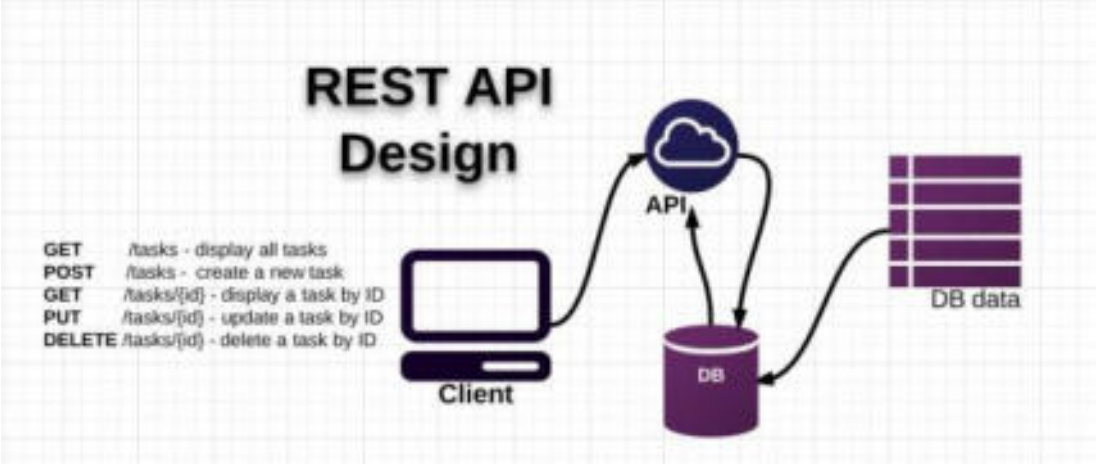
Hình 1.10 Logo ReactJS

1.7 RESTful API là gì?

1.7.1 Giới thiệu

RESTful API [9] là một tiêu chuẩn dùng trong việc thiết kế API cho các ứng dụng web (thiết kế Web services) để tiện cho việc quản lý các resource. Nó chú trọng vào tài nguyên hệ thống (tệp văn bản, ảnh, âm thanh, video, hoặc dữ liệu động…), bao gồm các trạng thái tài nguyên được định dạng và được truyền tải qua HTTP.

❖ Mô hình RESTful API được thể hiện như hình 1.11:



Hình 1.11 Mô hình RESTful API

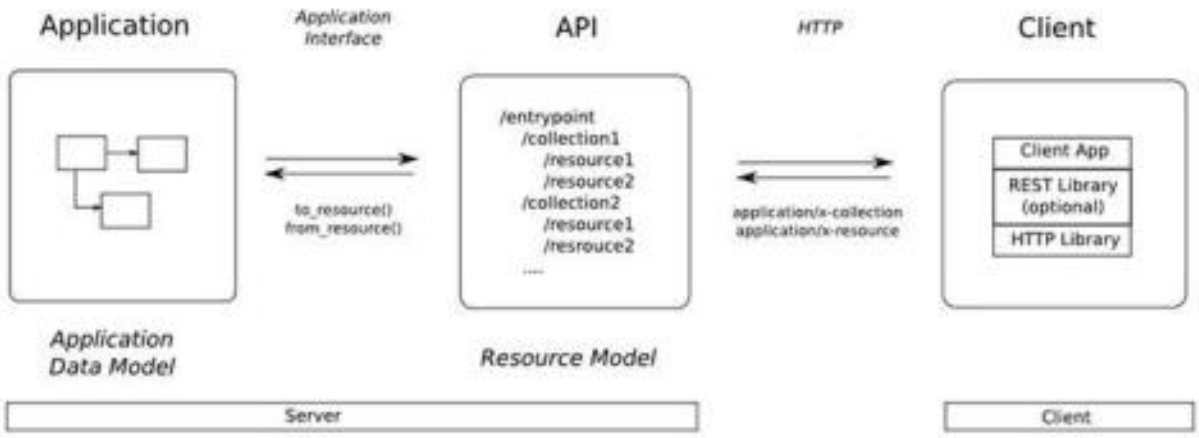
1.7.2 Giải thích các thành phần

- API (Application Programming Interface): Là một tập hợp các quy tắc và cơ chế mà các ứng dụng có thể tuân thủ để giao tiếp với nhau. API giúp các ứng dụng truy cập vào các dịch vụ và tài nguyên của nhau một cách dễ dàng. Dữ liệu được truy cập thông qua API được trả về dưới dạng JSON hoặc XML.
- REST (Representational State Transfer): Là một kiến trúc phần mềm dựa trên giao thức HTTP, giúp xác định cách mà các tài nguyên của hệ thống được truy cập và quản lý. RESTful API sử dụng các phương thức HTTP như GET, POST, PUT, DELETE để thực hiện các thao tác trên tài nguyên.

- RESTful API: Là một API được thiết kế dựa trên các nguyên tắc của REST, giúp quản lý và truy cập vào các tài nguyên của hệ thống một cách dễ dàng và hiệu quả.
- Chức năng quan trọng nhất của REST là quy định cách sử dụng các HTTP method (như GET, POST, PUT, DELETE…) và cách định dạng các URL cho ứng dụng web để quản các resource. RESTful không quy định logic code ứng dụng và không giới hạn bởi ngôn ngữ lập trình ứng dụng, bất kỳ ngôn ngữ hoặc framework nào cũng có thể sử dụng để thiết kế một RESTful API.

1.7.3 RESTful API hoạt động như nào?

❖ Mô hình hoạt động của RESTful API được thể hiện như hình 1.12:



Hình 1.12 Mô hình hoạt động của RESTful API

1.7.4 Authentication và dữ liệu trả về

RESTful API thường sử dụng các phương thức xác thực như OAuth2, JWT để xác thực người dùng và bảo vệ dữ liệu. Dữ liệu trả về từ RESTful API thường được định dạng dưới dạng JSON hoặc XML để dễ dàng xử lý và truyền tải.

❖ Dữ liệu trả về của RESTful API được thể hiện như hình 1.11:

```
{
  "access_token": "eyJhbGciOiJIUzI1NiIsInR5cCI6IkpXVCJ9.
    eyJ1c2VyX2lkIjoiZTAyNzZiYmQtNTYxZC00YTZmLWI2NDUtNWJiMDUxNTRmNTA4IiwiZXhwIjoxNzE4NjEyODU2fQ.
    _PCeqoi1A9BLhEcI8Z1TaaoeSwvDce_jYIsIwqS-hQo",
  "token_type": "bearer",
  "user": {
    "id": "e0276bbd-561d-4a6f-b645-5bb8S154f508",
    "email": "solitudepromotion@gmail.com",
    "display_name": "Solitude",
    "role": "user",
    "avatar_url": "https://i.imgur.com/sZUypfd.png",
    "subscription_id": "8f71b64d-af3b-4bf0-8b5a-ca5515309b31",
    "plan_title": "free",
    "remove_label": false
  }
}
```


Hình 1.13 Dữ liệu trả về của RESTful API

1.7.5 Status code

RESTful API sử dụng các mã trạng thái HTTP để thông báo kết quả của yêu cầu. Các mã trạng thái phổ biến như 200 (OK), 201 (Created), 400 (Bad Request), 401 (Unauthorized), 404 (Not Found), 500 (Internal Server Error)…

1.7.6 HTTP Request

RESTful API sử dụng các phương thức HTTP như GET, POST, PUT, DELETE để thực hiện các thao tác trên tài nguyên. Các phương thức này được sử dụng để thực hiện các thao tác như lấy dữ liệu, tạo mới dữ liệu, cập nhật dữ liệu, xóa dữ liệu.

CHƯƠNG 2: PHÂN TÍCH VÀ THIẾT KẾ HỆ THỐNG

2.1 Phân tích đề tài

2.1.1 Tóm tắt hoạt động của hệ thống mà dự án sẽ được xây dựng:

Hệ thống **AI hỗ trợ người dân về thủ tục hành chính** được xây dựng dưới dạng một ứng dụng web, cho phép người dùng tương tác trực tiếp với trí tuệ nhân tạo nhằm tra cứu, tư vấn và hướng dẫn thực hiện các thủ tục hành chính một cách nhanh chóng, chính xác và thuận tiện. Hệ thống vận hành theo mô hình **client–server**, trong đó giao diện người dùng đảm nhiệm việc hiển thị và tiếp nhận yêu cầu, còn backend chịu trách nhiệm xử lý nghiệp vụ và điều phối các dịch vụ AI.

Về tổng thể, hệ thống bao gồm ba thành phần chính: **Frontend**, **Backend** và **hệ thống mô hình AI**. Người dùng (bao gồm khách vãng lai và người dùng đã đăng ký) có thể đặt câu hỏi liên quan đến thủ tục hành chính thông qua chức năng hỏi đáp AI, hoặc **chủ động tra cứu danh sách các thủ tục hành chính có sẵn trong hệ thống**. Các thủ tục này được phân loại theo lĩnh vực, cơ quan giải quyết và mức độ phổ biến, giúp người dùng dễ dàng tiếp cận và tham khảo.

Bên cạnh đó, hệ thống còn cung cấp **chức năng đọc các bài viết blog về thủ tục hành chính**, trong đó trình bày, phân tích và hướng dẫn chi tiết các thủ tục thường gặp dưới dạng bài viết dễ hiểu. Các bài blog này giúp người dân nắm bắt thông tin trước khi thực hiện thủ tục hoặc trước khi đặt câu hỏi cho hệ thống AI, từ đó nâng cao hiệu quả sử dụng hệ thống.

Đối với **khách vãng lai**, hệ thống chỉ cho phép sử dụng mô hình GPT miễn phí với số lượt truy vấn giới hạn, đồng thời có thể tra cứu thủ tục cơ bản và đọc các bài blog công khai. Đối với **người dùng đã đăng ký**, hệ thống hỗ trợ sử dụng mô hình GPT miễn phí hoặc mô hình **GovAI trả phí**. Khi lựa chọn GovAI, hệ thống sẽ kiểm tra số dư tài khoản, thực hiện trừ tiền tương ứng với mỗi lượt sử dụng trước khi gửi yêu cầu đến mô hình AI.

Trong quá trình xử lý, với mô hình GovAI, hệ thống kết hợp **mô hình ngôn ngữ lớn** và **cơ sở dữ liệu thủ tục hành chính** thông qua cơ chế **Retrieval-Augmented Generation (RAG)** để truy xuất các văn bản liên quan, sau đó sinh câu trả lời dựa trên nguồn dữ liệu đã được kiểm chứng. Kết quả trả về cho người dùng là nội dung hướng dẫn rõ ràng, dễ hiểu và phù hợp với từng trường hợp cụ thể.

Ngoài ra, hệ thống lưu trữ **lịch sử tra cứu, lịch sử hỏi đáp, thông tin sử dụng mô hình và giao dịch nạp tiền** để phục vụ công tác quản lý và thống kê. Quản trị viên có thể theo dõi hoạt động của hệ thống, quản lý người dùng, nội dung thủ tục hành chính, bài blog và đánh giá hiệu quả hoạt động của mô hình AI. Qua đó, hệ thống hướng tới mục tiêu hỗ trợ người dân tiếp cận thủ tục hành chính thuận tiện hơn, giảm thời gian tìm kiếm thông tin và góp phần nâng cao hiệu quả cung cấp dịch vụ công.

2.1.2 Phạm vi dự án được ứng dụng

Dự án **"Xây dựng hệ thống AI hỗ trợ người dân về thủ tục hành chính"** được triển khai với phạm vi ứng dụng tập trung vào việc hỗ trợ tra cứu, tư vấn và hướng dẫn các thủ tục hành chính phổ biến, phù hợp với điều kiện thực tế và thời gian thực hiện của đề tài.

Về **đối tượng sử dụng**, hệ thống hướng tới ba nhóm chính: **khách vãng lai**, **người dùng đã đăng ký** và **người quản lý (quản trị viên)**. Mỗi nhóm đối tượng được phân quyền sử dụng các chức năng khác nhau, đảm bảo tính an toàn, hiệu quả và phù hợp với mục đích khai thác hệ thống.

Về **phạm vi nghiệp vụ**, hệ thống tập trung vào các thủ tục hành chính cơ bản và thường gặp của người dân, bao gồm các lĩnh vực như: hộ tịch, hộ khẩu, căn cước công dân, bảo hiểm, giáo dục, lao động – việc làm và một số thủ tục hành chính công phổ biến khác. Các thủ tục được lưu trữ, phân loại và cập nhật trong cơ sở dữ liệu để phục vụ cả chức năng tra cứu trực tiếp và hỏi đáp bằng AI.

Về **phạm vi công nghệ**, dự án ứng dụng các mô hình ngôn ngữ lớn để xây dựng hệ thống hỏi đáp thông minh. Người dùng đã đăng ký có thể lựa chọn sử dụng mô hình GPT miễn phí hoặc mô hình **GovAI trả phí** (xây dựng trên nền tảng mô hình ngôn ngữ lớn và dữ liệu thủ tục hành chính). Khách vãng lai chỉ được phép sử dụng mô hình GPT miễn phí với số lượt giới hạn. Hệ thống backend được xây dựng theo kiến trúc web hiện đại, có khả năng mở rộng và tích hợp với cơ sở dữ liệu phục vụ lưu trữ người dùng, nội dung thủ tục, blog và lịch sử sử dụng.

Về **chức năng hỗ trợ**, hệ thống cho phép người dùng tra cứu các thủ tục hành chính có sẵn, đọc các bài blog hướng dẫn về thủ tục hành chính, đặt câu hỏi cho AI, lưu lịch sử tra cứu và quản lý thông tin cá nhân. Các chức năng nâng cao như quản lý nạp tiền, quản lý mô hình AI và thống kê hệ thống được giới hạn cho người quản lý.

Về **giới hạn của dự án**, hệ thống chỉ đóng vai trò **tư vấn và hỗ trợ thông tin**, không thay thế cơ quan nhà nước hay có giá trị pháp lý trong việc giải quyết thủ tục hành chính. Thông tin do hệ thống cung cấp mang tính tham khảo, hỗ trợ người dân hiểu rõ quy trình và yêu cầu trước khi thực hiện thủ tục tại cơ quan có thẩm quyền.

Nhìn chung, phạm vi ứng dụng của dự án được xác định rõ ràng, tập trung vào tính khả thi và hiệu quả thực tiễn, đồng thời tạo nền tảng cho việc mở rộng hệ thống trong tương lai.

2.1.3 Đối tượng sử dụng

- Người quản lý.
- Người dùng (khách hàng) chưa có tài khoản.
- Người dùng (khách hàng) đã có tài khoản.

2.2 Phân tích nghiệp vụ

2.2.1 Chức năng cơ bản

- Nhắn tin với chatbot AI:

  - Người dùng truy cập hệ thống, dù đã đăng ký hay chưa đăng ký tài khoản, đều có thể thực hiện thao tác đặt câu hỏi và nhận câu trả lời từ chatbot AI về các thủ tục hành chính.Quản lý chatbot AI:
  - Tùy theo đối tượng người dùng, hệ thống sẽ cho phép sử dụng mô hình GPT miễn phí hoặc mô hình GovAI trả phí.

- Đăng ký tài khoản:
  - Người dùng truy cập ứng dụng có thể có hoặc không có tài khoản. Nếu có tài khoản sẽ được tiếp cận nhiều chức năng hơn. Thao tác đăng ký rất nhanh với một số ít thông tin cơ bản như: email, mật khẩu.
- Đăng nhập – đăng xuất:
  - Sử dụng tài khoản đã đăng ký để đăng nhập và trải nghiệm ứng dụng.
- Thay đổi mật khẩu:
  - Người dùng nhập đúng mật khẩu cũ, sau đó nhập mật khẩu mới để cập nhật mật khẩu cho tài khoản của mình.
- Thay đổi thông tin cá nhân:
  - Người dùng có thể thay các thông tin cá nhân của tài khoản
- Tra cứu thủ tục hành chính có sẵn:
  - Người dùng có thể tra cứu danh sách các thủ tục hành chính đã được hệ thống tổng hợp và lưu trữ.
  - Các thủ tục được phân loại theo lĩnh vực, cơ quan giải quyết và từ khóa tìm kiếm.
- Đọc bài blog về thủ tục hành chính:
  - Người dùng có thể đọc các bài blog hướng dẫn, phân tích và giải thích thủ tục hành chính theo ngôn ngữ dễ hiểu.
  - Các bài viết giúp người dân nắm bắt thông tin trước khi thực hiện thủ tục hoặc sử dụng chatbot AI.

2.2.2 Nghiệp vụ chính của người quản lý

- Nhắn tin với chatbot AI

- Quản lý chatbot AI: Cho phép xem danh sách, cập nhật, xóa các model AI trong hệ thống
- Xem danh sách các cuộc trò chuyện hiện có
- Xem thống kê theo
- Quản lý thông tin cá nhân
- Quản lý người dùng (khách hàng)
- Quản lý nạp tiền
- Quản lý blog

2.2.3 Nghiệp vụ chính của người dùng đã có tài khoản

- Nhắn tin với chatbot AI
- Xem các model mà website hiện đang hỗ trợ
- Xem thống kê lịch sử trò chuyện
- Xem thống kê lịch sử sử dụng số dư
- Quản lý thông tin cá nhân
- Nạp tiền và sử dụng dịch vụ trả phí
- Tra cứu thủ tục hành chính có sẵn
- Đọc blog

2.2.4 Nghiệp vụ chính của người dùng chưa có tài khoản

- Có thể nhắn tin với chatbot
- Đăng ký tài khoản
- Đọc blog
- Tra cứu thủ tục hành chính có sẵn
- Xem thống kê lịch sử trò chuyện

2.2.5 Yêu cầu

Ứng dụng cần đáp ứng các yêu cầu sau:

- **Trải nghiệm người dùng tốt:**

  Giao diện thân thiện, thao tác mượt mà và dễ sử dụng trên các thiết bị.

- **Tính thẩm mỹ và định hướng thương mại:**

  Giao diện cần được thiết kế đẹp, hiện đại, có chủ đề rõ ràng, phù hợp cho việc phát triển thương mại trong tương lai.

- **Độ chính xác cao của thông tin:**

  Kết quả trả về từ chatbot AI, đặc biệt là mô hình GovAI, phải bám sát dữ liệu thủ tục hành chính đã được cung cấp và đảm bảo độ tin cậy cao.

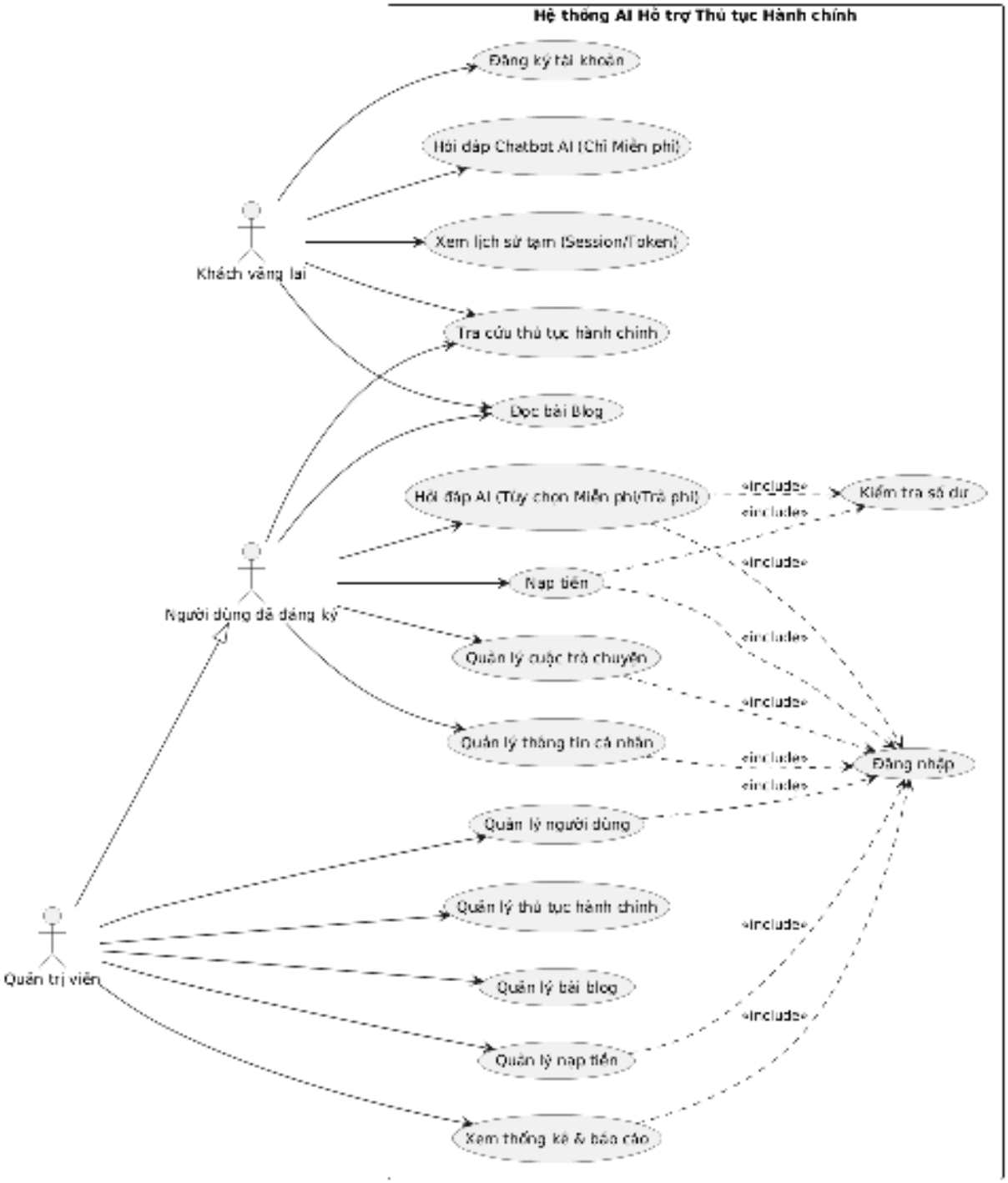
- **Bảo mật và an toàn dữ liệu:**

  Thông tin người dùng, lịch sử sử dụng và dữ liệu thanh toán phải được bảo vệ an toàn.

