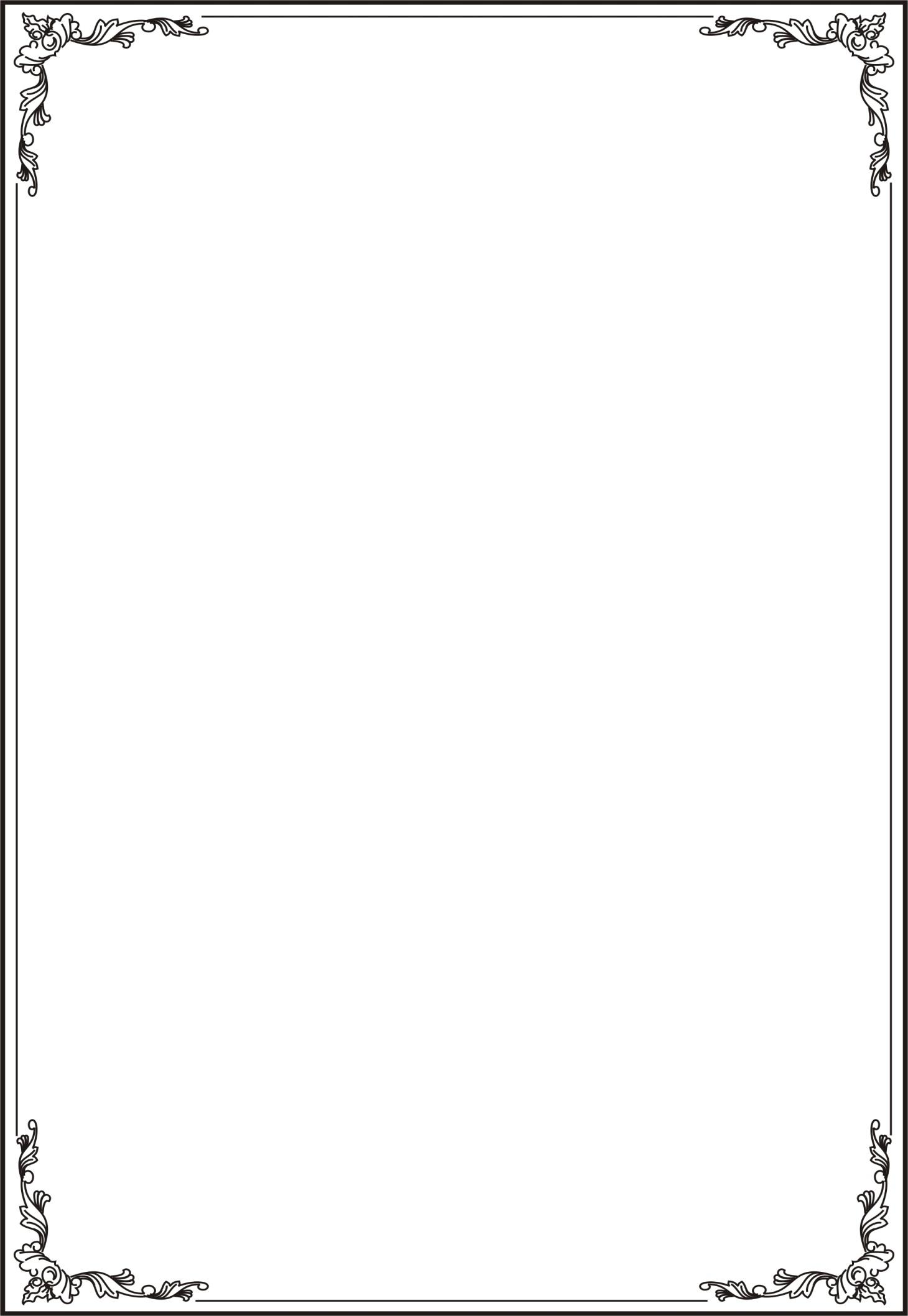
**TRƯỜNG ĐẠI HỌC THĂNG LONG**

**KHOA TOÁN TIN**

**-----🙞🙜🕮🙞🙜-----**



**KHOÁ LUẬN TỐT NGHIỆP**

**ĐỀ TÀI: TẠO TRỢ LÝ ẢO VỚI RASA**

**GIÁO VIÊN HƯỚNG DẪN** **NHÓM SINH VIÊN THỰC HIỆN**

A37134 – Lê Quang Huy

**Hà Nội – 2023**

**TRƯỜNG ĐẠI HỌC THĂNG LONG**

**KHOA TOÁN TIN**

**-----🙞🙜🕮🙞🙜-----**



**KHOÁ LUẬN TỐT NGHIỆP**

**ĐỀ TÀI: TẠO TRỢ LÝ ẢO VỚI RASA**

**GIÁO VIÊN HƯỚNG DẪN** **NHÓM SINH VIÊN THỰC HIỆN**

A37134 – Lê Quang Huy

**Hà Nội – 2023**

**LỜI CẢM ƠN**

Lời đầu tiên chúng em xin được phép gửi lời cảm ơn chân thành nhất tới toàn bộ các thầy cô giáo trong khoa Toán - Tin cũng như các thầy cô giảng dạy trong trường Đại học Thăng Long đã truyền đạt những kiến thức quý báu và bổ ích nhất cho chúng em trong những năm học vừa qua.

Đặc biệt, chúng em xin chân thành cảm ơn cô – ABC và ABC, giảng viên khoa Toán - Tin, trường Đại học Thăng Long đã tận tình hướng dẫn, động viên và trực tiếp giúp đỡ chúng em trong suốt quá trình triển khai và thực hiện phần mềm.

Để có được kết quả như ngày hôm nay, chúng em rất biết ơn thầy cô gia đình đã động viên, khích lệ, tạo mọi điều kiện thuận lợi nhất trong suốt quá trình học tập cũng như quá trình thực hiện phần mềm này.

Bên cạnh đó, chúng em xin chân thành cám ơn các bạn trong khoa Toán – Tin trường Đại học Thăng Long đã ủng hộ, giúp đỡ, chia sẻ kiến thức, kinh nghiệm và các tài liệu có được cho chúng em trong suốt học tại trường.

Vì kiến thức còn hạn chế, trong quá trình triển khai và thực hiện phần mềm này chúng em không tránh khỏi có những sai sót không mong muốn, chính vì vậy chúng em rất mong nhận được những ý kiến đóng góp từ các thầy cô cũng như các bạn trong khoa để hoàn thiện phần mềm này cũng như bổ sung những kiến thức còn thiếu sót.

