KHOA CÔNG NGHỆ THÔNG TIN

BỘ MÔN CÔNG NGHỆ PHẦN MỀM

Icon

Description automatically generated

ĐỒ ÁN MÔN HỌC

**THIẾT KẾ KIẾN TRÚC HỆ THỐNG**

**Đề tài**

**Phát triển Chatbot hỗ trợ sinh viên tra cứu thông tin**

**và giải đáp thắc mắc**

|  |  |
| --- | --- |
| **Nhóm:** | 12 |
| **Sinh viên thực hiện:** | Hoàng Văn Quyết- 0202366 |
| **Lớp học phần:** | 66PM1 |
| **Giảng viên hướng dẫn:** | Phạm Hữu Tùng |

Mục lục

[**Chương 1. Giới thiệu đề tài** 4](#_Toc199170033)

[1.1. Lý do chọn đề tài 4](#_Toc199170034)

[1.2. Mục tiêu của đồ án 4](#_Toc199170035)

[1.3. Phạm vi và giới hạn 4](#_Toc199170036)

[1.4. Phương pháp thực hiện 4](#_Toc199170037)

[**Chương 2. Phân tích hệ thống** 6](#_Toc199170038)

[2.1. Mô tả bài toán 6](#_Toc199170039)

[2.2. Yêu cầu chức năng 6](#_Toc199170040)

[2.3. Use Case Diagram tổng quát 7](#_Toc199170041)

[2.4. Mô tả các Use Case chính 7](#_Toc199170042)

[2.4.1. Đăng kí 7](#_Toc199170043)

[2.4.2. Đăng nhập 8](#_Toc199170044)

[2.4.3. Đăng xuất 9](#_Toc199170045)

[2.4.4. Nhắn tin 10](#_Toc199170046)

[2.4.5. Thêm cuộc trò chuyện 11](#_Toc199170047)

[2.4.6. Xóa cuộc trò chuyện 11](#_Toc199170048)

[2.5. Biểu đồ hoạt động (*Các chức năng chính*) 13](#_Toc199170049)

[2.5.1. Luồng xử lý tin nhắn người dùng 13](#_Toc199170050)

[2.5.2. Xem lịch sử chat 14](#_Toc199170051)

[2.5.3. Xử lý tài liệu đầu vào 15](#_Toc199170052)

[**Chương 3. Thiết kế hệ thống** 16](#_Toc199170053)

[3.1. Lựa chọn mô hình kiến trúc (Monolithic, Microservices, MVC, Layered, Clean Architecture, ...) 16](#_Toc199170054)

[3.2. Sơ đồ kiến trúc tổng thể 17](#_Toc199170055)

[3.3. Biểu đồ tuần tự (*các chức năng chính*) 18](#_Toc199170056)

[3.3.1. Xử lý tin nhắn người dùng 18](#_Toc199170057)

[3.3.2. Xem lịch sử đoạn chat 18](#_Toc199170058)

[3.3.3. Xử lý tài liệu 19](#_Toc199170059)

[3.4. Thiết kế lớp (Class Diagram) 20](#_Toc199170060)

[3.4.1. User Diagram 20](#_Toc199170061)

[3.4.2. Chat Diagram 21](#_Toc199170062)

[3.5. Thiết kế cơ sở dữ liệu (ERD, bảng, mối quan hệ) 22](#_Toc199170063)

[3.5.1. Bảng users 22](#_Toc199170064)

[3.5.2. Bảng chat\_sessions 22](#_Toc199170065)

[3.5.3. Messages 22](#_Toc199170066)

[3.5.4. Sơ đồ ERD 22](#_Toc199170067)

[3.6. Thiết kế giao diện người dùng (Mockup / Wireframe) 23](#_Toc199170068)

[**Chương 4. Cài đặt và triển khai hệ thống** 24](#_Toc199170069)

[4.1. Môi trường triển khai 24](#_Toc199170070)

[4.1.1. Môi trường Development/Local 24](#_Toc199170071)

[4.1.2. Cấu hình Environment Variables 24](#_Toc199170072)

[4.2. Cài đặt hệ thống 25](#_Toc199170073)

[4.2.1. Yêu cầu hệ thống 25](#_Toc199170074)