2.3 Thiết kế hệ thống

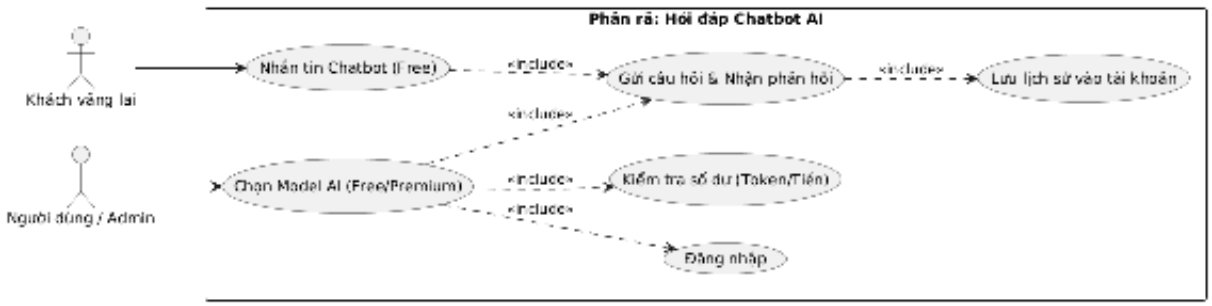
2.3.1 Sơ đồ ca sử dụng

- Sơ đồ ca sử dụng tổng quan của hệ thống



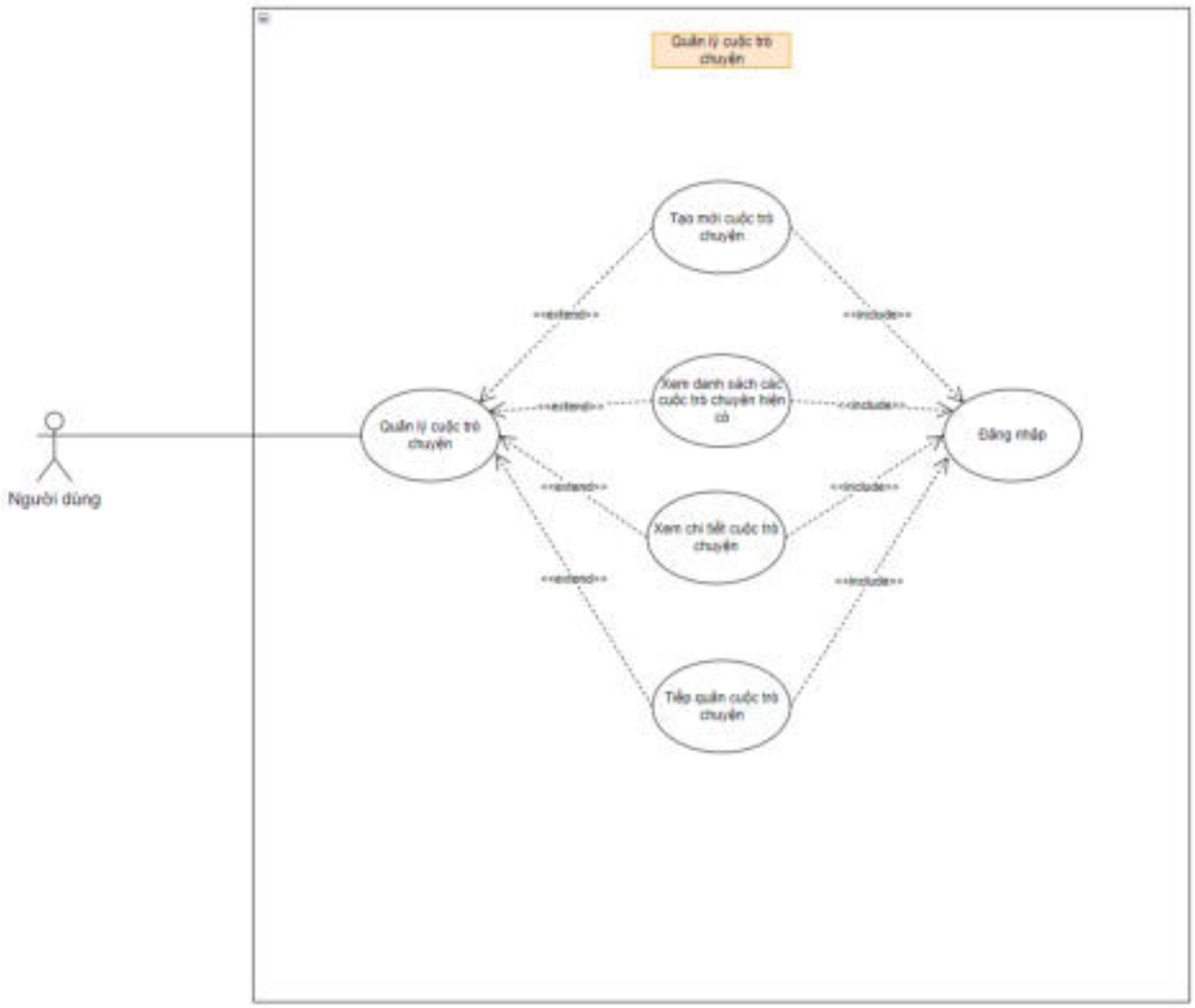
Hình 2.1 Biểu đồ usecase tổng quan hệ thống

- Sơ đồ phân rã ca sử dụng chức năng hỏi đáp



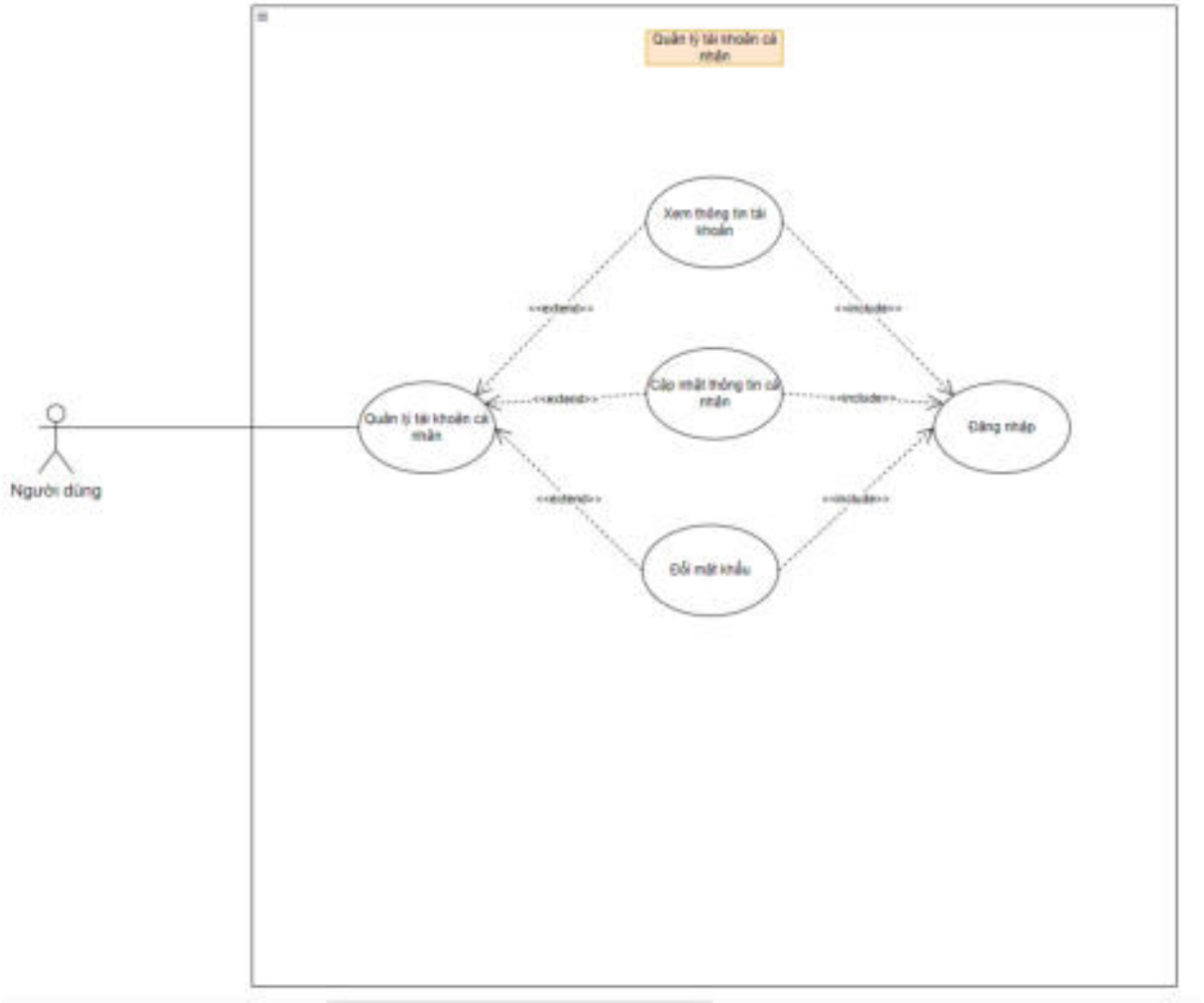
Hình 2.2 Sơ đồ phân rã ca sử dụng chức năng hỏi đáp và quản lý cuộc trò chuyện

- Sơ đồ phân rã ca sử dụng chức năng quản lý cuộc trò chuyện



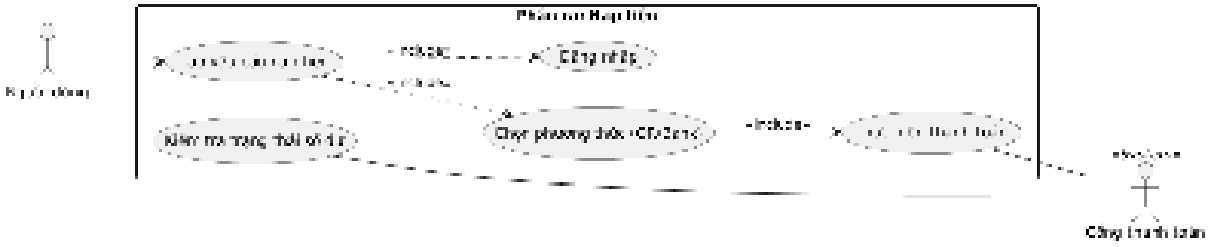
Hình 2.3 Sơ đồ phân rã ca sử dụng chức năng quản lý cuộc trò chuyện

- Sơ đồ phân rã ca sử dụng chức năng quản lý tài khoản cá nhân



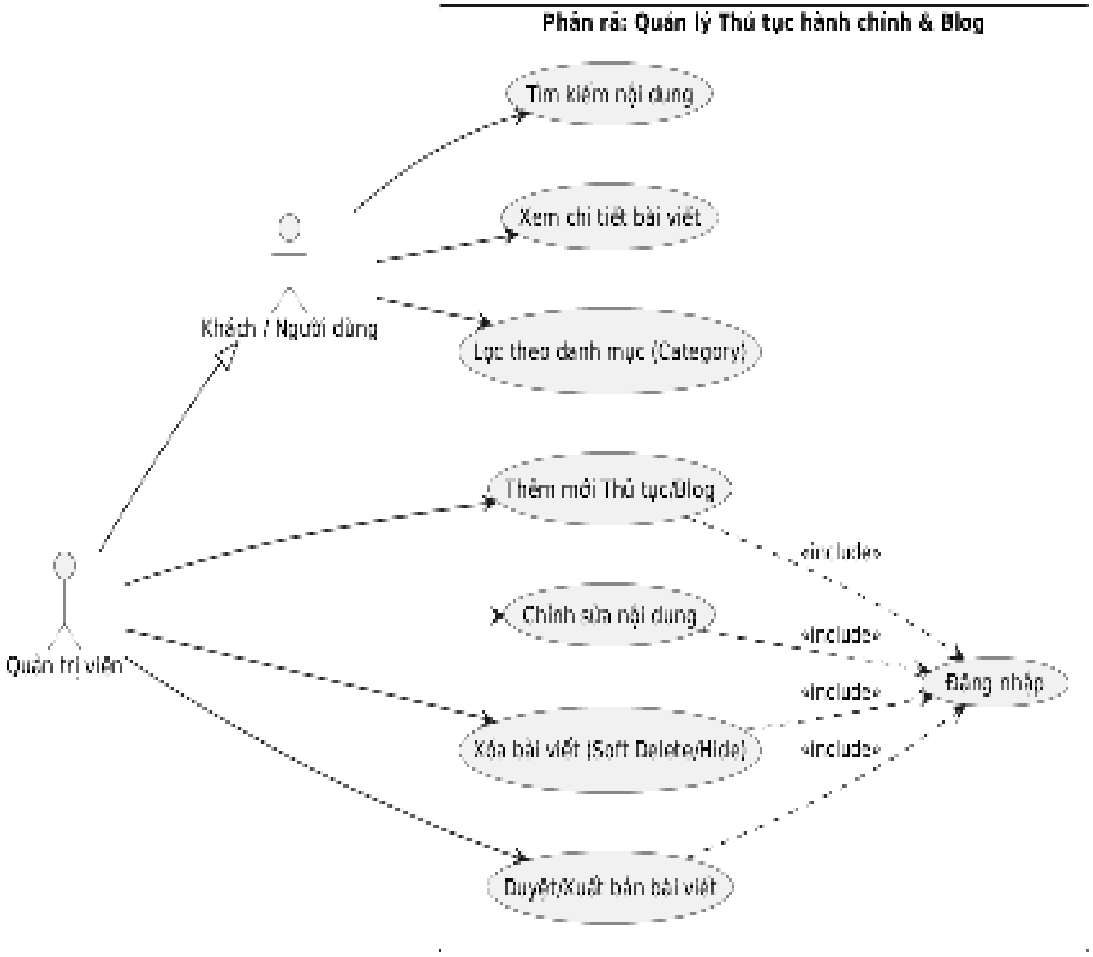
Hình 2.4 Sơ đồ phân rã ca sử dụng chức năng quản lý tài khoản cá nhân

- Sơ đồ phân rã ca sử dụng chức năng nạp tiền



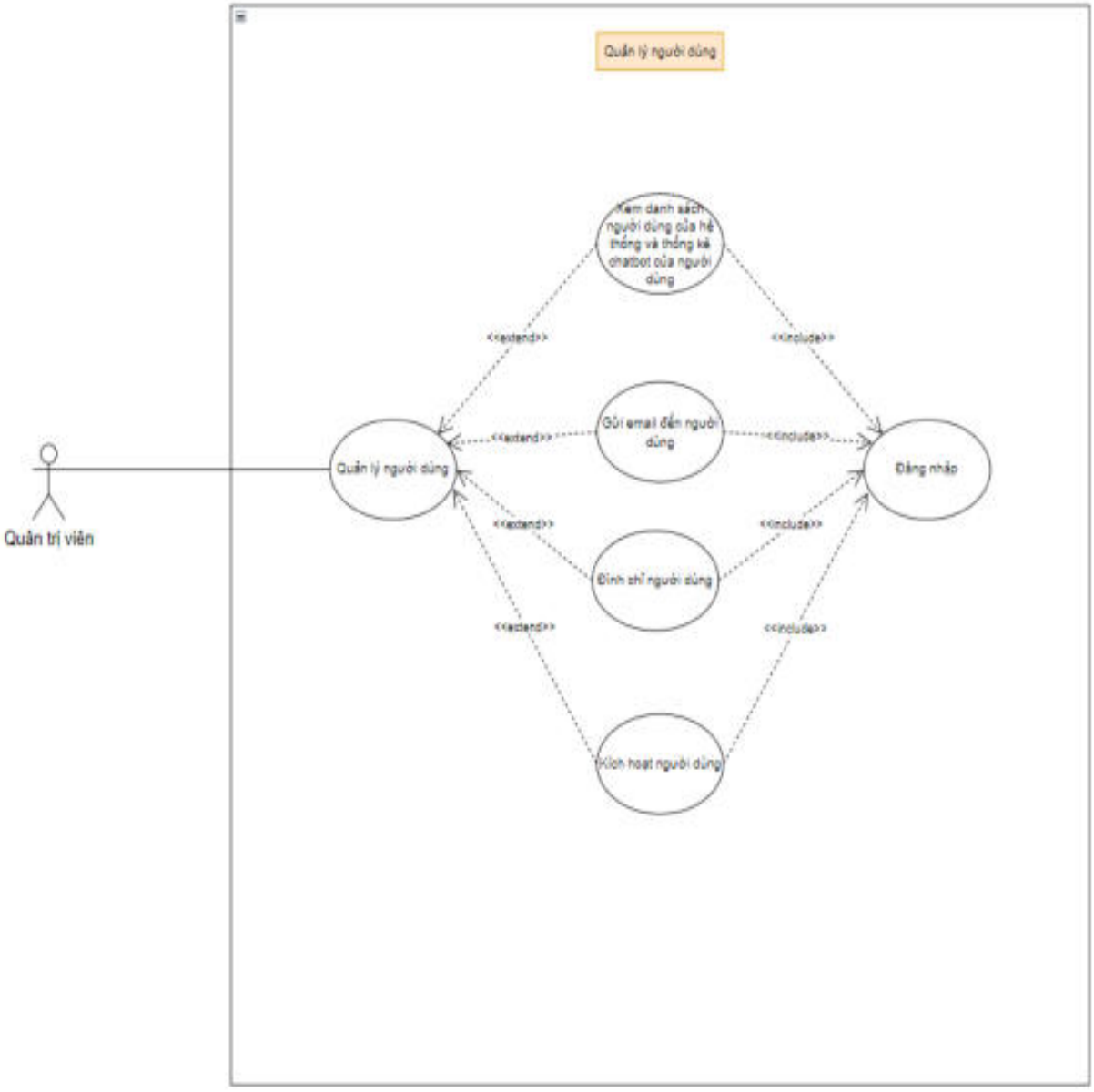
Hình 2.5 Sơ đồ phân rã ca sử dụng chức năng nạp tiền

- Sơ đồ phân rã ca sử dụng chức năng quản lý thủ tục hành chính & blog



Hình 2.6 Sơ đồ phân rã ca sử dụng chức năng quản lý thủ tục hành chính & blog

- Sơ đồ phân rã ca sử dụng chức năng quản lý người dùng



Hình 2.7 Sơ đồ phân rã ca sử dụng chức năng quản lý người dùng

2.3.2 Phân tích đặc tả yêu cầu chức năng

- Đăng ký

Mã chức năng	01
Tên chức năng	Đăng ký
Đối tượng sử dụng	Người dùng
Tiền điều kiện	Email không tồn tại trong hệ thống
Quy trình nghiệp vụ	Trang Đăng ký -> Điền tên, email, mật khẩu, nhập lại mật khẩu -> Nhấn nút "Sign up"
Kết quả	• Nếu thành công ⇨ Chuyển qua trang đăng nhập. • Nếu thất bại ⇨ Sẽ hiện thông báo lỗi.
Ghi chú	

Bảng 2.1 Đặc tả chức năng đăng ký

- Đăng nhập

Mã chức năng	02
Tên chức năng	Đăng nhập
Đối tượng sử dụng	Người dùng
Tiền điều kiện	Tài khoản tồn tại trong hệ thống
Quy trình nghiệp vụ	Trang Đăng nhập-> Điền email, mật khẩu -> Nhấn nút "Sign in"
Kết quả	• Nếu thành công ⇨ Vào trang chủ của website. • Nếu thất bại ⇨ Sẽ hiện thông báo lỗi.
Ghi chú	