Cuối cùng chúng em xin kính gửi lời chúc tới toàn thể các thầy cô trong khoa Toán -Tin cũng như các thầy cô giảng dạy trong trường Đại học Thăng Long luôn luôn mạnh khỏe và tràn đầy nhiệt huyết để truyền đạt kiến thức cho các thế hệ mai sau.

Em xin chân thành cảm ơn!

|  |  |
| --- | --- |
|  | Hà Nội, ngày 28 tháng 5 năm 2023  *Sinh viên thực hiện*  LÊ QUANG HUY |

**MỤC LỤC**

[LỜI GIỚI THIỆU 1](#_Toc138257716)

[CHƯƠNG 1. Tổng quan phần mềm 3](#_Toc138257717)

[1.1. Mô tả bài toán 3](#_Toc138257718)

[1.2. Yêu cầu nghiệp vụ 6](#_Toc138257719)

[1.3. Sơ đồ usecase các chức năng chính 6](#_Toc138257720)

[1.3.1. Sơ đồ Usecase 6](#_Toc138257721)

[1.3.2. Các tác nhân có trong hệ thống 6](#_Toc138257722)

[CHƯƠNG 2. Kiến trúc tổng thể của hệ thống 8](#_Toc138257723)

[2.1. Sơ đồ kiến trúc tổng thể 8](#_Toc138257724)

[2.1.1. Sơ đồ 8](#_Toc138257725)

[2.1.2. Mô tả sơ đồ 8](#_Toc138257726)

[2.2. Công nghệ sử dụng 9](#_Toc138257727)

[2.2.1. Giới thiệu về SQL 9](#_Toc138257728)

[2.2.2. Giới thiệu về Microsoft SQL Server 10](#_Toc138257729)

[2.2.3. Giới thiệu về Ngrok 10](#_Toc138257730)

[2.2.4. Rasa 11](#_Toc138257731)

[2.2.5. Facebook Developer 14](#_Toc138257732)

[2.3. Đặc tả nhóm chức năng 15](#_Toc138257733)

[2.3.1. UC1 Yêu cầu thông tin môn học 15](#_Toc138257734)

[2.3.2. UC2 Yêu cầu thông tin môn học có thể đăng ký với điều kiện cung cấp 16](#_Toc138257735)

[2.3.3. UC3 Yêu cầu thông tin lộ trình học gợi ý cho sinh viên 17](#_Toc138257736)

[2.3.4. UC4 Yêu cầu giải đáp một số vấn đề liên quan ĐKH hoặc CTĐT 18](#_Toc138257737)