[4.2.2. Hướng dẫn cài đặt từng bước 25](#_Toc199170075)

[**Chương 5. Kết quả và đánh giá** 27](#_Toc199170076)

[5.1. Kết quả thử nghiệm hệ thống 27](#_Toc199170077)

[5.2. Đánh giá hiệu quả hệ thống 27](#_Toc199170078)

[**Tài liệu tham khảo** 27](#_Toc199170079)

# Chương 1. Giới thiệu đề tài

## 1.1. Lý do chọn đề tài

- Trong bối cảnh nền giáo dục đang phát triển mạnh mẽ, nhu cầu tiếp cận thông tin một cách nhanh chóng và chính xác ngày càng tăng không chỉ đối với sinh viên mà còn đối với các phụ huynh. Tuy nhiên, các kênh thông tin truyền thống hay các thông báo thường khiến sinh viên cũng như phụ huynh mất thời gian tra cứu và rất dễ bỏ xót thông tin. Chính vì vậy mà em lựa chọn đề tài xây dựng chatbot nhằm hỗ trợ sinh viên và các bậc phụ huynh tiếp cận các thông tin về nhà trường một cách nhanh chóng và chính xác hơn. Với việc phát triển chatbot sẽ giúp cho việc nắm bắt thông tin, quy chế được chủ động và tiện lợi hơn, đồng thời làm giảm tải công việc cho phòng công tác sinh viên về vấn đề giải đáp thắc mắc của sinh viên.

## 1.2. Mục tiêu của đồ án

- Mục tiêu của đồ án sẽ giúp cho sinh viên rèn luyện và nâng cao hiểu biết về góc nhìn kiến trúc hệ thống, áp dụng các kiến thức đã học vào đồ án và mở rộng kiến thức về cách xử lý ngôn ngữ tự nhiên cũng như các công nghệ liên quan khác. Thông qua quá trình tìm hiểu và thực hiện đồ án sinh viên có thể học hỏi được các kĩ năng lập trình cũng như phân tích thiết kế hệ thống. Ngoài ra đây cũng là hành trang chuẩn bị cho công việc sau này của sinh viên. Mong rằng đồ án này không chỉ mang ý nghĩa học tập mà còn giúp sinh viên xây dựng một nền tảng kỹ thuật thật vững chắc phục vụ cho tương lai sau này .

## 1.3. Phạm vi và giới hạn

**a. Phạm vi**:

- Chatbot hỗ trợ người dùng trong việc tra cứu các thông tin liên quan:

* Quy chế đào tạo
* Thông tin học tập
* Thủ tục hành chính
* Thông báo nhà trường

- Chatbot có khả năng giao tiếp bằng tiếng Việt, cho phép người dùng nhập liệu câu hỏi.

**b. Giới hạn:**

- Cơ sở dữ liệu sử dụng là tập hợp các tài liệu quy chế, thông tin trường cung cấp sẵn, không tự động cập nhật dữ liệu mới từ hệ thống trường.

- Không tích hợp các chức năng nâng cao: xác thực tài khoản sinh viên, kết nối với hệ thống đào tạo chính thức…

## 1.4. Phương pháp thực hiện

1. Thu thập và phân tích yêu cầu:

* Thu thập các tài liệu chính thức từ nhà trường: quy chế, thông báo, ….
* Xác định các chức năng chính mà chatbot cần có

2. Thiết kế kiến trúc hệ thống

* Áp dụng các mô hình kiến trúc, design pattern,..
* Lựa chọn các công nghệ phù hợp

3. Xây dựng chatbot

* Xử lý các tài liệu đầu vào và lưu trữ
* Xử lý ngôn ngữ tự nhiên
* Triển khai chức năng tìm kiếm thông tin

4. Kiểm thử

* Sử dụng các câu hỏi để xem câu trả lời của chatbot
* Cải thiện chatbot

5. Triển khai và viết tài liệu hướng dẫn sử dụng

# Chương 2. Phân tích hệ thống

## 2.1. Mô tả bài toán

- Dự án xây dựng một chatbot thông minh có khả năng tư vấn và giải đáp các câu hỏi liên quan đến thông tin và quy chế đào tạo của trường đại học. Chatbot sử dụng ngôn ngữ tự nhiên để tương tác với người dùng, giúp họ dễ dàng tra cứu các thông tin cần thiết mọi lúc, mọi nơi mà không cần đến sự hỗ trợ trực tiếp từ con người.