Bảng 2.2 Đặc tả chức năng đăng nhập

- Quên mật khẩu

Mã chức năng	03
Tên chức năng	Quên mật khẩu
Đối tượng sử dụng	Người dùng
Tiền điều kiện	Tài khoản tồn tại trong hệ thống
Quy trình nghiệp vụ	Trang Quên mật khẩu -> Điền email -> Nhấn nút "Submit"
Kết quả	• Nếu thành công ⇨ Người dùng nhận được đường link để khôi phục mật khẩu • Nếu thất bại ⇨ Sẽ hiện thông báo lỗi.
Ghi chú	

Bảng 2.3 Đặc tả chức năng quên mật khẩu

- Đăng xuất

Mã chức năng	04
Tên chức năng	Đăng xuất
Đối tượng sử dụng	Người dùng
Tiền điều kiện	Người dùng đã đăng nhập vào hệ thống
Quy trình nghiệp vụ	Nhấn vào avatar người dùng ở khu vực navbar -> Hiển thị trang thông tin tài khoản -> Nhấn nút ""
Kết quả	• Thành công ⇨ Chuyển qua trang đăng nhập. • Nếu thất bại ⇨ Sẽ hiện thông báo lỗi.
Ghi chú	

Bảng 2.4 Đặc tả chức năng đăng xuất

- Tạo mới cuộc trò chuyện

Mã chức năng	05
Tên chức năng	Tạo cuộc trò chuyện
Đối tượng sử dụng	Người dùng
Tiền điều kiện	
Quy trình nghiệp vụ	Nhấn vào nút "Chat với AI" ở trang chủ hoặc nút "Tạo cuộc trò chuyện mới" ở khu vực chat
Kết quả	• Nếu thành công ⇨ Chuyển hướng tới trang chat. • Nếu thất bại ⇨ Sẽ hiện thông báo lỗi.
Ghi chú	

Bảng 2.5 Đặc tả chức năng tạo mới cuộc trò chuyện

- Xem thống kê

Mã chức năng	06
Tên chức năng	Xem thống kê
Đối tượng sử dụng	Người dùng
Tiền điều kiện	Người dùng phải đăng nhập vào hệ thống
Quy trình nghiệp vụ	Nhấn vào khu vực chứa icon "Dashboard" ở khu vực navbar => Chọn thời gian bạn muốn xem thống kê
Kết quả	• Nếu thành công ⇨ Hiển thị 4 biểu đồ đường với các giá trị trong đó. ⇨ Hiển thị 1 bảng danh sách thống kê các thông tin chatbot mà người dùng tạo • Nếu thất bại ⇨ Sẽ hiện thông báo lỗi.
Ghi chú	

Bảng 2.6 Đặc tả chức năng xem thống kê

- Xem danh sách cuộc trò chuyện

Mã chức năng	07
Tên chức năng	Xem danh sách cuộc trò chuyện
Đối tượng sử dụng	Người dùng
Tiền điều kiện	Người dùng phải đăng nhập vào hệ thống
Quy trình nghiệp vụ	Nhấn vào khu vực chứa icon "Trò chuyện" ở khu vực navbar
Kết quả	• Nếu thành công ⇨ Hiển thị bảng danh sách các cuộc hội thoại của chatbot mà người dùng tạo. • Nếu thất bại ⇨ Sẽ hiện thông báo lỗi.
Ghi chú	

Bảng 2.7 Đặc tả chức năng xem danh sách cuộc trò chuyện

- Xem danh sách người dùng

Mã chức năng	08
Tên chức năng	Xem danh sách người dùng
Đối tượng sử dụng	Người quản lý (Người tạo nên ứng dụng)
Tiền điều kiện	Người quản lý phải đăng nhập vào hệ thống
Quy trình nghiệp vụ	Nhấn vào khu vực chứa icon "Admin" ở khu vực navbar
Kết quả	• Nếu thành công ⇨ Hiển thị bảng danh sách người dùng của ứng dụng. • Nếu thất bại ⇨ Sẽ hiện thông báo lỗi.
Ghi chú	

Bảng 2.8 Đặc tả chức năng xem danh sách người dùng của ứng dụng

- Xem thông tin tài khoản

Mã chức năng	09
Tên chức năng	Xem thông tin tài khoản
Đối tượng sử dụng	Người dùng
Tiền điều kiện	- Đã đăng nhập vào hệ thống
Quy trình nghiệp vụ	Nhấn vào avatar người dùng ở khu vực navbar
Kết quả	• Thành công ⇨ Hiển thị trang thông tin tài khoản. • Nếu thất bại ⇨ Sẽ hiện thông báo lỗi.
Ghi chú	

Bảng 2.9 Đặc tả chức năng xem thông tin tài khoản

- Cập nhật thông tin

Mã chức năng	10
Tên chức năng	Cập nhật thông tin
Đối tượng sử dụng	Người dùng
Tiền điều kiện	- Đã đăng nhập vào hệ thống
Quy trình nghiệp vụ	Nhấn vào avatar người dùng ở khu vực navbar -> Hiển thị trang thông tin tài khoản -> Nhấn vào "Thông tin cá nhân" -> Nhập thông tin mới và nhấn lưu
Kết quả	• Thành công ⇨ Hiển thị thông báo cập nhật thông tin thành công. • Nếu thất bại ⇨ Sẽ hiện thông báo lỗi.
Ghi chú	

Bảng 2.10 Đặc tả chức năng cập nhật thông tin tài khoản

- Đổi mật khẩu

Mã chức năng	11
Tên chức năng	Đổi mật khẩu
Đối tượng sử dụng	Người dùng
Tiền điều kiện	Đã đăng nhập vào hệ thống
Quy trình nghiệp vụ	Nhấn vào avatar người dùng ở khu vực navbar -> Hiển thị trang thông tin tài khoản -> Nhấn vào nút "Đổi mật khẩu" -> Hiển thị trang đổi mật khẩu -> Nhập mật khẩu củ, mật khẩu mới và nhấn nút "Lưu mật khẩu"
Kết quả	• Thành công ⇨ Hiển thị thông báo đổi mật khẩu thành công. • Nếu thất bại ⇨ Sẽ hiện thông báo lỗi.
Ghi chú	

Bảng 2.11 Đặc tả chức năng đổi mật khẩu

- Nạp tiền

Mã chức năng	12
Tên chức năng	Nạp tiền
Đối tượng sử dụng	Người dùng & Người quản lý
Tiền điều kiện	Đã đăng nhập vào hệ thống
Quy trình nghiệp vụ	Nhấn vào avatar người dùng ở khu vực navbar -> Nhấn vào nút "Nạp tiền"-> Hiển thị trang nhập số tiền-> Hiển thị QR-> Chuyển tiền và đợi
Kết quả	• Thành công ⇨ Hiển thị thông báo nạp tiền thành công • Nếu thất bại, sẽ hiện thông báo lỗi.
Ghi chú	

Bảng 2.12 Đặc tả chức năng nạp tiền

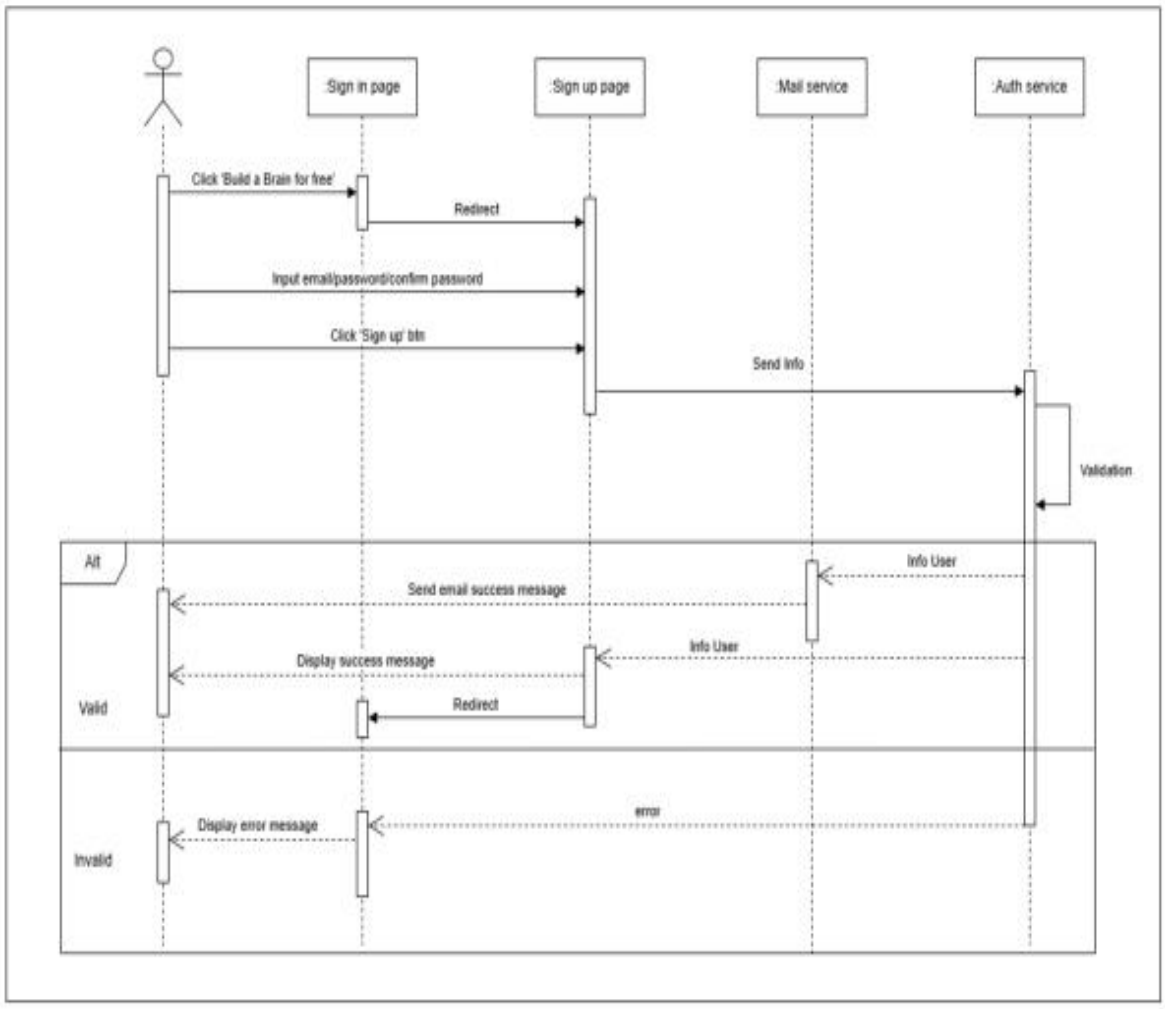
- Quản lý nạp tiền

Mã chức năng	13
Tên chức năng	Quản lý nạp tiền
Đối tượng sử dụng	Người quản lý (Người tạo nên ứng dụng)
Tiền điều kiện	Đã đăng nhập vào hệ thống
Quy trình nghiệp vụ	Nhấn vào nút Admin ở thanh navbar -> Nhấn vào nút "Nạp tiền"-> Người dùng có thể xem danh sách/ trạng thái các giao dịch nộp tiền-> Người dùng có thể từ chối hoặc xác nhận giao dịch thủ công
Kết quả	• Thành công ⇨ Hiển thị thông báo cập nhận trạng thái giao dịch thành công • Nếu thất bại ⇨ Sẽ hiện thông báo lỗi.
Ghi chú	

Bảng 2.13 Đặc tả chức năng quản lý nạp tiền

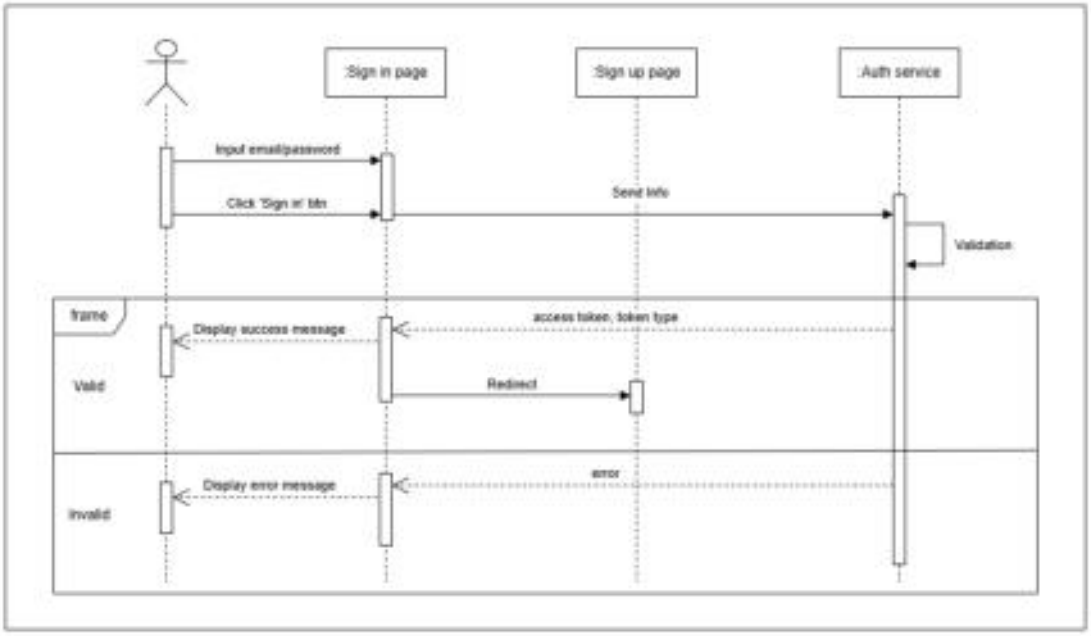
2.3.3 Sơ đồ tuần tự

- Đăng ký



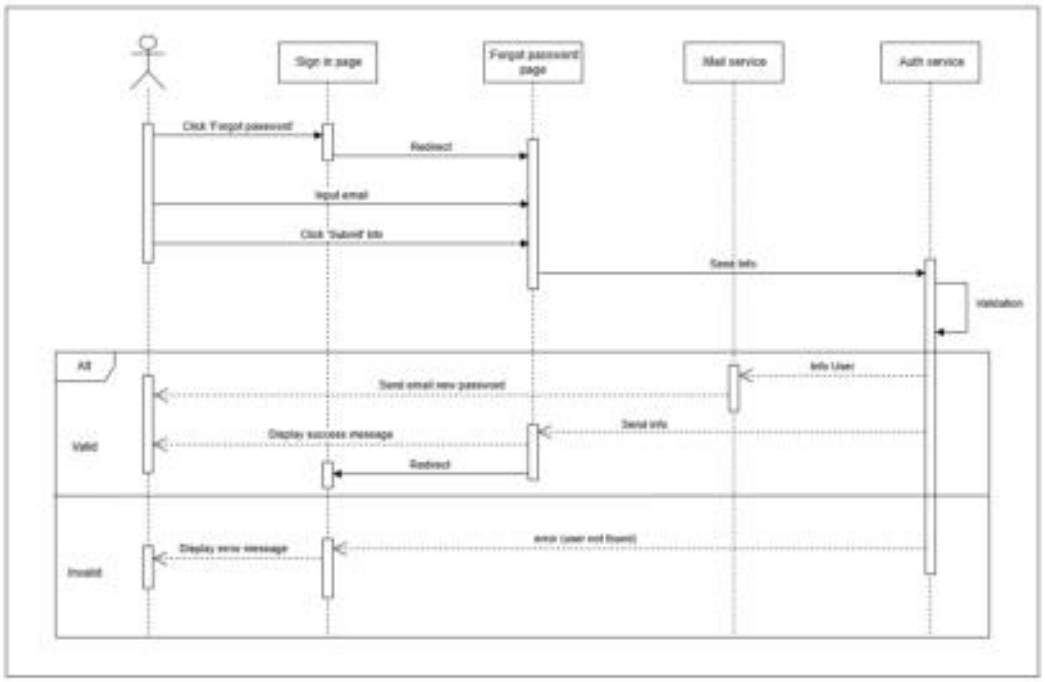
Hình 2.8 Sơ đồ tuần tự chức năng đăng ký