[CHƯƠNG 3. Thiết kế cơ sở dữ liệu và tạo dữ liệu huấn luyện 19](#_Toc138257738)

[3.1. Thiết kế Cơ sở dữ liệu 19](#_Toc138257739)

[3.1.1. Sơ đồ Diagram của Database 19](#_Toc138257740)

[3.1.2. Danh sách các bảng 19](#_Toc138257741)

[3.1.3. Chi tiết các bảng 19](#_Toc138257742)

[3.2. Tạo dữ liệu huấn luyện 20](#_Toc138257743)

[3.2.1. Dữ liệu cho intent: Yêu cầu thông tin môn học 21](#_Toc138257744)

[3.2.2. Dữ liệu cho intent: Yêu cầu thông tin môn học với điều kiện tiên quyết 22](#_Toc138257745)

[3.2.3. Dữ liệu cho intent: Yêu cầu lộ trình học 22](#_Toc138257746)

[3.2.4. Dữ liệu cho intent: Yêu cầu giải đáp một số thắc mắc CTĐT hoặc ĐKH 23](#_Toc138257747)

[CHƯƠNG 4. Thiết kế Actions 24](#_Toc138257748)

[4.1. Giới thiệu chung 24](#_Toc138257749)

[4.1.1. Custom Actions 24](#_Toc138257750)

[4.1.2. Thiết kế chức năng 24](#_Toc138257751)

[4.2. Cấu trúc thư mục 26](#_Toc138257752)

[4.3. Thuật toán trong Custom Actions 27](#_Toc138257753)

[4.3.1. Mục tiêu 27](#_Toc138257754)

[4.3.2. Giải pháp 27](#_Toc138257755)

[CHƯƠNG 5. Machine learning 30](#_Toc138257756)

[5.1. Giới thiệu chung 30](#_Toc138257757)

[5.2. Mô hình Neural Network 31](#_Toc138257758)

[5.2.1. Giới thiệu mô hình 31](#_Toc138257759)

[5.2.2. Mô hình Recurrent Neural Network 33](#_Toc138257760)

[5.2.3. Các thành phần khác trong Pipelines và Policies 37](#_Toc138257761)

[CHƯƠNG 6. Cài đặt và thực nghiệm 41](#_Toc138257762)

[6.1. Cài đặt 41](#_Toc138257763)

[6.2. Thực nghiệm 44](#_Toc138257764)

[6.2.1. Yêu cầu thông tin môn học 44](#_Toc138257765)

[6.2.2. Yêu cầu lộ trình học 46](#_Toc138257766)

[6.2.3. Hỏi giải đáp một số vấn đề 46](#_Toc138257767)

[KẾT LUẬN 47](#_Toc138257768)

[TÀI LIỆU THAM KHẢO 49](#_Toc138257769)

**DANH SÁCH HÌNH ẢNH**

[Hình 1.1 Khung CTĐT ngành Công nghệ thông tin khóa 32[1] 5](#_Toc138257684)

[Hình 1.2 Sơ đồ Use Case với tác nhân User (Sinh viên, người dùng) 6](#_Toc138257685)

[Hình 2.1 Sơ đồ kiến trúc tổng thể 8](#_Toc138257686)

[Hình 2.2 SQL 9](#_Toc138257687)

[Hình 2.3 Microsoft SQL Server 10](#_Toc138257688)

[Hình 2.4 Ngrok 10](#_Toc138257689)

[Hình 2.5 Rasa 11](#_Toc138257690)

[Hình 2.6 Facebook Develepers 14](#_Toc138257691)

[Hình 3.1 Diagram 19](#_Toc138257692)

[Hình 4.1 Sơ đồ tuần tự chức năng giải đáp yêu cầu của User 25](#_Toc138257693)

[Hình 4.2 Cấu trúc thư mục Rasa 26](#_Toc138257694)

[Hình 4.3 Sơ đồ khối thuật toán nhận diện yêu cầu và trả về phản hồi 29](#_Toc138257695)

[Hình 5.1 Kiến trúc Neural Network 32](#_Toc138257696)

[Hình 5.2 Mô hình RNN Many to many 33](#_Toc138257697)

[Hình 5.3 Mô hình RNN 34](#_Toc138257698)

[Hình 5.4 Công thức của hàm Softmax 36](#_Toc138257699)

[Hình 5.5 Đồ thị của hàm Softmax 36](#_Toc138257700)

[Hình 5.6 Pipelines phần 1 37](#_Toc138257701)