- Mục tiêu:

* Cung cấp nền tảng hỗ trợ tự động tra cứu thông tin đào tạo qua chatbot.
* Rút ngắn thời gian tiếp cận thông tin cho sinh viên và người quan tâm.
* Giảm tải khối lượng công việc cho bộ phận tư vấn đào tạo.
* Tăng tính tiện lợi, khả dụng và thân thiện trong việc tiếp cận thông tin học vụ.

## 2.2. Yêu cầu chức năng

- Hệ thống cho phép người dùng là **khách vãng lai,** **người dùng đăng ký tài khoản** tương tác thông qua các hoạt động nhắn tin với **chatbot**

- Các chức năng chính:

1. Đăng kí, đăng nhập, đăng xuất cho người dùng

2. Trao đổi với chatbot về các thông tin, quy chế đào tạo của nhà trường

3. Thêm đoạn chat mới

4. Xóa đoạn chat cũ

## 2.3. Use Case Diagram tổng quát

*A diagram of a person with text

AI-generated content may be incorrect.*

## 2.4. Mô tả các Use Case chính

### 2.4.1. Đăng kí

|  |  |
| --- | --- |
| ID | UC24.1 |
| Name | Đăng ký |
| Brief description | Chức năng cho phép khách vãng lai đăng ký tài khoản |
| Actors | Khách vãng lai |
| Pre-conditions | Người dùng chưa có tài khoản |
| Basic flow | 1. Người dùng vào trang đăng ký  2. Người dùng điền đầy đủ thông tin  3. Người dùng ấn “Đăng ký”  4. Hệ thống đăng ký tài khoản cho người dùng và chuyển hướng người dùng đến trang đăng nhập |
| Alternative flow | Thông tin đăng ký không đúng định dạng |
| Result | Người dùng đăng ký được tài khoản của mình |
| Exceptions | Lỗi kết nối hoặc lỗi hệ thống  Người dùng đăng ký tài khoản đã tồn tại trong hệ thống |

### 2.4.2. Đăng nhập

|  |  |
| --- | --- |
| ID | UC32 |
| Name | Đăng nhập |
| Brief description | Chức năng cho phép người dùng đã có tài khoản của hệ thống đăng nhập vào hệ thống |
| Actors | Người dùng |
| Pre-conditions | Người dùng đã có tài khoản |
| Basic flow | 1. Người dùng vào trang đăng nhập  2. Người dùng điền thông tin  3. Người dùng ấn “Đăng nhập”  4. Hệ thống chuyển hướng người dùng đến trang chủ |
| Alternative flow | Tài khoản hoặc mật khẩu của người dùng không đúng |
| Result | Người dùng đăng nhập được vào hệ thống. |
| Exceptions | Lỗi kết nối hoặc lỗi hệ thống  Người dùng chưa có tài khoản |

### 2.4.3. Đăng xuất

|  |  |
| --- | --- |
| ID | UC33 |
| Name | Đăng xuất |
| Brief description | Chức năng cho phép người dùng đăng xuất khỏi hệ thống |
| Actors | Người dùng |
| Pre-conditions | Người dùng đã đăng nhập vào hệ thống |
| Basic flow | 1. Người dùng bấm vào Avatar góc trái màn hình  2. Người dùng chọn đăng xuất  3. Hệ thống hỏi lại xem người dùng có muốn đăng xuất hay không  4. Người dùng đồng ý đăng xuất  5. Hệ thống đăng xuất tài khoản và chuyển hướng đến trang đăng nhập |
| Alternative flow | Người dùng không bấm đồng ý đăng xuất |
| Result | Người dùng đăng xuất thành công |
| Exceptions | Lỗi kết nối hoặc lỗi hệ thống |