- Đăng nhập

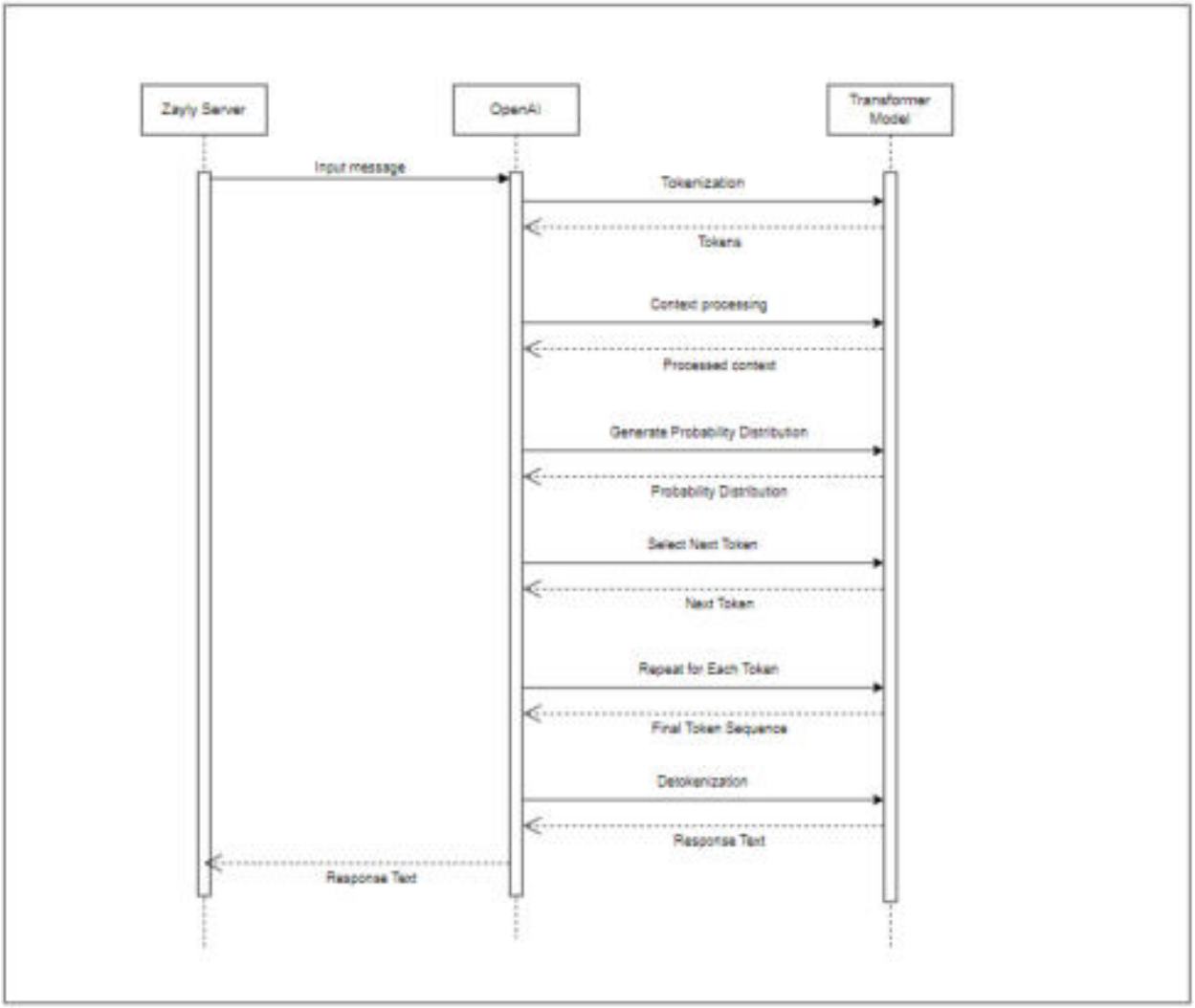


Hình 2.9 Sơ đồ tuần tự chức năng đăng nhập

- Quên mật khẩu

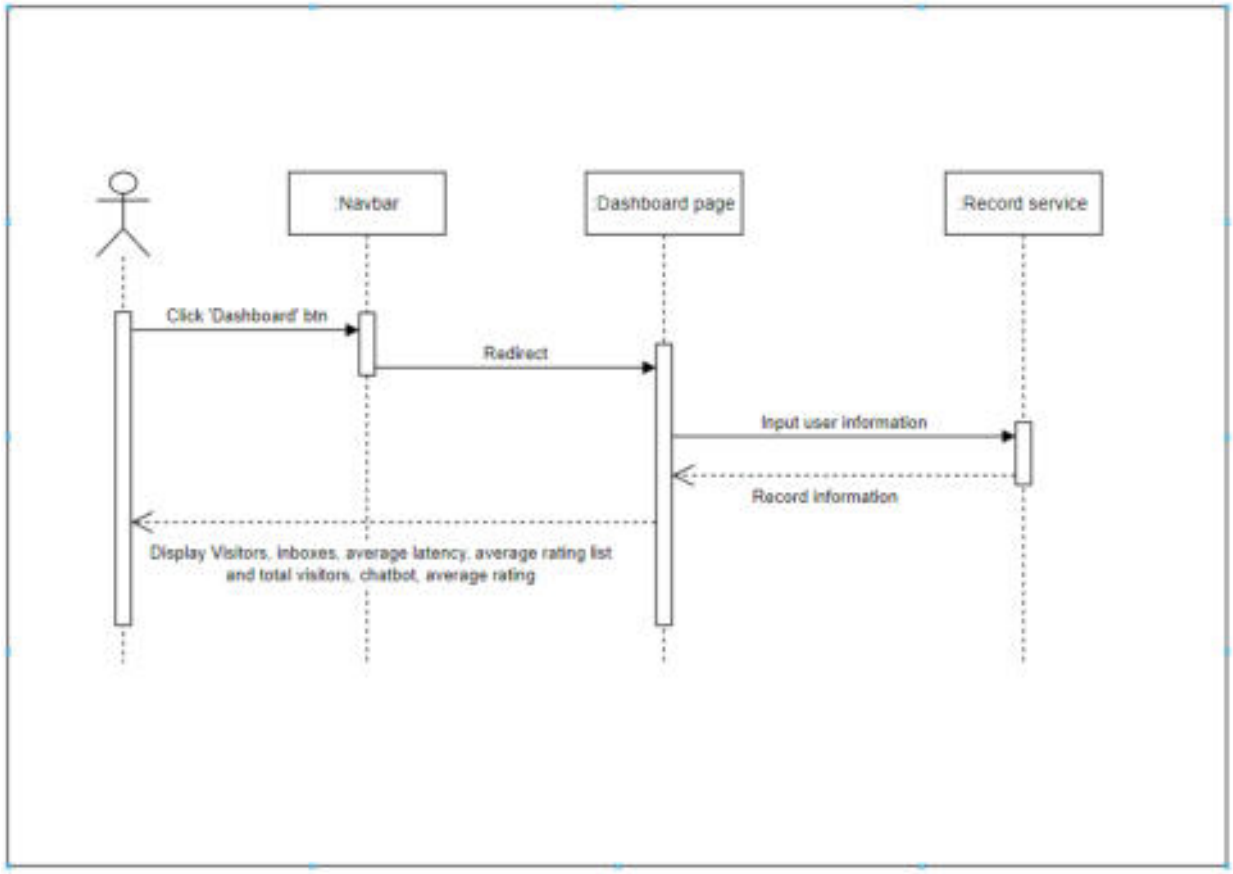


Hình 2.10 Sơ đồ tuần tự chức năng quên mật khẩu



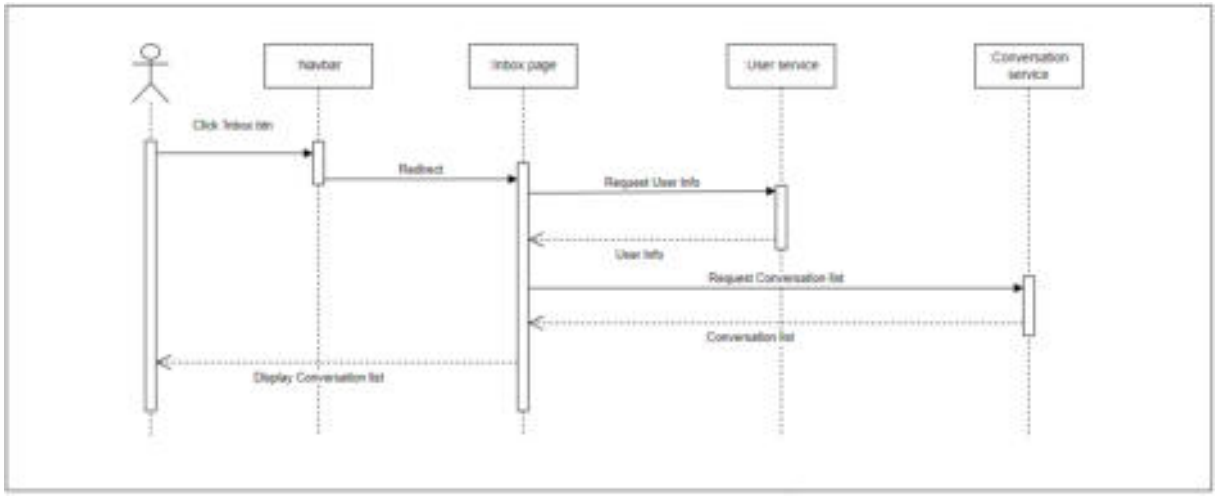
Hình 2.11 Sơ đồ tuần tự giao tiếp giữa server và OpenAI

- Xem thống kê



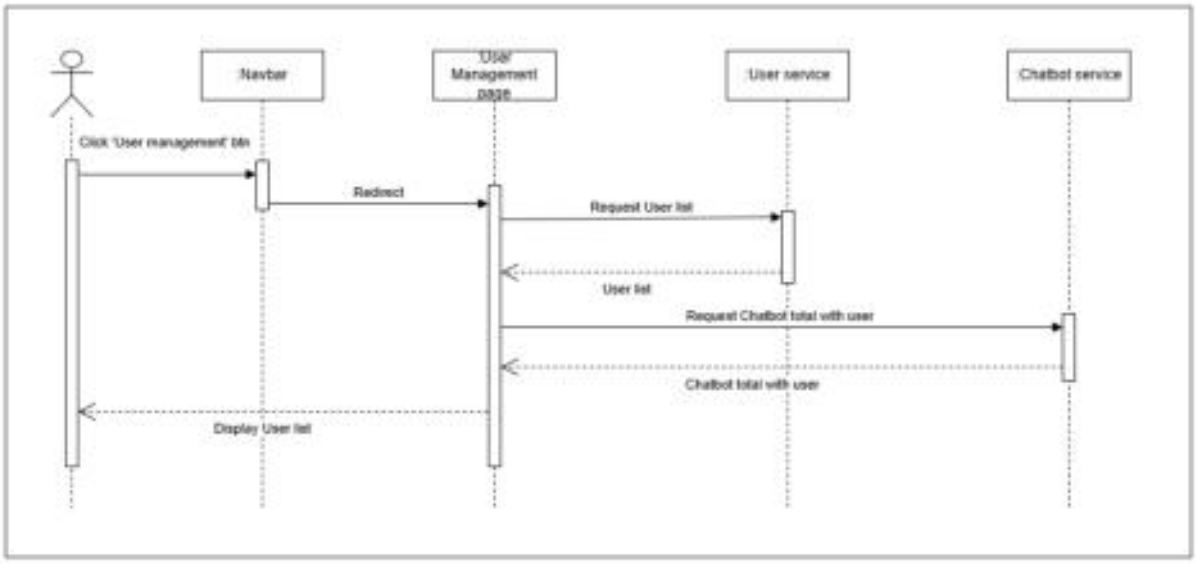
Hình 2.12 Sơ đồ tuần tự chức năng xem thống kê

- Xem danh sách cuộc trò chuyện



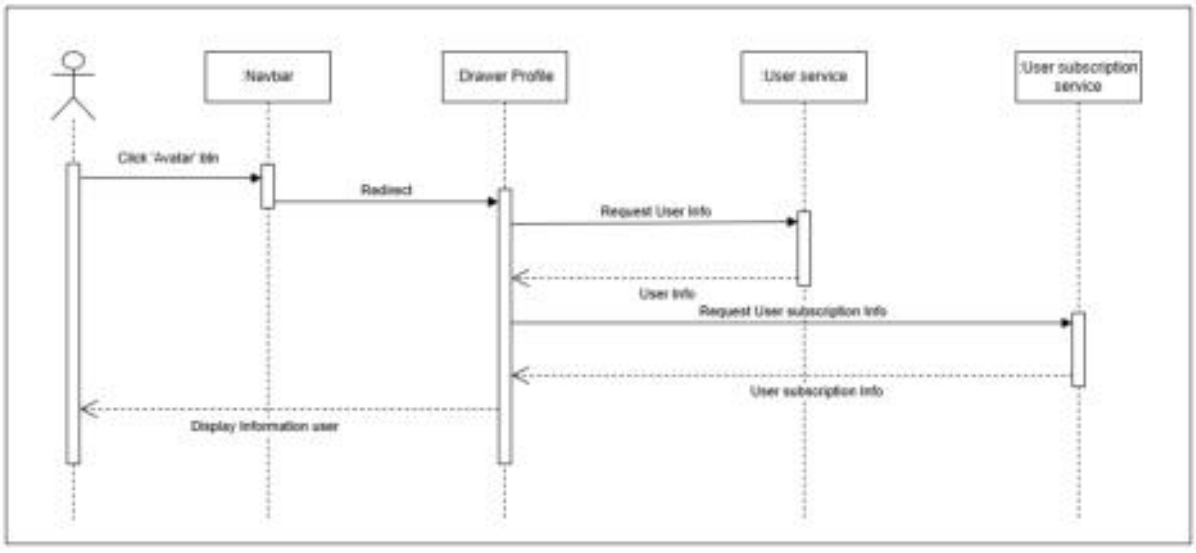
Hình 2.13 Sơ đồ tuần tự chức năng xem danh sách cuộc trò chuyện

- Xem danh sách người dùng



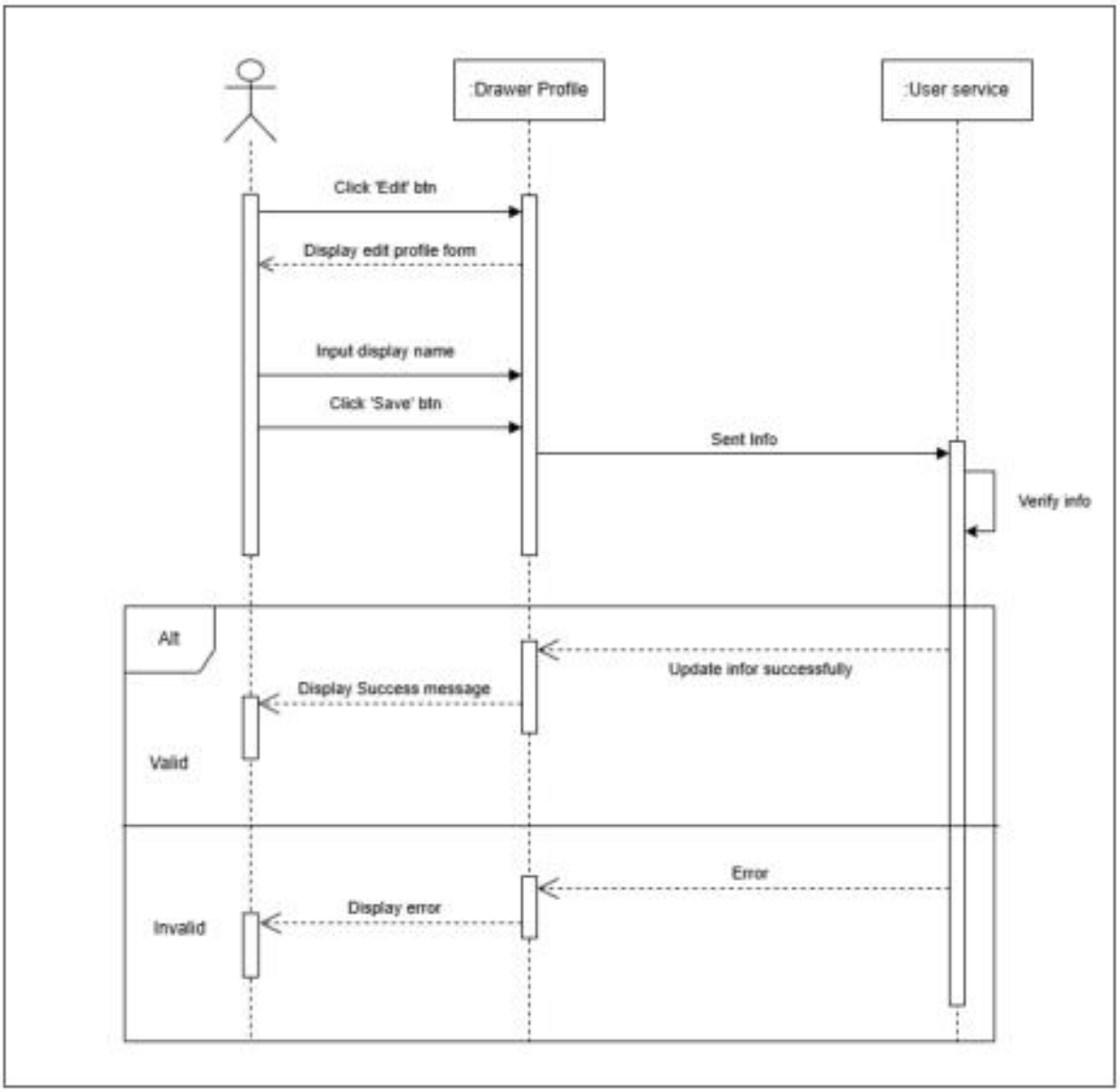
Hình 2.14 Sơ đồ tuần tự chức năng xem danh sách người dùng

- Xem thông tin tài khoản



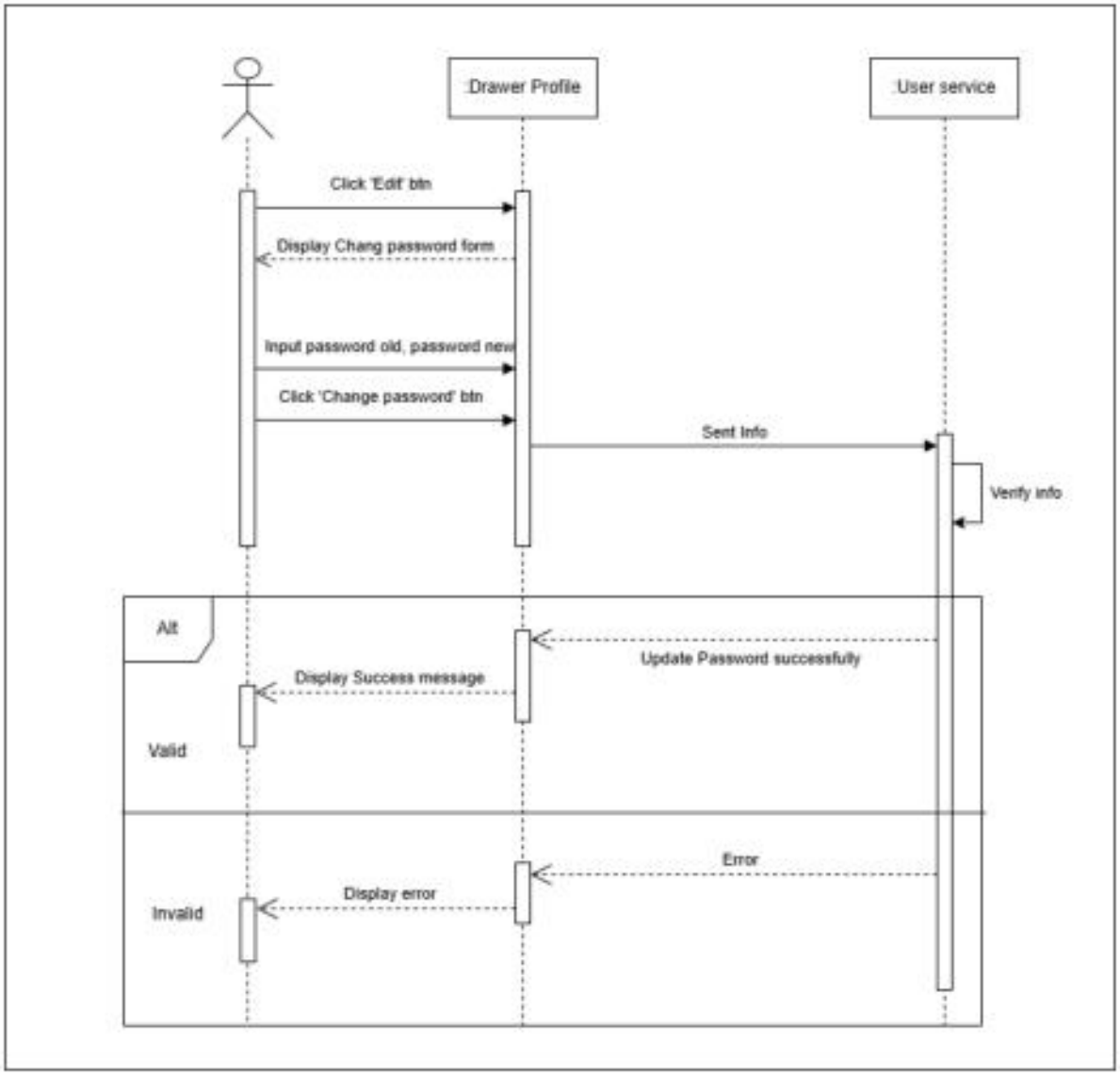
Hình 2.15 Sơ đồ tuần tự chức năng xem thông tin tài khoản

- Cập nhật thông tin tài khoản



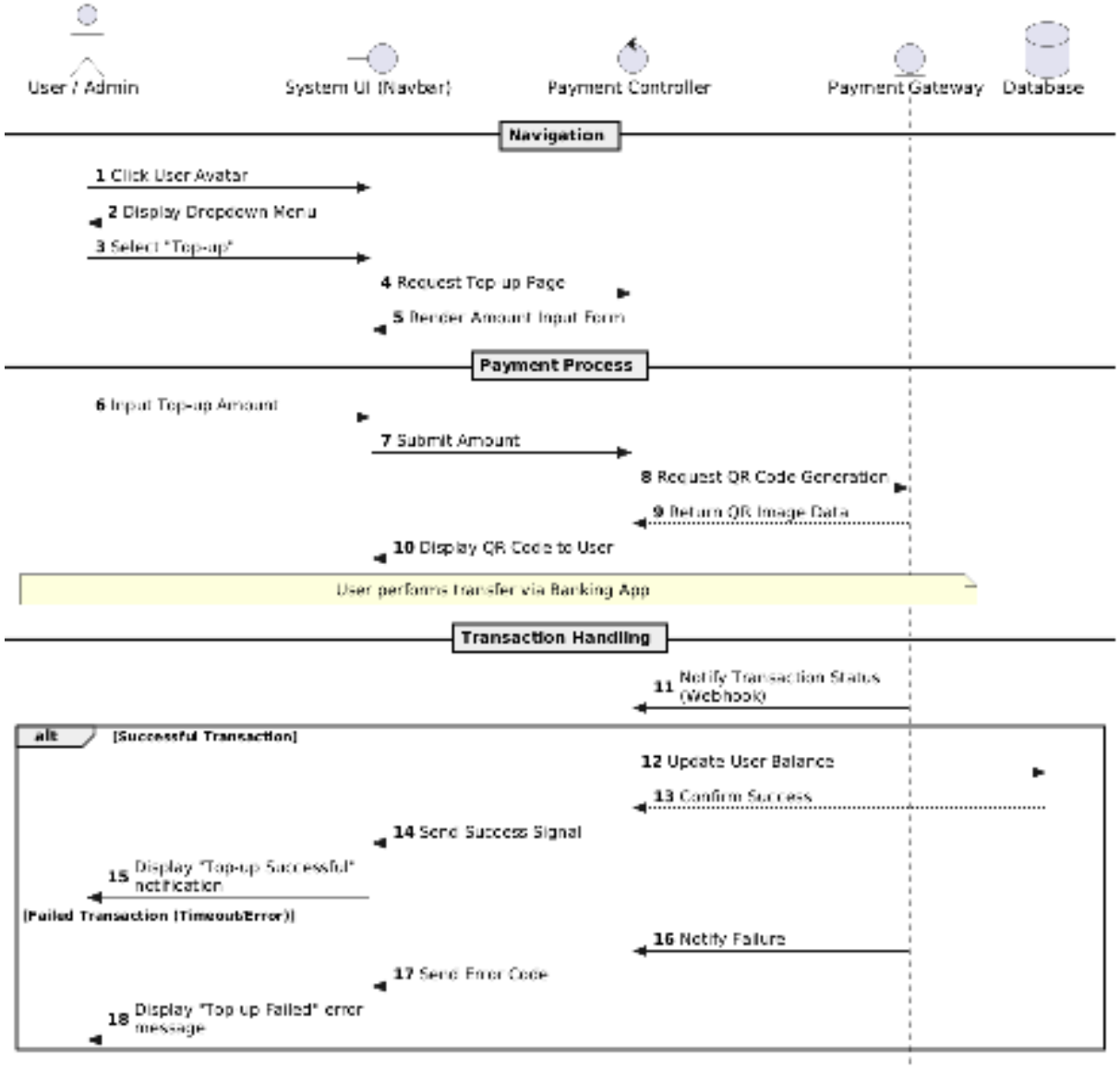
Hình 2.16 Sơ đồ tuần tự chức năng cập nhật thông tin tài khoản