[Hình 5.7 Pipelines phần 2 38](#_Toc138257702)

[Hình 5.8 Policies 39](#_Toc138257703)

[Hình 6.1 Giao diện mã truy cập 41](#_Toc138257704)

[Hình 6.2 Giao diện Webhooks 42](#_Toc138257705)

[Hình 6.3 Khởi tạo ngrok 42](#_Toc138257706)

[Hình 6.4 Giao diện chỉnh sửa URL gọi lại 43](#_Toc138257707)

[Hình 6.5 Giao diện cài đặt thông tin cơ bản 43](#_Toc138257708)

[Hình 6.6 File credentials.yml 44](#_Toc138257709)

[Hình 6.7 Liệt kê các môn học có chung tên 44](#_Toc138257710)

[Hình 6.8 Liệt kê các môn thuộc một học phần 45](#_Toc138257711)

[Hình 6.9 Chi tiết một môn học 45](#_Toc138257712)

[Hình 6.10 Yêu cầu các môn có thể học 45](#_Toc138257713)

[Hình 6.11 Lộ trình học của năm 46](#_Toc138257714)

[Hình 6.12 Lộ trình học của kỳ và năm 46](#_Toc138257715)

**DANH MỤC TỪ VIẾT TẮT**

|  |  |  |
| --- | --- | --- |
| **STT** | **Từ viết tắt** | **Ý nghĩa** |
| **1** | CTĐT | Chương trình đào tạo |
| **2** | KKT | Khối kiến thức |
| **3** | UC | Use-case |
| **4** | ĐKH | Đăng ký học |
| **5** | CSDL | Cơ sở dữ liệu |
| **6** | ĐKTQ | Điều kiện tiên quyết |
| **7** | CVHT | Cố vấn học tập |
| **8** | RNN | Recurrent Neural Network |
| **9** | NN | Neural Network |

# LỜI GIỚI THIỆU

Ngày nay với sự phát triển mạnh mẽ của xã hội và khoa học công nghệ, công nghệ thông tin là một trong những ngành lớn mạnh hàng đầu. Các ứng dụng tin học được áp dụng trong hầu hết các lĩnh vực của đời sống. Nó trở thành công cụ hữu ích cho con người trong các hoạt động lưu trữ, xử lý thông tin một cách nhanh chóng và đem lại hiệu quả cao.

Hằng năm, mỗi tân sinh viên của trường Đại học Thăng Long đều được nhận một cuốn hướng dẫn học tập với nội dung chính là chương trình đào tạo các ngành học giúp sinh viên có cái nhìn tổng quát nhất về quá trình học tập trong tương lai của bản thân. Nhưng kể cả khi đã có trong tay hướng dẫn thì cũng vẫn sẽ gặp ít nhiều trong việc định hướng và lên kế hoạch đăng ký học cho tương lai. Là một sinh viên em cũng đã từng gặp những bất cập như vậy những lúc như này nếu có thể liên lạc với cố vấn học tập thì có thể dễ dàng giải đáp các thắc mắc nhưng do số lượng sinh viên quá lớn dẫn đến việc cố vấn học tập không thể trả lời toàn bộ sinh viên được. Hiểu được vấn đề này cùng với sự phát triển của khoa học công nghệ, việc tin học hóa trong quy trình quản lý tư vấn của cố vấn học tập hướng dẫn và giải đáp chương trình đào tạo đối với nhà trường và sinh viên là thực sự cần thiết. Việc đưa những thông tin này và thiết kế dạng trợ lý ảo sẽ khiến việc tiếp cận trở nên dễ dàng hơn.

Báo cáo này tổng kết lại quy trình phát triển phần mềm, nội dung báo cáo bao gồm:

**Chương 1: Tổng quan phần mềm**

Giới thiệu bài toán hệ thống cần giải quyết và phân tích các yêu cầu nghiệp vụ để hiểu rõ những chức năng cơ bản và cần thiết của một cố vấn học tập và xây dựng cơ sở dữ liệu, thiết kế dữ liệu huấn luyện và thiết kế mô hình Học máy phù hợp.

**Chương 2: Kiến trúc hệ thống**

Ở chương này, sẽ đưa ra kiến trúc tổng thể của hệ thống kèm theo mô tả cho từng thành phần kiến trúc và các công nghệ được áp dụng phù hợp cho từng thành phần này.

Đặc tả các chức năng dựa theo từng nhóm chức năng của phần mềm như: các tác nhân sử dụng chức năng, luồng hoạt động của chức năng đó và có kèm theo giao diện mô tả cho mỗi luồng xử lý.