### 2.4.4. Nhắn tin

|  |  |
| --- | --- |
| ID | UC34 |
| Name | Chat với bot |
| Brief description | Chức năng cho phép sử dụng chức năng chat với bot |
| Actors | Khách vãng lai, Người dùng |
| Pre-conditions | Người dùng hoặc khách vãng lai truy cập vào hệ thống |
| Basic flow | 1. Người dùng truy cập vào hệ thống  2. Hệ thống hiển thị thanh nhập dữ liệu  3. Người dùng nhập câu hỏi muốn bot trả lời  4. Bot sẽ trả về câu trả lời trên màn hình |
| Alternative flow |  |
| Result | Người dùng chat với bot và nhận được câu trả lời |
| Exceptions | Lỗi kết nối hoặc lỗi hệ thống |

### 2.4.5. Thêm cuộc trò chuyện

|  |  |
| --- | --- |
| ID | UC35 |
| Name | Thêm cuộc trò chuyện |
| Brief description | Chức năng cho phép người dùng thêm cuộc trò chuyện khác |
| Actors | Người dùng |
| Pre-conditions | Người dùng đăng nhập vào hệ thống |
| Basic flow | 1. Người dùng đăng nhập vào hệ thống  2. Người dùng bấm vào nút “Thêm cuộc trò chuyện”  2. Hệ thống hiển thị cuộc trò chuyện mới |
| Alternative flow | Khách vãng lai sẽ bị xóa cuộc trò chuyện hiện tại |
| Result | Người dùng thêm cuộc trò chuyện mới |
| Exceptions | Lỗi kết nối hoặc lỗi hệ thống |

### 2.4.6. Xóa cuộc trò chuyện

|  |  |
| --- | --- |
| ID | UC36 |
| Name | Xóa cuộc trò chuyện |
| Brief description | Chức năng cho phép người dùng xóa cuộc trò chuyện cũ |
| Actors | Người dùng |
| Pre-conditions | Người dùng đã đăng nhập vào hệ thống |
| Basic flow | 1. Người dùng nhấn vào lịch sử trò chuyện  2. Người dùng bấm vào nút “Xóa”  3. Hệ thống hiện thông báo người dùng có muốn xóa cuộc trò chuyện không  4. Người dùng đồng ý  5. Hệ thống xóa cuộc trò chuyện và thông báo thành công |
| Alternative flow | Người dùng bấm vào nút “Không” |
| Result | Người dùng đã xóa cuộc trò chuyện thành công |
| Exceptions | Lỗi kết nối hoặc lỗi hệ thống  Người dùng chưa đăng nhập vào hệ thống hoặc không có cuộc trò chuyện nào |

## 2.5. Biểu đồ hoạt động (*Các chức năng chính*)

### 2.5.1. Luồng xử lý tin nhắn người dùng

A diagram of a flowchart

AI-generated content may be incorrect.

### 2.5.2. Xem lịch sử chat

A diagram of a flowchart

AI-generated content may be incorrect.

### 2.5.3. Xử lý tài liệu đầu vào

A diagram of a flowchart

AI-generated content may be incorrect.

# Chương 3. Thiết kế hệ thống

## 3.1. Lựa chọn mô hình kiến trúc (Monolithic, Microservices, MVC, Layered, Clean Architecture, ...)

- Dự án này sử dụng Layered Architecture (Kiến trúc phân lớp) kết hợp với Repository Pattern và Dependency Injection.

- Các lớp kiến trúc chính:

1. Presentation Layer (Lớp trình bày)

* Templates HTML với Flask
* Static CSS files

2. Service Layer (Lớp dịch vụ)

* ChatbotService
* Quản lý logic nghiệp vụ và tích hợp RAG chain

3. Repository Layer (Lớp truy cập dữ liệu)

* Interface pattern với IUserRepository và IChatRepository
* Implementation với SQLAlchemy

4. Data Layer (Lớp dữ liệu)

* Models với SQLAlchemy ORM
* Message model

5. Dependency Injection Pattern

* Hệ thống sử dụng inject library để quản lý dependencies
* Cấu hình binding các service

6. Singleton Pattern:

* Được áp dụng trong ChatBotService để đảm bảo chỉ có một instance RAG Chain
* VectorStoreManager để quản lý Chroma client

7. Strategy Pattern:

* Được sử dụng trong document processing để chọn chiến lược chia text phù hợp với từng loại file

## 3.2. Sơ đồ kiến trúc tổng thể

A screenshot of a computer

AI-generated content may be incorrect.