- Đổi mật khẩu



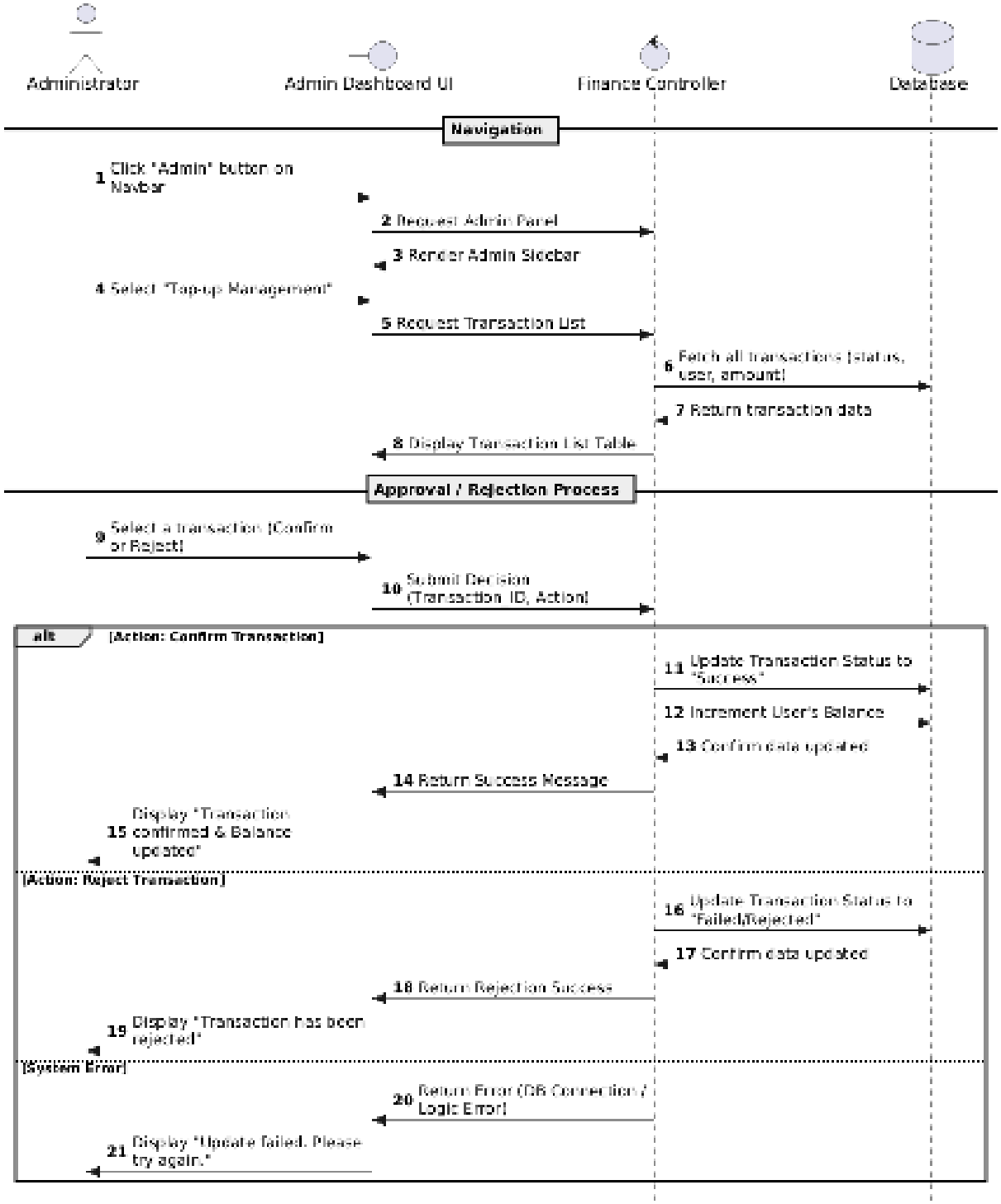
Hình 2.17 Sơ đồ tuần tự chức năng đổi mật khẩu

- Nạp tiền



Hình 2.18 Sơ đồ tuần tự chức năng nạp tiền

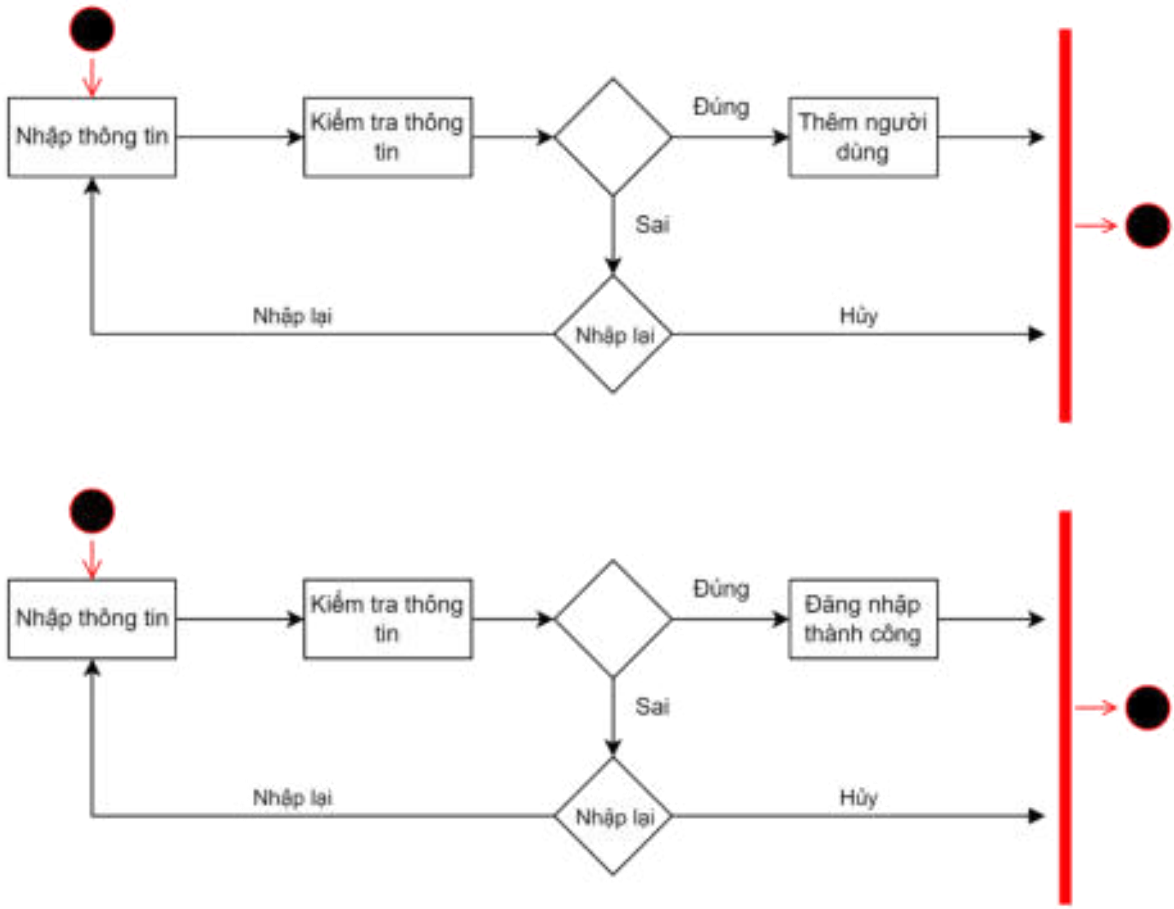
- Quản lý nạp tiền



Hình 2.19 Sơ đồ tuần tự chức năng quản lý nạp tiền

2.3.4 Sơ đồ nguyên lý hoạt động

- **Sơ đồ nguyên lý hoạt động đăng ký, đăng nhập**



Hình 2.20 Sơ đồ hoạt động: Đăng nhập và đăng ký

Để đăng ký tài khoản, người dùng nhập các thông tin yêu cầu và gửi yêu cầu đến hệ thống. Hệ thống sẽ tiến hành kiểm tra thông tin và đăng ký tài khoản mới. Bao gồm email, tên người dùng và mật khẩu. Mật khẩu phải hơn 5 chữ số. Email không được trùng với email đã đăng ký. Nếu đăng ký thành công, đồng thời thêm dữ liệu người dùng mới. Ngược lại, nếu thông tin không hợp lệ, hệ thống sẽ yêu cầu người dùng nhập lại các thông tin.

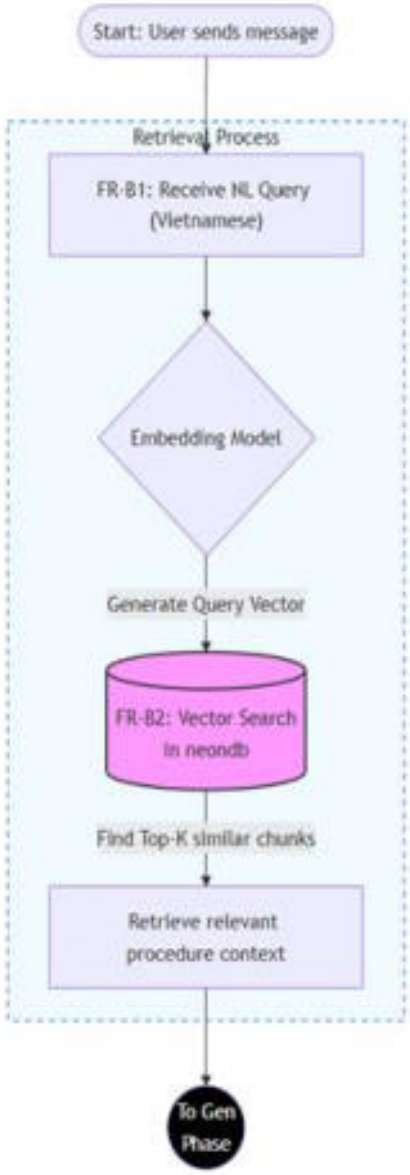
Sau khi đăng ký thành công sẽ chuyển đến trang đăng nhập. Đăng nhập thành công sẽ chuyển đến trang người dùng. Tại đây sẽ tùy vào phân quyền cao nhất của người dùng có mà sẽ hiển thị các chức năng cho phù hợp.

- **Sơ đồ nguyên lý hoạt động: Tạo cuộc hội thoại mới**

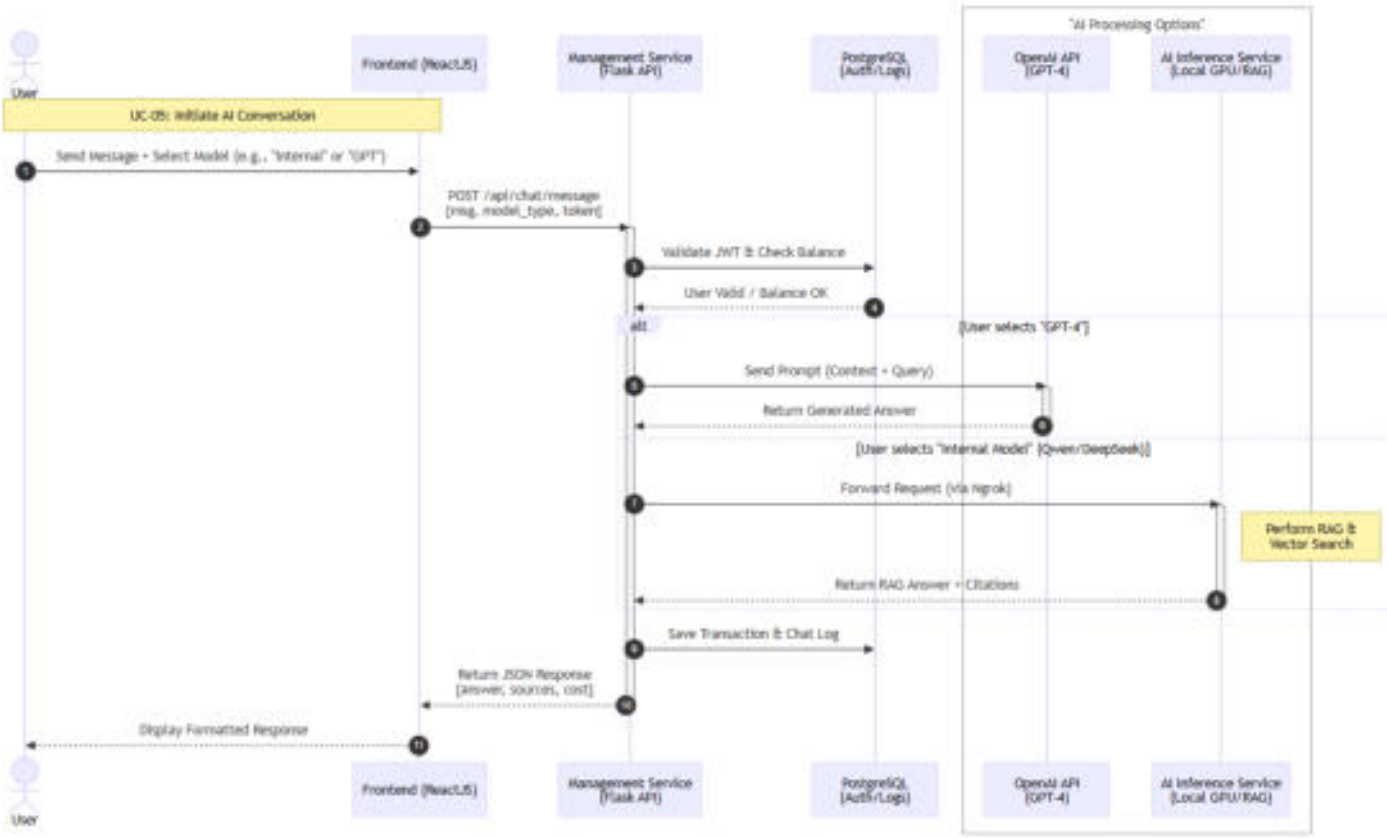
Chức năng **Tạo cuộc hội thoại mới** cho phép người dùng khởi tạo một phiên trò chuyện với hệ thống chatbot AI. Người dùng có thể **đã đăng nhập** hoặc **chưa đăng nhập (khách vãng lai)**, hệ thống sẽ xử lý theo hai luồng khác nhau nhưng vẫn đảm bảo trải nghiệm thống nhất.

Quy trình hoạt động được mô tả như sau:

1. **Người dùng truy cập hệ thống**
   Người dùng có thể truy cập hệ thống mà không cần đăng nhập hoặc đăng nhập bằng tài khoản đã đăng ký.
2. **Kiểm tra trạng thái đăng nhập**
   - Nếu **chưa đăng nhập**:
     Hệ thống tự động tạo một **tài khoản tạm thời dựa trên token phiên** để quản lý cuộc hội thoại.
   - Nếu **đã đăng nhập**:
     Hệ thống xác thực thông tin người dùng thông qua JWT và sử dụng tài khoản hiện có.
3. **Khởi tạo cuộc hội thoại mới**
   Hệ thống tạo một bản ghi cuộc hội thoại mới trong cơ sở dữ liệu, gắn với:
   - Token phiên (đối với khách vãng lai), hoặc
   - ID người dùng (đối với người dùng đã đăng nhập).
4. **Sẵn sàng nhận tin nhắn đầu tiên**
   Sau khi cuộc hội thoại được tạo thành công, hệ thống chuyển sang trạng thái sẵn sàng tiếp nhận câu hỏi đầu tiên từ người dùng.



Hình 2.21 Sơ đồ hoạt động – RAG & Chat



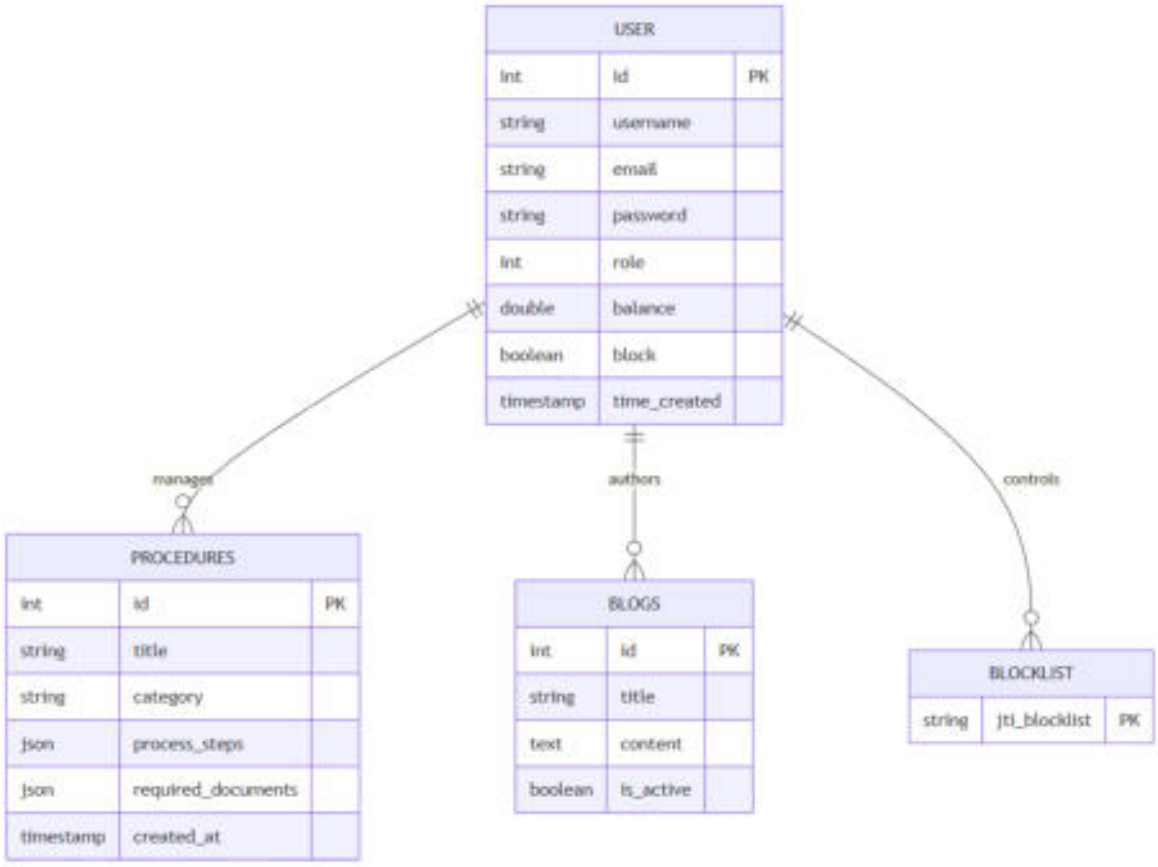
Hình 2.22 Sơ đồ hoạt động: Tạo một cuộc hội thoại mới

2.4 Thiết kế cơ sở dữ liệu

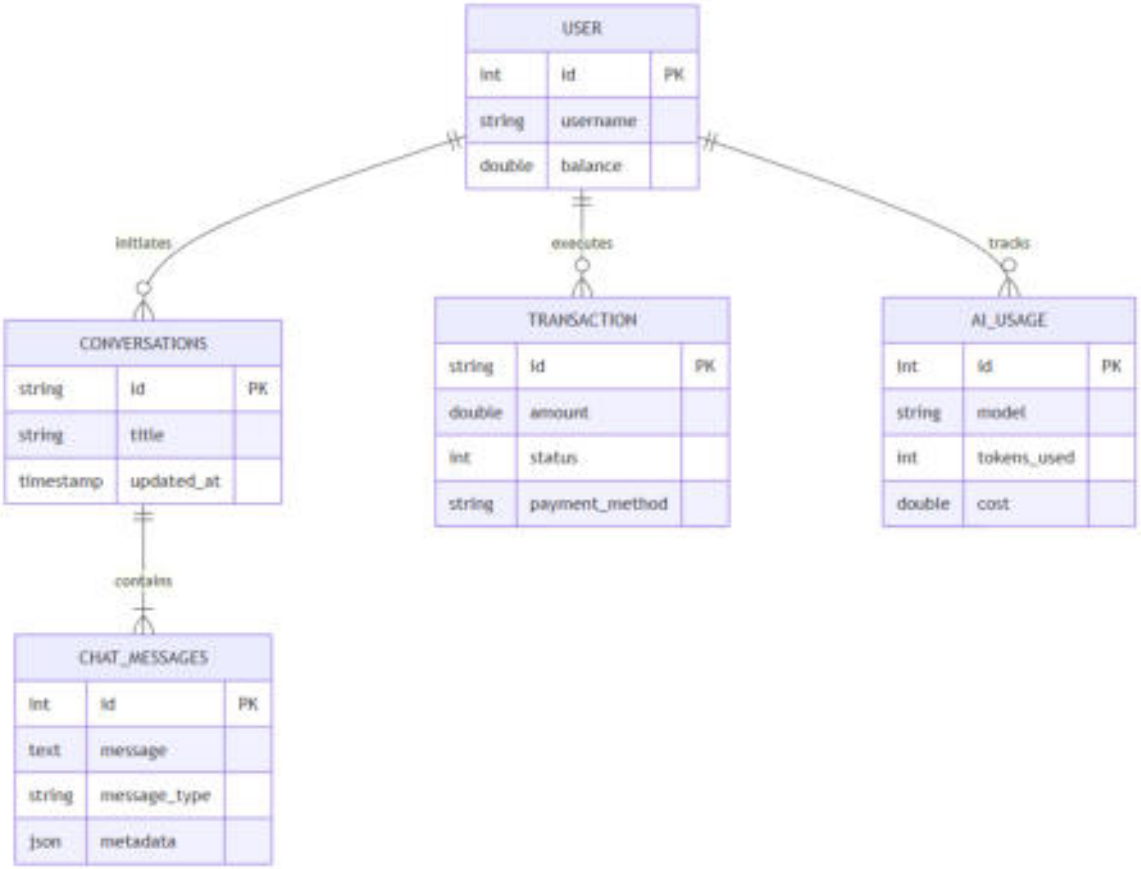
Sơ đồ cơ sở dữ liệu được chuẩn hóa về dạng **3NF (Dạng chuẩn 3)** để đảm bảo tính nhất quán của dữ liệu.