**Chương 3: Thiết kế cơ sở dữ liệu và tạo dữ liệu huấn luyện**

Ở chương này sẽ đưa ra lược đồ cơ sở dữ liệu của toàn bộ hệ thống cũng như xây dựng dữ liệu để huấn luyện mô hình.

**Chương 4: Thiết kế Actions**

Chúng ta sẽ tìm hiểu và làm rõ hơn một trong những yếu tố quan trọng nhất trong Rasa và thức tinh chỉnh sao cho phù hợp với từng trường hợp của người dùng yêu cầu.

**Chương 5: Machine Learning**

Ở chương này chúng ta sẽ đi vào Học máy đây là một phần quan trọng và cốt lõi của khoá luận này. Việc áp dụng Học máy vào sẽ khiến cho trợ lý ảo chạy với đầu ra tốt hơn cải thiện độ chính xác hơn việc sử dụng mô hình ban đầu của Rasa.

**Chương 6: Cài đặt và thực nghiệm**

Chỉ ra cách thiết kế và môi trường cài đặt hệ thống, hệ thống file mã nguồn. Thực hiện thử nghiệm để đánh giá độ hoàn thiện.

# Tổng quan phần mềm

## Mô tả bài toán

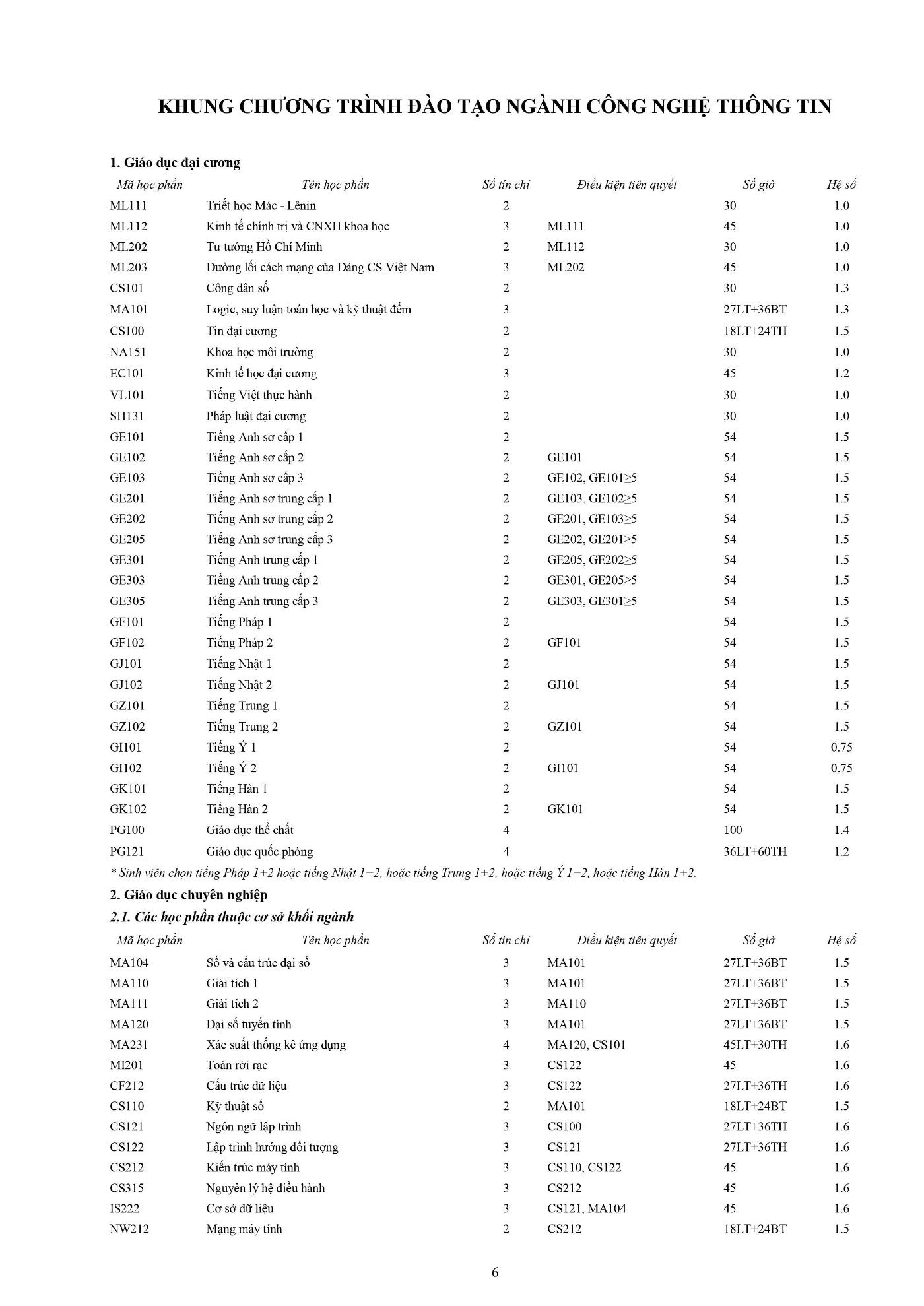
Hằng năm, mỗi sinh viên mới nhập học đều được phát một cuốn hướng dẫn học tập, trong cuốn hướng dẫn này có nội dung quan trọng là chương trình đào tạo các ngành. Chương trình đào tạo (CTĐT) mỗi ngành bao gồm danh sách các môn học được phân chia vào các học phần khác nhau như: Giáo dục đại cương, Giáo dục chuyên nghiệp: Học phần thuộc cơ sở khối ngành, Học phần bắt buộc, Học phần lựa chọn và Thực tập, Khoá luận tốt nghiệp, Chuyên đề tốt nghiệp.