## 3.3. Biểu đồ tuần tự (*các chức năng chính*)

### 3.3.1. Xử lý tin nhắn người dùng

A screenshot of a computer screen

AI-generated content may be incorrect.

### 3.3.2. Xem lịch sử đoạn chat

A diagram with text and words

AI-generated content may be incorrect.

### 3.3.3. Xử lý tài liệu

A diagram of a project

AI-generated content may be incorrect.

## 3.4. Thiết kế lớp (Class Diagram)

### 3.4.1. User Diagram

A computer screen shot of a computer program

AI-generated content may be incorrect.

### 3.4.2. Chat Diagram

A diagram of a company

AI-generated content may be incorrect.

## 3.5. Thiết kế cơ sở dữ liệu (ERD, bảng, mối quan hệ)

### 3.5.1. Bảng users

|  |  |  |  |
| --- | --- | --- | --- |
| **Trường** | **Kiểu** | **Ràng buộc** | **Mô tả** |
| Id | Interger(11) | Primary key | Mã định danh người dùng |
| Username | Varchar(20) | Unique, not null | Tên đăng nhập của người dùng |
| Email | Varchar(100) | Unique, not null | Địa chỉ email của người dùng |
| Password\_hash | Varchar(255) | Not null | Mật khẩu mã hóa |

### 3.5.2. Bảng chat\_sessions

|  |  |  |  |
| --- | --- | --- | --- |
| **Trường** | **Kiểu** | **Ràng buộc** | **Mô tả** |
| Id | Interger(11) | Primary key | Mã định danh session |
| User\_id | Interger(11) | Khóa ngoại bảng **users**, not null | Người tạo session |
| Start\_time | Datetime | Default: datetime.now | Thời gian tạo session |

### 3.5.3. Messages

|  |  |  |  |
| --- | --- | --- | --- |
| **Trường** | **Kiểu** | **Ràng buộc** | **Mô tả** |
| Id | Interger(11) | Primary key | Mã định danh tin nhắn |
| Session\_id | Interger(11) | Khóa ngoại bảng chat\_**sessions**, not null | Tin nhắn thuộc session nào |
| Sender\_type | Varchar(10) | Not null | Tin nhắn của ‘user’ hay ‘bot’ |
| Content | Text | Not null | Nội dung tin nhắn |
| Timestamp | Datetime | Default:datetime.now | Thời gian tạo tin nhắn |

### 3.5.4. Sơ đồ ERD

A close-up of a card

AI-generated content may be incorrect.

## 3.6. Thiết kế giao diện người dùng (Mockup / Wireframe)

# Chương 4. Cài đặt và triển khai hệ thống

## 4.1. Môi trường triển khai

### 4.1.1. Môi trường Development/Local

1. Web Framework & Server

- Dự án sử dụng Flask framework với development server.

2. Database Configuration

- Hệ thống sử dụng MySQL database.

- Database connection được thiết lập thông qua SQLAlchemy.

3. Session Management

- Hệ thống sử dụng filesystem-based session storage.

4.AI/ML Components

**- Vector Database**: ChromaDB

**- OCR Engine**: Tesseract

**- AI Model**: Gemini 2.0 Flash thông qua API

### 4.1.2. Cấu hình Environment Variables

1. Dự án yêu cầu các biến môi trường sau:

**- SECRET\_KEY**: Cho Flask session security

**- DATABASE\_URL**: Cho database connection

**- GEMINI\_API\_KEY**: Cho Gemini AI API

**- DATA\_DIRECTORY**: Cho data storage path

**- FLASK\_ENV**: Để xác định môi trường (development/production)

2. Logging & Debugging

- Hệ thống có logging configuration và debug mode được bật dựa trên environment variable.

3. Dependency Management

- Dự án sử dụng dependency injection pattern

4. Template & Static Files

- Giao diện web được phục vụ thông qua Flask templates với static CSS files.