2.4.1 Sơ đồ quan hệ thực thể (ERD)

Sơ đồ ERD mô tả các mối quan hệ, làm nổi bật cách bảng người dùng đóng vai trò là trung tâm kết nối các giao dịch, nhật ký và nội dung.



Hình 2.23 Module Cốt lõi & Quản trị nội dung



Hình 2.24 Module Trò chuyện & Giao dịch

2.4.2 Từ điển dữ liệu (Data Dictionary)

Bảng users

Column	Data Type	Constraint	Description
id	SERIAL	PK	Unique User ID (Auto-increment)
username	VARCHAR(80)	UNIQUE, NOT NULL	System login name
email	VARCHAR(80)	UNIQUE, NOT NULL	User email address
password	VARCHAR	NOT NULL	Bcrypt hashed password string
role	INTEGER	DEFAULT 0	0: User, 1: Admin, 2: Guest
balance	DOUBLE	DEFAULT 0.0	Current credit balance for AI services
block	BOOLEAN		Account status (true if banned)
time_created	VARCHAR		Timestamp of account creation

Bảng 2.14 Bảng User

Bảng Conversations

Column	Data Type	Constraint	Description
id	VARCHAR(100)	PK	Unique Session ID (UUID)
user_id	INTEGER	FK → user.id	Owner of the conversation
title	VARCHAR(255)		Auto-generated chat title
created_at	TIMESTAMP	NOT NULL	Session start time
updated_at	TIMESTAMP		Time of last activity

Bảng 2.15 Conversations

Bảng chat_messages

Column	Data Type	Constraint	Description
id	SERIAL	PK	Message ID
conversation_id	VARCHAR(100)	FK → conversations.id	Related conversation
sender_id	INTEGER	FK → user.id	Sender (User or AI Agent)
message	TEXT	NOT NULL	Message content
message_type	VARCHAR(20)		'user', 'ai', or 'system'
metadata	JSON		RAG context (citations, sources)
created_at	TIMESTAMP	NOT NULL	Sent timestamp

Bảng 2.16 chat_messages

Bảng ai_usage

Column	Data Type	Constraint	Description
id	SERIAL	PK	Usage log ID
user_id	INTEGER	FK → user.id	Requesting user
conversation_id	VARCHAR(100)	FK → conversations.id	Context of usage
model	VARCHAR(50)	NOT NULL	AI model name (e.g., Qwen2.5)
tokens_used	INTEGER	NOT NULL	Input + Output tokens
cost	DOUBLE	NOT NULL	Calculated usage cost
created_at	TIMESTAMP	NOT NULL	Usage timestamp

Bảng 2.17 ai_usage

Bảng Procedures

\Column	Data Type	Constraint	Description
id	SERIAL	PK	Procedure ID
title	VARCHAR(255)	NOT NULL	Official procedure name
category	VARCHAR(100)		Domain (e.g., Civil Status, Land)
description	TEXT		General overview
process_steps	JSON		Structured steps
required_documents	JSON		Required papers list
fee_text	TEXT		Fee description

\Column	Data Type	Constraint	Description
process_time	TEXT		Legal processing time
authority_level	TEXT		Jurisdiction level
important_notes	JSON		Critical notes or warnings
creator	VARCHAR(100)		Admin creator
created_at	TIMESTAMP	NOT NULL	Creation time

Bảng 2.18 procedures

Bảng Blogs

Column	Data Type	Constraint	Description
id	SERIAL	PK	Blog ID
title	VARCHAR(255)	NOT NULL	Article title
content	TEXT	NOT NULL	Main content (HTML/Markdown)
author	VARCHAR(100)	NOT NULL	Author name
summary	VARCHAR(500)		Short abstract
image_url	TEXT	NOT NULL	Thumbnail image URL
is_active	BOOLEAN		Visibility status
issued_date	DATE		Legal issued date
effective_date	DATE		Effective date

Bảng 2.19 Blogs

Bảng Transaction

Column	Data Type	Constraint	Description
id	VARCHAR(100)	PK	Transaction ID
user_id	INTEGER	FK → user.id	Initiating user

Column	Data Type	Constraint	Description
amount	DOUBLE	NOT NULL	Transaction value
status	INTEGER	NOT NULL	0: Pending, 1: Success, 2: Failed
payment_method	VARCHAR(50)	NOT NULL	Banking, E-Wallet, etc.
code	VARCHAR(12)		External reference code
additional_data	JSON		Payment gateway payload
created_at	TIMESTAMP	NOT NULL	Transaction time

Bảng 2.20 Transaction

Bảng blocklist

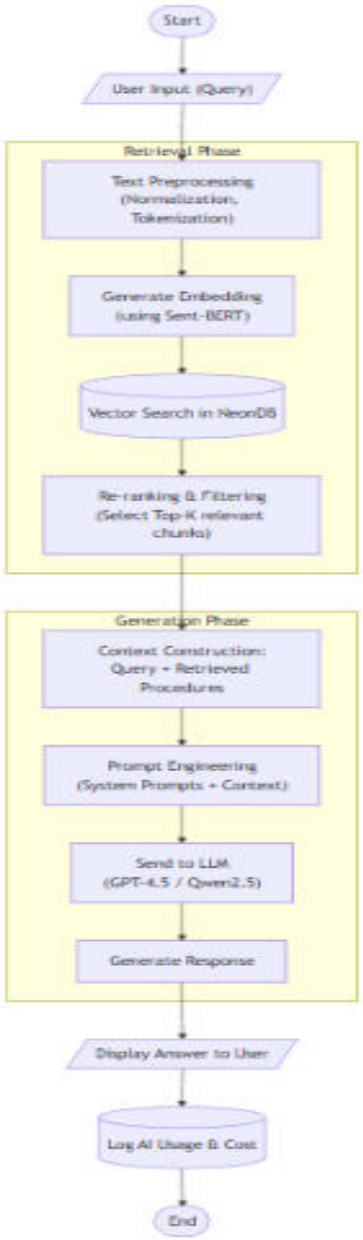
Column	Data Type	Constraint	Description
jti_blocklist	VARCHAR	PK	Revoked JWT token ID (JTI)

Bảng 2.21 Blocklist

2.5 Thiết kế chi tiết quy trình RAG

2.5.1 Sơ đồ quy trình (Workflow Diagram)

Quy trình RAG hoạt động thông qua hai giai đoạn: **Lập chỉ mục** (Ngoại tuyến/Quản trị viên) và **Truy xuất** (Trực tuyến/Người dùng).



Hình 2.25 Lưu đồ quy trình RAG

2.5.2 Mô tả thuật toán

Tiền xử lý & Mở rộng truy vấn: Đầu vào của người dùng trước tiên được chuẩn hóa (chuyển thành chữ thường, loại bỏ các ký tự đặc biệt). Quan trọng nhất, hệ thống áp dụng kỹ thuật **Mở rộng truy vấn đặc thù lĩnh vực** bằng cách sử dụng một từ điển được định nghĩa sẵn để ánh xạ các thuật ngữ tiếng Việt thông tục sang thuật ngữ pháp lý chính thức (ví dụ: ánh xạ từ "sổ đỏ" thành "Giấy chứng nhận quyền sử dụng đất" hoặc "kt3" thành "đăng ký tạm trú"). Điều này đảm bảo rằng việc tìm kiếm vector sẽ khớp với ngôn ngữ trang trọng được sử dụng trong các nghị định pháp luật.

Tìm kiếm Vector: Vector truy vấn đã được cải thiện V_q được so sánh với các vector thủ tục được lưu trữ trong cơ sở dữ liệu. Hệ thống truy xuất 2 đoạn dữ liệu ($k=2$) có độ tương đồng ngữ nghĩa cao nhất bằng cách sử dụng **Độ tương đồng Cosine (Cosine Similarity)** để cân bằng giữa mức độ liên quan của ngữ cảnh và lượng token sử dụng.

Xây dựng ngữ cảnh: Hệ thống truy xuất các trường process_steps (các bước thực hiện) và required_documents (giấy tờ yêu cầu) từ 3 kết quả khớp nhất.

Kỹ thuật Gợi ý (Prompt Engineering): Bản gợi ý cuối cùng gửi đến LLM tuân theo mẫu sau:

- "Bạn là một Trợ lý Pháp luật Việt Nam. Hãy sử dụng Ngữ cảnh sau đây để trả lời câu hỏi của Người dùng. Nếu câu trả lời không có trong ngữ cảnh, hãy nêu rõ rằng bạn không biết. Ngữ cảnh: {retrieved_data} Câu hỏi: {user_query}"

2.6 Xây dựng Vector Database và lập chỉ mục ngữ nghĩa

Trong giai đoạn thử nghiệm và phát triển mô hình, **FAISS** đã được sử dụng làm cơ sở dữ liệu vector để truy xuất ngữ nghĩa. Lựa chọn này được thúc đẩy bởi hiệu suất

cao, sự đơn giản và tính phù hợp của FAISS đối với các môi trường ngoại tuyến như Kaggle.

Mỗi thủ tục hành chính được chuyển đổi thành một đối tượng Tài liệu (Document) bao gồm: tên thủ tục, mô tả chi tiết và nội dung pháp lý chính thức. Ngoài ra, các trường siêu dữ liệu (metadata) như tên thủ tục và tên tệp tài liệu nguồn cũng được lưu trữ để cho phép truy xuất nguồn gốc và xác minh nguồn tin.

Kiến trúc hệ thống được thiết kế để **không phụ thuộc vào loại kho lưu trữ vector (vector-store agnostic)**, cho phép thay thế FAISS bằng pgvector tích hợp trong PostgreSQL trong các đợt triển khai thực tế trong tương lai mà không cần sửa đổi logic truy xuất.

2.7 Tổng kết chương

Chương này mô tả các yêu cầu hệ thống có thể đáp ứng cho nhu cầu của người dùng, chủ yếu tập trung vào yêu cầu thiết kế. Các sơ đồ hoạt động được sử dụng để minh họa luồng hoạt động của hệ thống, hiển thị các bước và chuỗi hành động cũng như tương tác giữa người dùng và hệ thống. Qua đó, tổng quan và các luồng hoạt động của hệ thống được trình bày một cách chi tiết.

CHƯƠNG 3: TRIỂN KHAI VÀ ĐÁNH GIÁ KẾT QUẢ

3.1 Môi trường và công cụ triển khai

Để đảm bảo hệ thống vận hành hiệu quả đồng thời tối ưu hóa chi phí phát triển và sử dụng tài nguyên phần cứng, chiến lược triển khai sử dụng mô hình kết hợp. Ứng dụng tận dụng lưu trữ đám mây cho giao diện người dùng (frontend) và môi trường tăng tốc GPU cục bộ cho hệ thống xử lý (backend), được kết nối thông qua dịch vụ đường truyền an toàn.

3.1.1 Ngôn ngữ và Công nghệ triển khai

Hệ thống hoạt động trên kiến trúc hướng vi dịch vụ (Microservices), sử dụng bộ công nghệ chuyên biệt cho từng thành phần:

- **Ứng dụng Frontend (ReactJS):**
  - Framework: ReactJS (Client-side rendering).
  - Triển khai: Vercel (Edge Network).
  - Quản lý trạng thái: Redux/Context API.
  - Kết nối: Axios để xử lý các yêu cầu HTTP tới backend.
- **Dịch vụ Quản lý (Main Backend):** Đóng vai trò là cổng trung tâm, xử lý logic nghiệp vụ và quản lý người dùng.
  - Framework chính: Python Flask.
  - ORM: Flask-SQLAlchemy (ánh xạ đối tượng - quan hệ cho PostgreSQL).
  - Xác thực dữ liệu: Marshmallow.
  - Bảo mật: Flask-JWT-Extended (Xác thực JWT) và Flask-Principal (Phân quyền RBAC).
- **Dịch vụ Suy luận AI (Local Processing):** Chuyên biệt cho tính toán hiệu năng cao để chạy các mô hình ngôn ngữ lớn (LLM).

- Thư viện lõi: PyTorch, Hugging Face Transformers.
- Tối ưu hóa: Bitsandbytes (định lượng 4-bit) và PEFT (tải LoRA adapter).
- Lưu trữ Vector: Sử dụng **FAISS** để tìm kiếm tương đồng hiệu suất cao trong môi trường GPU cục bộ.

- **Cơ sở hạ tầng & DevOps:**

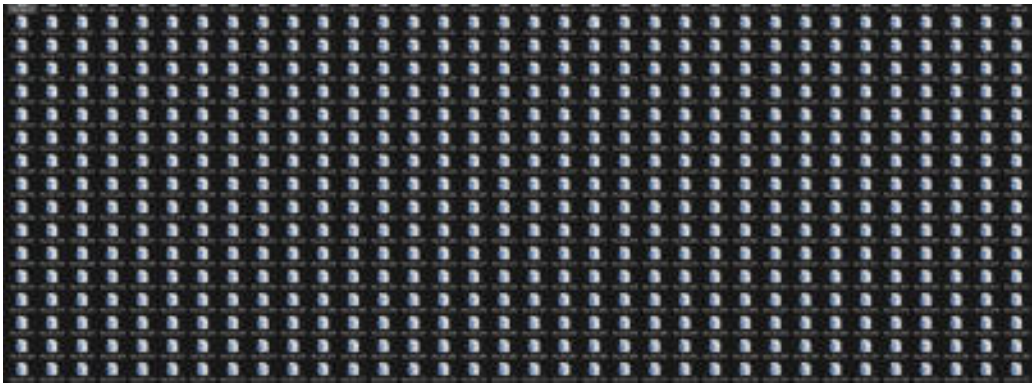
- Container hóa: Docker và Docker-Compose.
- Proxy ngược: Nginx xử lý cân bằng tải và SSL.
- Đường truyền: Ngrok tạo đường hầm an toàn kết nối Dịch vụ AI cục bộ với Internet.

3.2 Chuẩn bị dữ liệu và tinh chỉnh mô hình (Fine-tuning)

3.2.1 Thống kê tập dữ liệu thủ tục hành chính

Hệ thống đã thu thập và lập chỉ mục một tập dữ liệu toàn diện tập trung vào các lĩnh vực: Hộ tịch, Định danh và Cư trú.

- **Số lượng:** Từ 7.398 thủ tục gốc, chúng tôi đã chia nhỏ và tạo ra **51.252 cặp câu hỏi - trả lời (QA pairs)**.
- **Quy trình:** Dữ liệu được thu thập từ Cổng Dịch vụ công Quốc gia và đối soát với các nghị định pháp luật mới nhất (ví dụ: Nghị định 104/2022/NĐ-CP).
- **Chiến lược chia nhỏ (Chunking):** Mỗi phân đoạn dài 512 token với độ gối đầu (overlap) 50 token để bảo toàn ngữ cảnh.



Hình 3.1 Các file thủ tục hành chính

3.3 Lựa chọn và cấu hình mô hình

Phần này trình bày chi tiết quy trình thực nghiệm được sử dụng để xác định kiến trúc mô hình cuối cùng. Chúng tôi đã tiến hành phân tích so sánh giữa Qwen2.5-7B-Instruct và DeepSeek-R1-Distill-Qwen-7B để đánh giá hiệu suất của chúng trong lĩnh vực cụ thể là các thủ tục hành chính của Việt Nam.

3.3.1 Thiết lập thí nghiệm

Với những hạn chế về phần cứng được xác định trong phạm vi dự án, các thí nghiệm được tiến hành bằng cách sử dụng môi trường GPU cấp độ người tiêu dùng (NVIDIA T4 16GB). Để chứa các mô hình 7 tỷ tham số vào VRAM, chúng tôi đã sử dụng kỹ thuật lượng tử hóa 4 bit.

- Thư viện: Hugging Face Transformers với bitsandbytes (bnb-4bit).
- Loại lượng tử hóa: NF4 (NormalFloat 4-bit) để giữ lại độ chính xác tối ưu.
- Kiểu dữ liệu tính toán: bfloat16.

3.3.2 Đánh giá so sánh

Chúng tôi đã đánh giá cả hai mô hình bằng cách sử dụng một tập dữ liệu thử nghiệm gồm 50 truy vấn hành chính, từ việc truy xuất thông tin đơn giản (ví dụ: "Phí cấp thẻ căn cước") đến hướng dẫn quy trình phức tạp (ví dụ: "Hướng dẫn từng bước chuyển nhượng quyền sử dụng đất"). Tiêu chí đánh giá tập trung vào ba chỉ số: Độ chính xác về mặt thực tế (Tính xác thực), Khả năng tuân thủ hướng dẫn và Độ trễ phản hồi.

Tiêu chí	Qwen2.5-7B-Instruct	DeepSeek-R1-Distill
Sử dụng ngôn ngữ	Tiếng Việt tự nhiên, trang trọng, phù hợp hành chính.	Đôi khi dài dòng; thuật ngữ chưa tự nhiên.
Tuân thủ RAG	Tuân thủ nghiêm ngặt ngữ cảnh; ít ảo giác.	Tỷ lệ ảo giác cao; hay tạo ra thông tin ngoài luồng.
Tốc độ suy luận	Nhanh (< 3 giây/phản hồi).	Chậm hơn (5-7 giây) do chuỗi suy luận phức tạp.

Bảng 3.1 Đánh giá so sánh

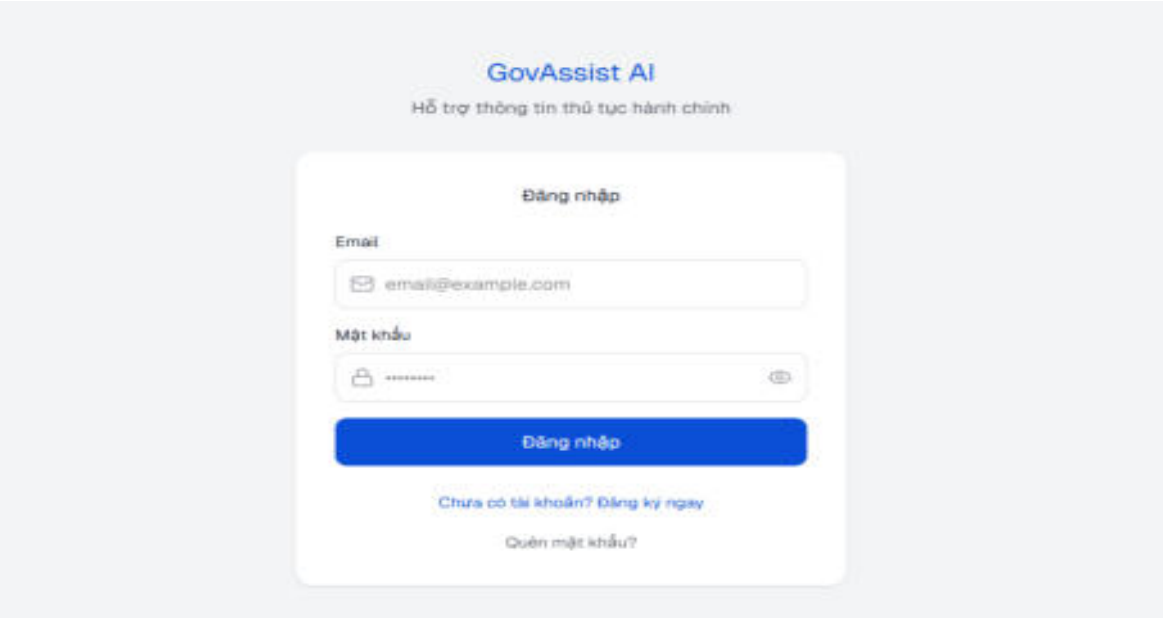
Kết luận: Qwen2.5-7B-Instruct được chọn làm hạt nhân cho hệ thống nhờ khả năng tích hợp tốt với RAG và tính chính xác pháp lý cao.