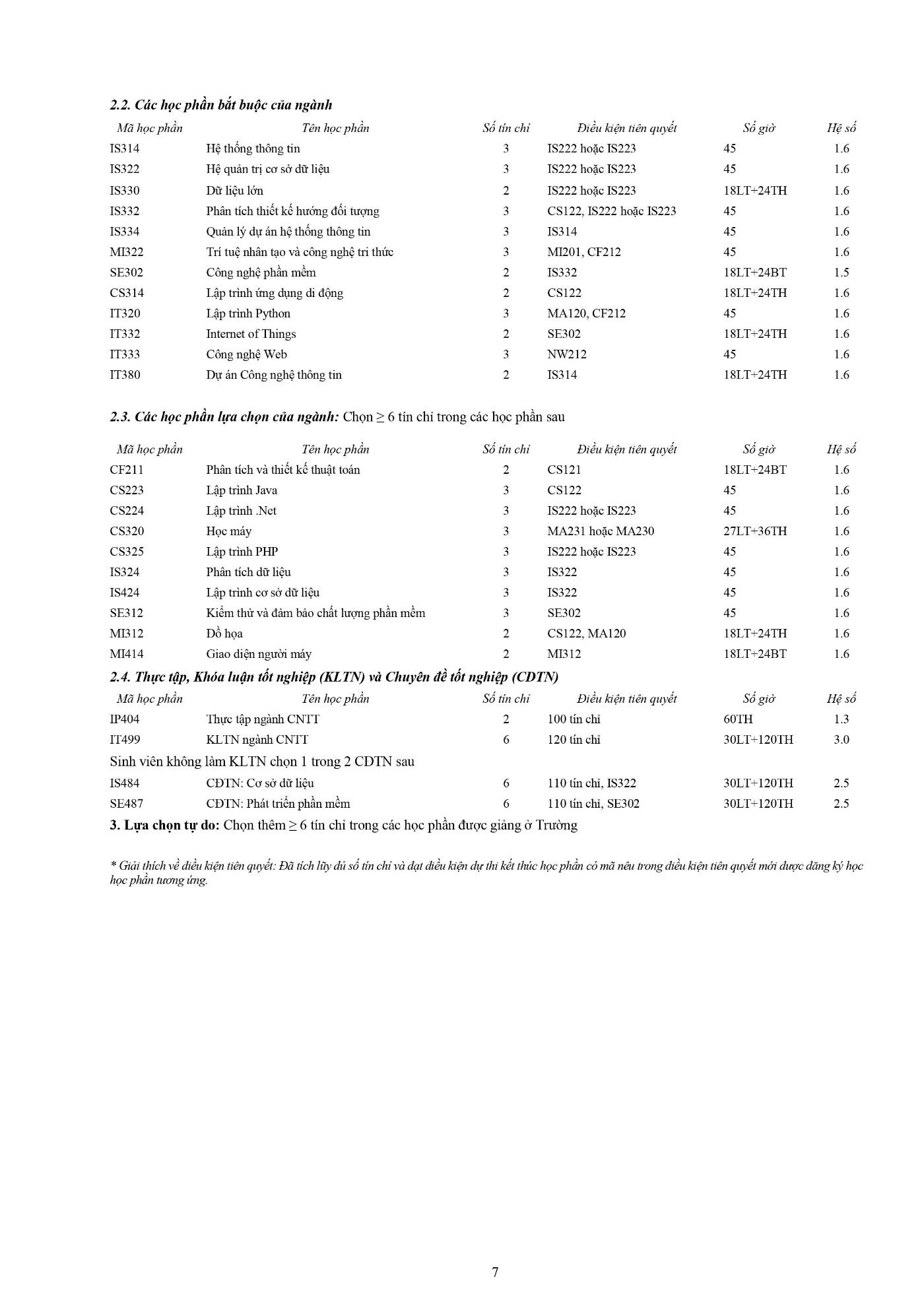
Việc có nhiều học phần trong mỗi chương trình đào tạo của từng chuyên ngành học dẫn đến việc sinh viên mới nhập học dễ bị rối và khó có thể hình dung và lên được kế hoạch học tập trong tương lai. Cũng chưa hiểu rõ một số vấn đề liên quan đến đăng ký học dẫn đến sinh viên sẽ phải liên tục liên lạc với cố vấn học tập. Việc này khiến cho cố vấn học tập không thể trả lời hết các yêu cầu và cũng như không giải quyết nhanh vấn đề của tất cả sinh viên.