## 4.2. Cài đặt hệ thống

### 4.2.1. Yêu cầu hệ thống

**1. Phần mềm cần thiết**

* Python 3.8+
* MySQL Server
* Tesseract OCR
* Git

**2. API Keys**

* Google Gemini API Key (cho AI model)

### 4.2.2. Hướng dẫn cài đặt từng bước

**Bước 1: Clone repository**

1. *git clone https://github.com/quyethv203/HUCE\_Chatbot.git*
2. *cd HUCE\_Chatbot*

**Bước 2: Cài đặt dependencies**

*pip install -r requirements.txt*

**Bước 3: Cài đặt Tesseract OCR**

**Windows**: Tải và cài đặt từ [GitHub Tesseract](https://github.com/UB-Mannheim/tesseract/wiki)

**Bước 4: Thiết lập MySQL Database**

1. Tạo database mới:
2. **CREATE** DATABASE zodiac\_chatbot;
3. Cấu hình connection string

**Bước 5: Cấu hình Environment Variables**

1. Tạo file **.env** hoặc thiết lập biến môi trường:
2. export GEMINI\_API\_KEY="your\_gemini\_api\_key\_here"
3. export DATABASE\_URL="mysql+pymysql://username:password@localhost:3306/zodiac\_chatbot"
4. export SECRET\_KEY="your\_secret\_key\_here"
5. export FLASK\_ENV="development"
6. export GEMINI\_MODEL="your\_gemini\_model\_here"
7. export SENTENCE\_TRANSFORMER\_MODEL\_NAME="your\_gemini\_model\_here"
8. export CHUNK\_SIZE
9. export CHUNK\_OVERLAP
10. export DATA\_DIRECTORY: đường dẫn đến file data
11. export VECTOR\_DB\_PATH: đường dẫn lưu vector database
12. export TESSERACT\_PATH: đường dẫn cài đặt tesseract

**Bước 6: Khởi tạo Database**

1. Chạy ứng dụng lần đầu để tạo các bảng
2. Database sẽ được khởi tạo tự động

**Bước 7: Chuẩn bị dữ liệu Vector Database**

1. Tạo thư mục data:

***mkdir*** *-p* ***data****/documents*

***mkdir*** *-p* ***data****/chroma\_db*

1. Đặt các file tài liệu (PDF, DOCX, TXT) vào **data /**
2. Chạy document processor để tạo vector embeddings:

*python app/rag/document\_processor.py*

1. Quá trình này được thực hiện trong document\_processor.py

**Bước 8: Test RAG Chain**

1. Kiểm tra RAG chain hoạt động:

***python*** *test\_rag\_chain.****py***

1. Script test sẽ kiểm tra API key và RAG chain như trong test\_rag\_chain.py

**Bước 9: Chạy ứng dụng**

***python*** *run.****py***

Ứng dụng sẽ chạy với cấu hình debug được thiết lập trong run.py

# Chương 5. Kết quả và đánh giá

## 5.1. Kết quả thử nghiệm hệ thống

* + RAG chain hoạt động ổn định với cả Basic và Advanced mode.
  + Chatbot trả lời đúng trọng tâm và không tự nghĩ ra câu trả lời.
  + Tích hợp đầy đủ với Gemini Al, Sentence transformer model, ChromaDB.
  + Hỗ trợ nhiều định dạng file khác nhau.
  + Xử lý các văn bản có cấu trúc và văn bản thường

## 5.2. Đánh giá hiệu quả hệ thống

1. Ưu điểm:

* + Các chức năng hoạt động ổn định
  + Giao diện thân thiện với người dùng
  + Áp dụng được các mẫu thiết kế giúp dễ dàng mở rộng, sửa đổi

2. Nhược điểm:

* + Vector Database cần triển khai lên cloud thay vì local
  + Nguồn tài liệu vẫn chưa phong phú
  + Phụ thuộc vào API key GEMINI

# Tài liệu tham khảo

1. Tài liệu kĩ thuật: Flask Framework Document, LangChain Document, Google AI Gemini API, ChromaDB Document, Refactor.guru,..

2. Các diễn đàn Công nghệ: Viblo, Facebook, Github,..

3. Công cụ hỗ trợ: Gemini AI, ChatGPT

4. Tài liệu học tập Đại học Xây Dựng Hà Nội.