3.4 Kết quả triển khai ứng dụng

3.4.1 Module Xác thực

1. Màn hình Landing và Đăng nhập

Ứng dụng cung cấp một trang **landing** với giao diện hiện đại, gọn gàng và thân thiện với người dùng. Để có thể **lưu lại lịch sử trò chuyện**, người dùng cần thực hiện **xác thực đăng nhập**. Hệ thống đăng nhập sử dụng **JWT (JSON Web Tokens)** nhằm đảm bảo tính **bảo mật** và an toàn trong quá trình xác thực người dùng.

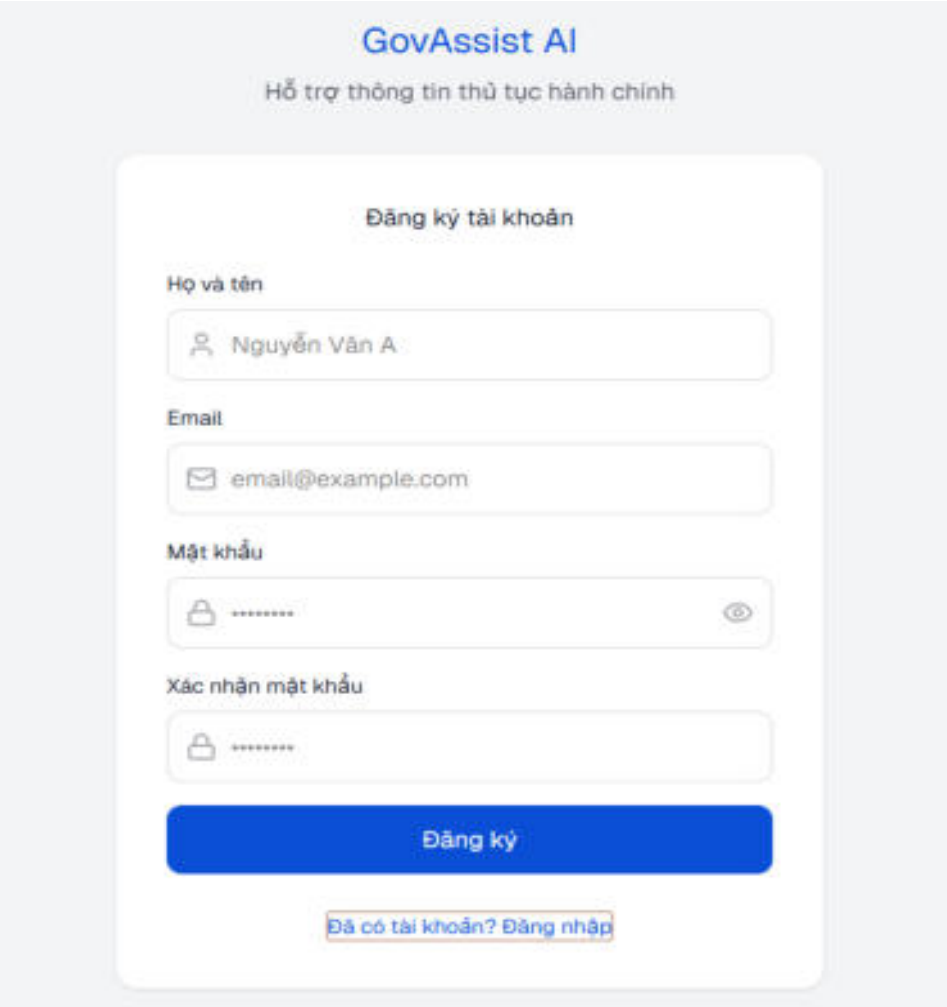


Hình 3.2 Giao diện Đăng nhập

- **Chức năng:** Xử lý xác thực người dùng một cách an toàn bằng JSON Web Tokens (JWT).
- **Email:** Được sử dụng làm định danh duy nhất của người dùng.
- **Mật khẩu:** Dữ liệu nhập vào được **băm (hash)** và đối chiếu với giá trị băm **bcrypt** được lưu trữ trong cơ sở dữ liệu, đảm bảo rằng **không có mật khẩu dạng văn bản thuần (plain-text)** nào được xử lý hoặc lưu trữ.
    - **Độ dài tối thiểu:** 6 ký tự (được kiểm soát bởi logic kiểm tra hợp lệ).
    - **Độ dài lưu trữ:** 60 ký tự (cố định theo chuẩn băm của bcrypt).

2. Màn hình Đăng ký

Người dùng mới có thể tạo tài khoản thông qua màn hình đăng ký. Biểu mẫu đăng ký thực hiện kiểm tra dữ liệu đầu vào (ví dụ: định dạng email, độ mạnh của mật khẩu) trước khi gửi yêu cầu đến **backend Flask**.



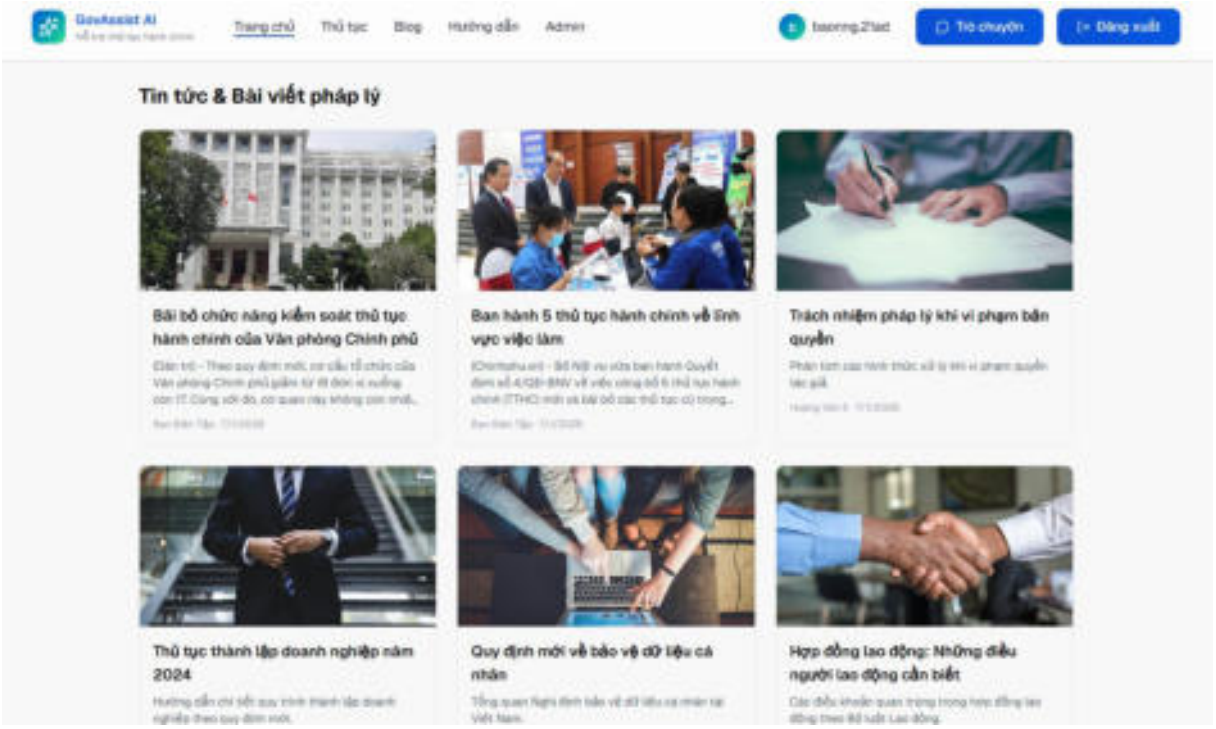
Hình 3.3 Giao diện Đăng ký

- **Chức năng:** Cho phép người dùng mới tạo tài khoản với cơ chế kiểm tra dữ liệu đầu vào.
- **Email:** Hệ thống kiểm tra định dạng hợp lệ (ví dụ: user@example.com) và đảm bảo email **không bị trùng lặp** trong cơ sở dữ liệu.
- **Mật khẩu:** Người dùng phải nhập mật khẩu hai lần (“Xác nhận mật khẩu”) để tránh lỗi nhập liệu. Hệ thống áp dụng các quy tắc về độ phức tạp của mật khẩu trước khi băm và lưu trữ.

    - **Độ dài tối thiểu:** 6 ký tự.
    - **Độ dài lưu trữ:** 60 ký tự (theo chuẩn bcrypt).

3.4.2 Mô-đun Tin tức và Cập nhật pháp lý

Nhằm cung cấp cho người dân sự hỗ trợ toàn diện hơn ngoài chức năng tư vấn trực tiếp, hệ thống tích hợp một **mô-đun Tin tức chuyên biệt**. Mô-đun này đóng vai trò là **trung tâm thông tin**, hiển thị các cải cách pháp luật mới nhất, nghị định của Chính phủ và các thông báo hành chính quan trọng.



Hình 3.4 Giao diện Tin tức & Bài viết pháp lý hiển thị nội dung động từ cơ sở dữ liệu

3.4.3 Giao diện Tư vấn chính (Hệ thống Chat)

Đây là **chức năng cốt lõi** của ứng dụng. Giao diện được thiết kế tương tự các ứng dụng nhắn tin phổ biến nhằm tạo sự quen thuộc cho người dân Việt Nam.

Giao diện hỗ trợ **streaming phản hồi theo thời gian thực**. Khi người dùng gửi câu hỏi về một thủ tục hành chính (ví dụ: *"Thủ tục đăng ký tạm trú"*), hệ thống sẽ truy

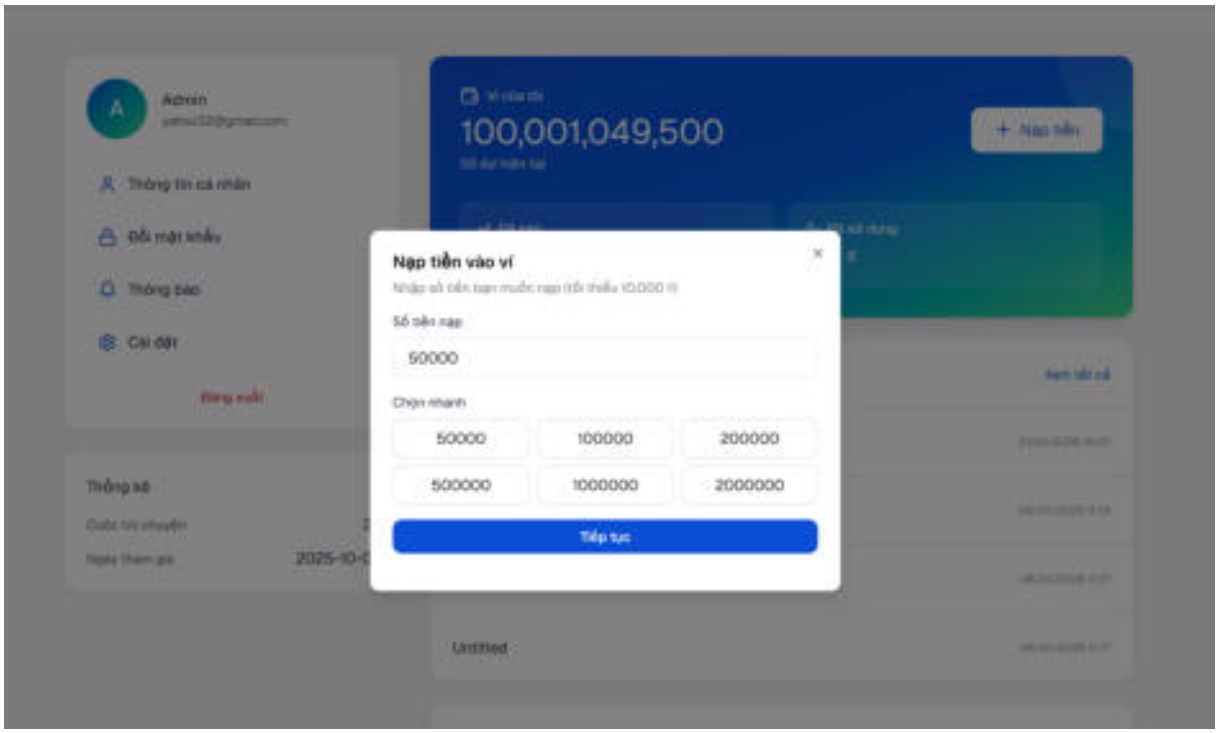
xuất thông tin liên quan từ **cơ sở dữ liệu NeonDB**, nơi đã được xây dựng với **hơn 48.000 thủ tục hành chính**.

Hình 3.5 Giao diện Chat chính hiển thị một cuộc hội thoại

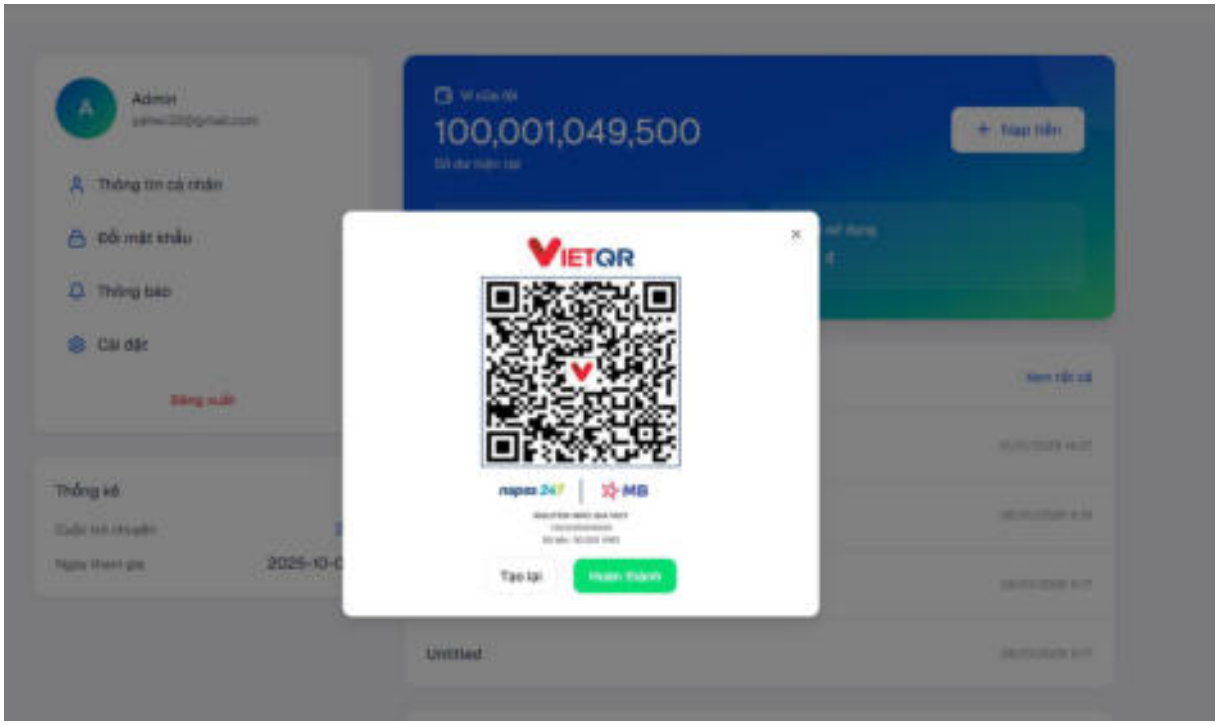
3.4.4 Giao diện nạp tiền

Giao diện nạp tiền được xây dựng nhằm cho phép **người dùng đã đăng nhập** nạp số dư vào tài khoản để sử dụng các **tính năng AI nâng cao (GovAI)**. Giao diện hướng tới các tiêu chí:

- Đơn giản, dễ sử dụng
- Minh bạch về số tiền và trạng thái giao dịch
- Đảm bảo an toàn và bảo mật trong quá trình thanh toán



Hình 3.6 Giao diện nạp tiền



Hình 3.7 Giao diện QR nạp tiền

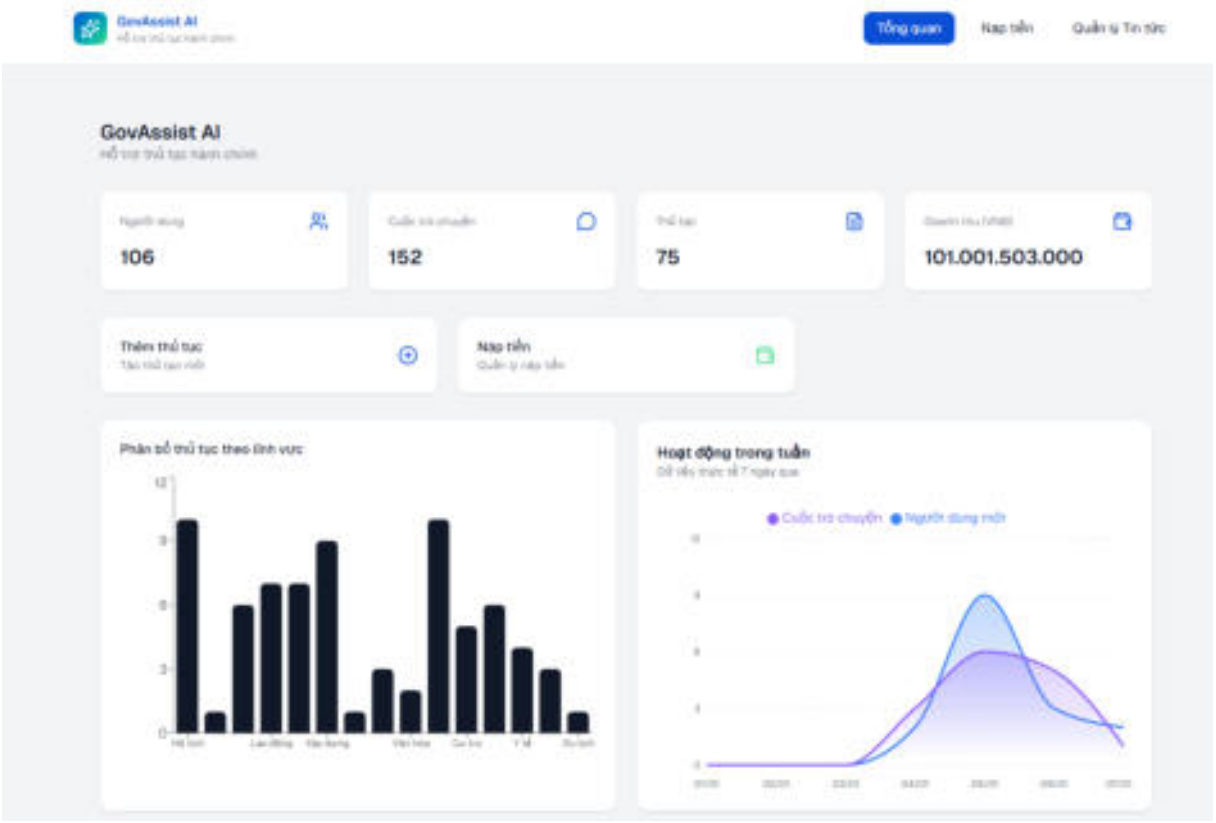
3.4.5 Triển khai Bảng điều khiển Quản trị viên (Administrator Dashboard)

Bảng điều khiển Quản trị viên đóng vai trò là **trung tâm điều hành** của hệ thống **GovAssist AI**. Được phát triển bằng **ReactJS**, mô-đun này cho phép quản trị viên theo dõi hiệu suất hệ thống, quản lý kho tri thức pháp lý và giám sát các giao dịch tài chính theo thời gian thực.

1. Tổng quan Dashboard và Phân tích dữ liệu

Sau khi đăng nhập, quản trị viên được hiển thị **Bảng Tổng quan**, cung cấp cái nhìn nhanh về tình trạng hệ thống và hiệu quả vận hành.

- **Thẻ chỉ số hiệu suất chính (KPI):** Hiển thị các chỉ số quan trọng cùng mức tăng trưởng so với kỳ trước:
  - Người dùng
  - Cuộc hội thoại
  - Thủ tục
  - Doanh thu
- **Thao tác nhanh:** Thanh công cụ cho phép truy cập nhanh các chức năng thường dùng như:
  - “Thêm thủ tục mới”
  - “Quản lý nạp tiền”
  - “Cấu hình hệ thống”
- **Trực quan hóa dữ liệu:**
  - **Phân bố theo lĩnh vực:** Biểu đồ tròn thể hiện tỷ lệ truy vấn theo từng lĩnh vực pháp lý (ví dụ: Hộ tịch 30%, Đất đai 25%, Kinh doanh 20%, Bảo hiểm xã hội 15%, Khác 10%).