Trợ lý ảo sẽ có chức năng thay thế cho một cố vấn học tập, dựa vào các học phần đã có và chương trình học của khối ngành từ các giảng viên. Trợ lý ảo hoàn toàn có thể trả lời các câu hỏi liên quan đến vấn đề lộ trình học, tra cứu thông tin môn học, giải đáp thắc mắc sinh viên liên quan đến việc ĐKH.

Sinh viên hay người sử dụng hệ thống khi muốn tư vấn sẽ cần đưa ra một số thông tin cần thiết như cần lộ trình học của năm học nào hoặc năm học và kỳ học, tên môn học cần tra cứu, hay đã qua các môn học điều kiện tiên quyết nào và có thể học môn gì tiếp theo. Và trợ lý ảo sẽ nhận từng yêu cầu từ phía sinh viên hay người sử dụng và phân loại yêu cầu (intent) để từ đó đưa ra các phản hồi (actions) tương ứng.

Sau đây là chương trình học cụ thể của khối ngành Công nghệ thông tin:





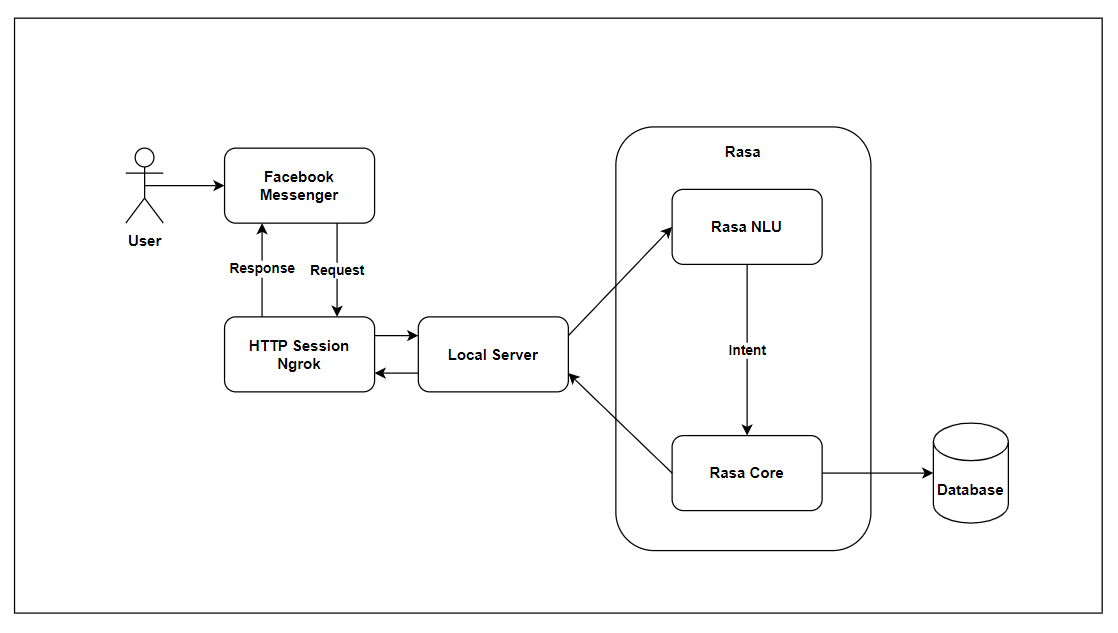
Hình . Khung CTĐT ngành Công nghệ thông tin khóa 32[1]

## Yêu cầu nghiệp vụ

* Phân loại được yêu cầu (intent) của người dùng.
* Dựa vào yêu cầu (intent) đã phân loại được phản hồi (actions) lại phía người dùng kết quả ứng với yêu cầu (intent).