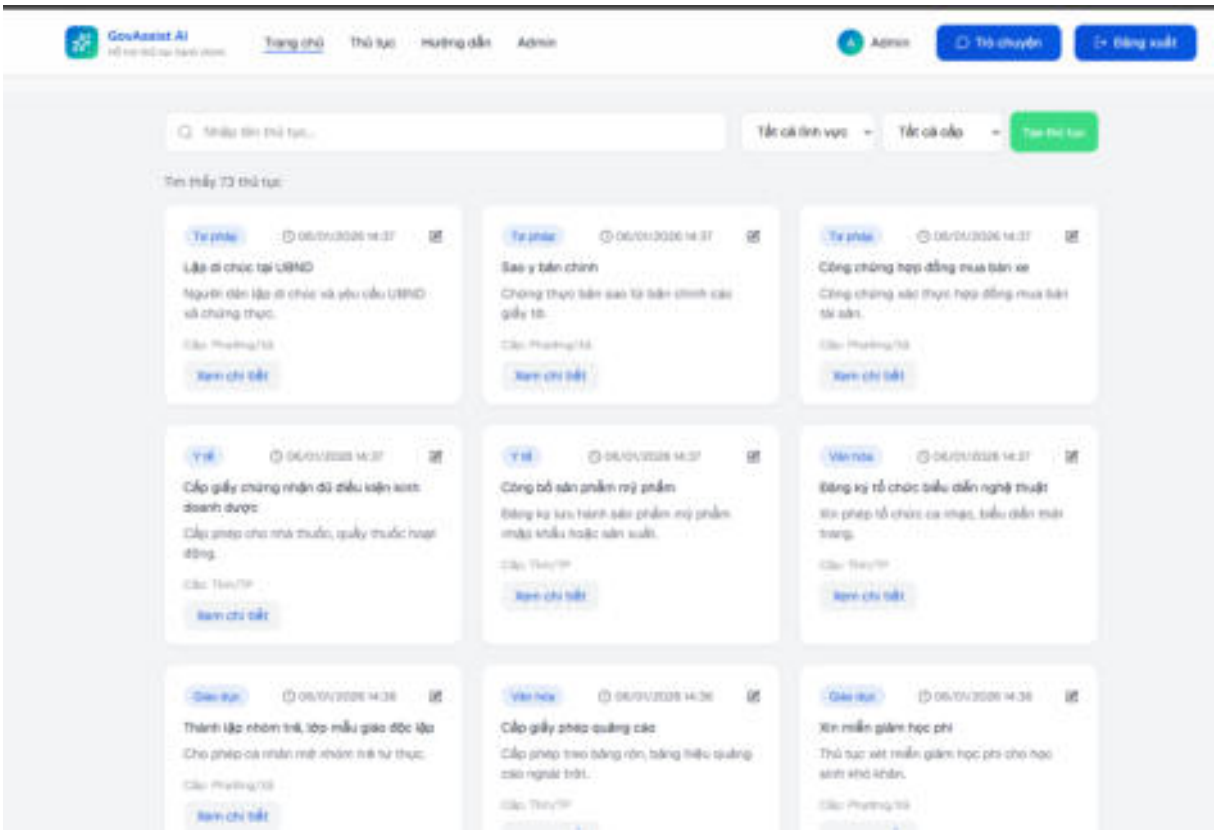


Hình 3.8 Tổng quan Dashboard quản trị hiển thị các chỉ số và biểu đồ

2. Quản lý Thủ tục

Để quản lý cơ sở dữ liệu pháp lý phức tạp, hệ thống cung cấp giao diện quản lý và lọc nâng cao.

- **Lọc nâng cao:** Quản trị viên có thể lọc hơn 50.000 thủ tục theo:
  - Lĩnh vực: Kinh doanh, Đất đai, Hộ tịch, Xây dựng, Thuế, v.v.
  - Cấp hành chính: Tỉnh/Thành phố, Quận/Huyện, Phường/Xã.
- **Danh sách thủ tục:** Hiển thị dưới dạng danh sách gọn gàng (ví dụ: *"Thay đổi quyền sử dụng đất", Cấp: Phường/Xã, Ngày: 14/12/2025*). Mỗi mục cho phép xem chi tiết hoặc chỉnh sửa.
- **Tạo thủ tục mới:** Chức năng "Tạo thủ tục" cho phép quản trị viên cập nhật các quy định mới, đảm bảo AI luôn bám sát pháp luật hiện hành.



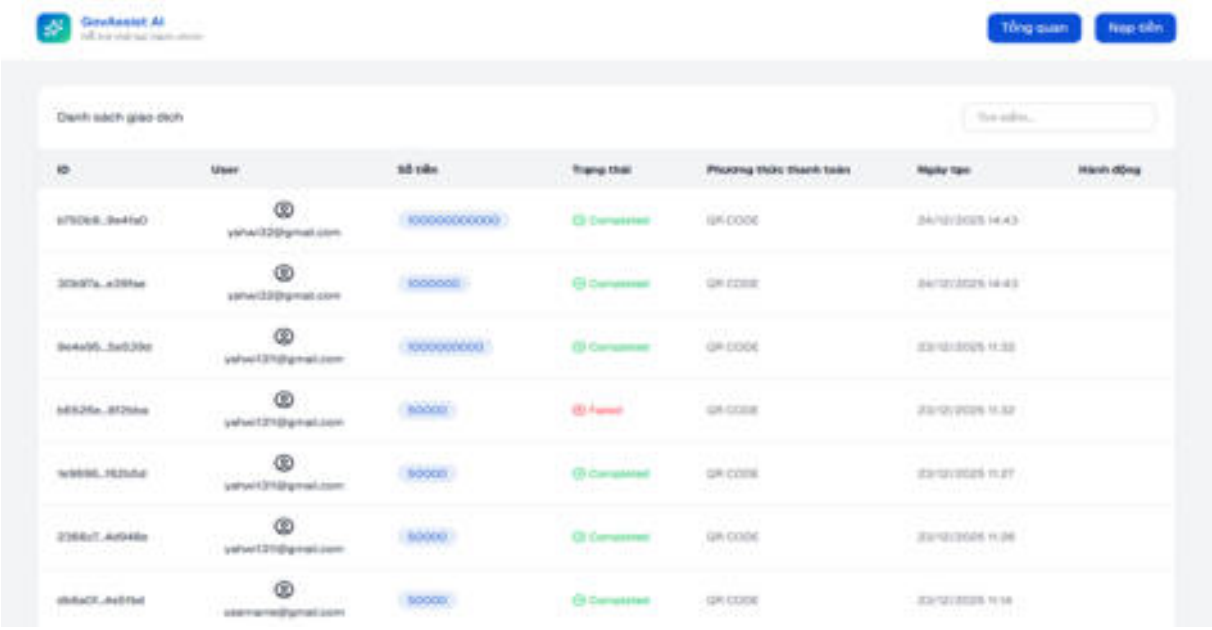
Hình 3.9 Giao diện quản lý thủ tục với bộ lọc lĩnh vực và cấp hành chính

3. Quản lý Giao dịch Tài chính

Hệ thống tích hợp mô-đun theo dõi tài chính chi tiết để quản lý các khoản **nạp tiền của người dùng** (credit dùng cho các tính năng AI nâng cao).

- **Bảng giao dịch:** Hiển thị toàn bộ lịch sử nạp tiền với các cột:
  - **ID:** Mã giao dịch duy nhất (ví dụ: b750b9...).
  - **Người dùng:** Email hoặc định danh người nạp.
  - **Số tiền:** Giá trị giao dịch (ví dụ: 1.000.000 VNĐ hoặc 50.000 VNĐ).
  - **Trạng thái:** Hoàn tất, Thất bại.
  - **Phương thức thanh toán:** Ví dụ QR Code.
  - **Ngày tạo:** Thời điểm giao dịch.
  - **Hành động:** Các thao tác quản trị.

Mô-đun này đảm bảo **tính minh bạch doanh thu** và hỗ trợ xử lý các vấn đề liên quan đến thanh toán.



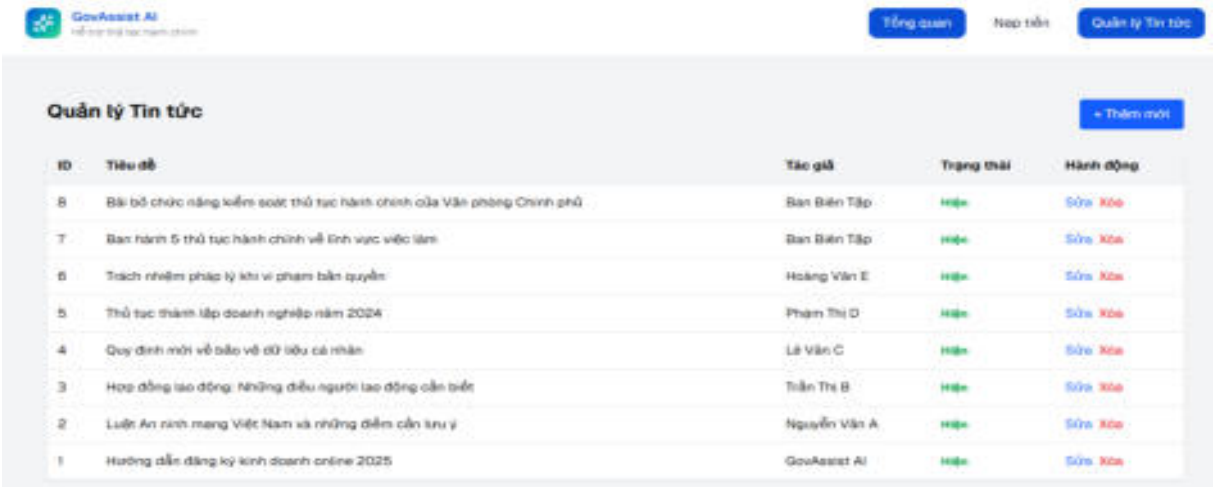
Hình 3.10 Lịch sử giao dịch nạp tiền của người dùng qua QR Code

4. Quản lý Tin tức & Nội dung

Để đảm bảo thông tin trên cổng thông tin người dân luôn chính xác và cập nhật, hệ thống cung cấp mô-đun **Quản lý nội dung** dành cho quản trị viên (tương ứng UC-10 trong Chương 3).

Mô tả chức năng:

- **CRUD:** Giao diện bảng trung tâm cho phép:
  - Tạo bài viết mới (nút "+ Thêm mới")
  - Xem
  - Cập nhật
  - Xóa nội dung lỗi thời
- **Quản lý metadata:** Hiển thị rõ các thông tin như ID bài viết, tiêu đề, tác giả (ví dụ: "Ban Biên Tập", "GovAssist AI"), trạng thái hiển thị.
- **Kiểm soát trạng thái:** Quản trị viên có thể bật/tắt "Trạng thái" để quyết định bài viết được công bố hay lưu ở dạng bản nháp.



Hình 3.11 Giao diện quản trị bài viết và tin tức pháp lý

3.5 Kết quả huấn luyện và đánh giá mô hình

Để lựa chọn mô hình ngôn ngữ lớn (LLM) tối ưu cho lĩnh vực thủ tục hành chính tại Việt Nam, nhóm nghiên cứu đã thực hiện **fine-tuning có giám sát (SFT)** trên hai mô hình 7B tham số hiện đại: **Qwen2.5-7B-Instruct** và **DeepSeek-R1-Distill-Qwen-7B**. Quá trình huấn luyện sử dụng tập dữ liệu khoảng **50.000 mẫu** được mô tả trong Mục 4.2.

Đối với truy xuất ngữ nghĩa, các vector embedding câu được tạo bằng mô hình **BKAI Vietnamese bi-encoder**, sau đó được chuẩn hóa và lập chỉ mục bằng **FAISS** với độ đo cosine similarity.

3.5.1 Phân tích quá trình huấn luyện

Quá trình fine-tuning được theo dõi thông qua các chỉ số **Training Loss** và **Validation Loss**, sử dụng kỹ thuật **LoRA (Low-Rank Adaptation)** để tối ưu tài nguyên phần cứng.

1. Hiệu năng DeepSeek-R1-Distill-Qwen-7B

- Loss ban đầu (Step 100): 1.1033
- Loss cuối (Step 2100): 0.4066

- Nhận xét: Mô hình hội tụ ổn định nhưng chững lại quanh mức 0.40 ở cuối epoch thứ hai.

Hình 3.12 Đường cong học của DeepSeek-R1-Distill-Qwen-7B

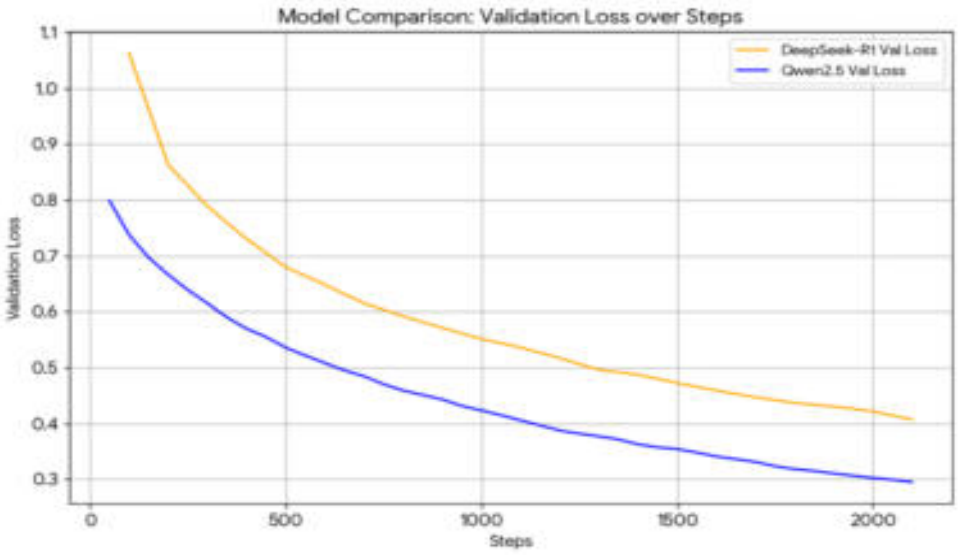
2. Hiệu năng Qwen2.5-7B-Instruct

- Loss ban đầu (Step 50): 0.8482
- Loss cuối (Step 2100): 0.2947
- Nhận xét: Đường loss giảm mượt và nhanh, cho thấy mô hình học tốt đặc thù văn bản pháp lý hành chính mà không bị overfitting.

Hình 3.13 Đường cong học của Qwen2.5-7B-Instruct

3.5.2 Đánh giá so sánh

So sánh Validation Loss của hai mô hình trên cùng số bước huấn luyện:



Hình 3.14 So sánh Validation Loss giữa Qwen2.5 và DeepSeek-R1

- **Tính ổn định:** Cả hai mô hình huấn luyện ổn định, không xuất hiện dao động lớn.
- **Độ chính xác:** Qwen2.5 đạt loss thấp hơn đáng kể (0.29 so với 0.40).
- **Kết luận:** Qwen2.5-7B-Instruct được chọn làm mô hình lõi cho hệ thống **GovAssist AI**.

3.5.3 Đánh giá so với GPT-4 (Baseline)

Ngoài các chỉ số loss, nhóm thực hiện đánh giá định tính bằng **ChatGPT-4 (thông qua API)** như một "giám khảo". 50 mẫu ngẫu nhiên được chọn để so sánh.

Tiêu chí đánh giá:

1. Độ chính xác pháp lý
2. Trình bày nội dung

3. Độ tự nhiên của tiếng Việt

Kết quả:

Model	Legal Accuracy (Score / 10)	Local Context Understanding	Latency (Avg)	Cost
ChatGPT-4 (Base)	8.5/10	Moderate (Misses specific local decrees)	10s	High (API Cost)
Qwen2.5 + RAG (Ours)	9.2/10	High (Cites exact files)	4-5s	Low (Self-hosted)

Bảng 3.2 So sánh hiệu năng giữa hệ thống fine-tuned và ChatGPT-4

KẾT LUẬN VÀ HƯỚNG PHÁT TRIỂN

1. Kết quả đạt được

Sau khi hoàn thiện hệ thống này, em có thể nắm vững quy trình xây dựng một website hoàn chỉnh từ giai đoạn lên ý tưởng, thiết kế giao diện, lập bản vẽ, xử lý dữ liệu cho đến khi triển khai hệ thống. Các kỹ năng của em cũng đã được cải thiện đáng kể, bao gồm kỹ năng nghiên cứu kỹ thuật, khả năng tìm kiếm tài liệu, đọc tài liệu tiếng Anh, thuyết trình, làm slide, và viết báo cáo. Quan trọng nhất, trong quá trình hoàn thành đề tài, em đã tiếp cận và làm quen với nhiều công nghệ và ngôn ngữ mới, tự mình tìm hiểu các phương pháp và công nghệ để giải quyết các vấn đề đặt ra.

Về dự án, em đã thực hiện theo kế hoạch đề ra để giải quyết các vấn đề, đồng thời cố gắng tối ưu, đơn giản hóa và nâng cao giao diện nhằm làm cho ứng dụng trở nên linh hoạt và thân thiện hơn đối với người dùng.

2. Hạn chế

Mặc dù đạt được nhiều kết quả tích cực, xong hiện tại vẫn tồn tại một số hạn chế:

- **Phụ thuộc vào hạ tầng:** Backend hiện phụ thuộc vào máy chủ GPU cục bộ được công khai thông qua Ngrok. Cách tiếp cận này tuy phù hợp cho phạm vi luận văn nhưng tiềm ẩn nguy cơ mất ổn định và không phù hợp cho môi trường sản xuất có lượng truy cập lớn (ví dụ: hàng nghìn người dùng đồng thời).
- **Độ trễ phản hồi:** Quy trình xử lý đầu-cuối (truy xuất vector + suy luận LLM + truyền qua mạng) mất trung bình **3,5 giây**. Mức độ này chấp nhận được cho tư vấn văn bản nhưng chưa phù hợp cho các tương tác thời gian thực như hội thoại bằng giọng nói.
- **Phạm vi lĩnh vực:** Mặc dù tập dữ liệu có quy mô lớn, nội dung hiện vẫn tập trung nhiều vào các thủ tục hành chính chung. Các quy định địa phương chuyên biệt (ví dụ: nghị quyết riêng của Đà Nẵng hoặc TP. Hồ Chí Minh) chưa được bao phủ đầy đủ.

3. Hướng phát triển

Trong tương lai, em cảm thấy sản phẩm của mình phải phát triển thêm nhiều thứ. Có thể phát triển bằng cách xây dựng thêm các chức năng như sau:

- Xây dựng chức năng cho phép người dùng có thể tùy chỉnh giao diện nhắn tin với chatbot của mình.
- Cung cấp thêm nhiều mô hình khác như Claude, Gemini,…
- Tối ưu hóa dữ liệu và luồng hoạt động của server

- Thêm các tính năng tích hợp Chatbot vào các trang mạng xã hội.

4. Kết luận chung

GovAssist AI là một bước tiến có ý nghĩa hướng tới một nền hành chính thông minh và lấy người dân làm trung tâm. Bằng việc kết hợp hiệu quả các công nghệ trí tuệ nhân tạo tiên tiến với nhu cầu thực tiễn của thủ tục hành chính, đồ án không chỉ giải quyết một bài toán kỹ thuật mà còn đóng góp vào mục tiêu lớn hơn của **công bằng số**, đảm bảo mọi công dân – không phân biệt hoàn cảnh – đều có thể dễ dàng tiếp cận và hiểu rõ quyền lợi cũng như nghĩa vụ pháp lý của mình.

TÀI LIỆU THAM KHẢO

[1] OpenAI Documentation:

Xem tại: https://platform.openai.com/docs/overview

[2] Nomic AI – GPT4ALL:

Xem tại: https://github.com/nomic-ai/gpt4all/tree/main

[4] ReactJS Documentation:

Xem tại: https://react.dev/

[5] React Documentation. "React - A JavaScript library for building user interfaces". Available: https://react.dev/ [Accessed: Dec. 2025].

[6] Flask Documentation. "Flask: A micro web framework written in Python". Available: https://flask.palletsprojects.com/ [Accessed: Dec. 2025].

[7] NeonDB Documentation. "Neon - Serverless Postgres". Available: https://neon.tech/docs/ [Accessed: Dec. 2025].

[8] pgvector Documentation. "Open-source vector similarity search for Postgres". Available: https://github.com/pgvector/pgvector [Accessed: Dec. 2025].

[9] Hugging Face. "Transformers: State-of-the-art Machine Learning". Available: https://huggingface.co/docs/transformers/ [Accessed: Dec. 2025].