## Sơ đồ usecase các chức năng chính

### Sơ đồ Usecase



Hình . Sơ đồ Use Case với tác nhân User (Sinh viên, người dùng)

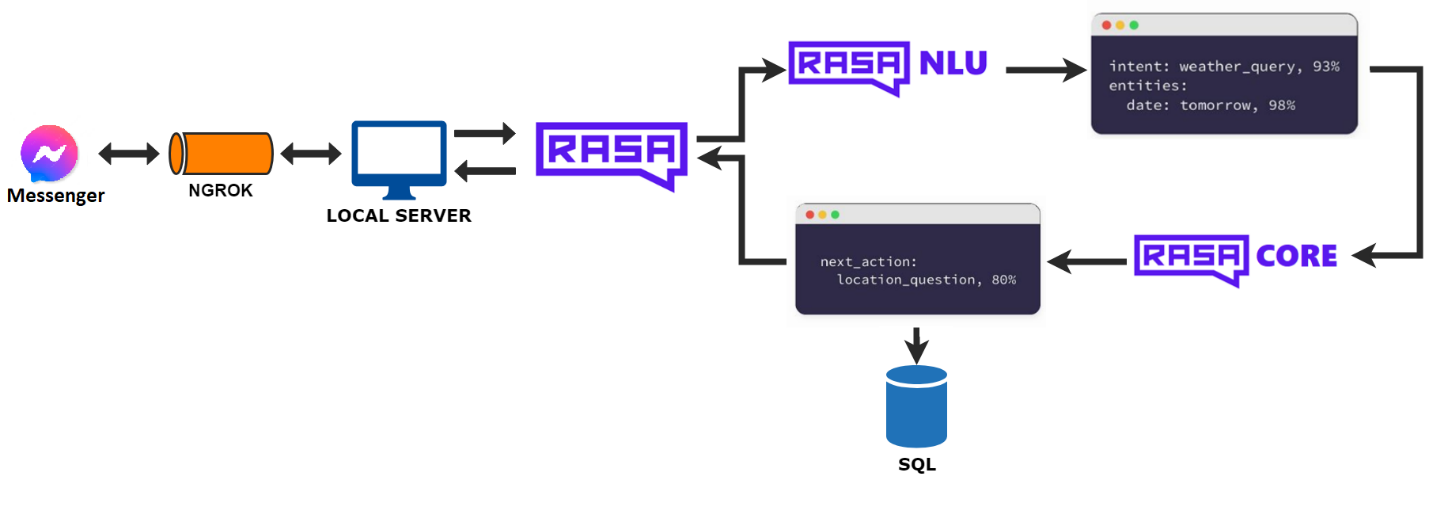
### Các tác nhân có trong hệ thống

| **Tác nhân** | **Vai trò** |
| --- | --- |
| User (Sinh viên) | Tác nhân có vai trò sử dụng các chức năng chính của hệ thống:  - Truy cập vào Facebook Messenger  - Gửi yêu cầu (intent) cho trợ lý ảo |
| Trợ lý ảo (ChatBot) | Tác nhân có vai trò phản hồi người dùng:  - Truy cập CSDL và trích xuất dữ liệu  - Phản hồi (actions) người dùng tương ứng với yêu cầu (intent) |

# Kiến trúc tổng thể của hệ thống

## Sơ đồ kiến trúc tổng thể

### Sơ đồ



Hình . Sơ đồ kiến trúc tổng thể

### Mô tả sơ đồ

Sinh viên (người dùng) truy cập Messenger của Facebook thực hiện thao tác gửi tin nhắn, thông qua Ngrok gửi một request đến server tại đây server nhận request và chuyền tin đến Rasa server thông qua Rasa NLU nhận diện được yêu cầu (intent) của người dùng hay từ bây giờ sẽ gọi là ý định khi xác định được ý định (intent) thông qua Rasa Core sẽ thực hiện thao tác lựa chọn hành động (Actions Selection) phản hồi về phía sinh viên thông qua Messenger.

## Công nghệ sử dụng



Hình . SQL

### Giới thiệu về SQL

SQL (Structured Query Language)[2] là một ngôn ngữ lập trình dùng để quản lý và tương tác với các hệ quản trị cơ sở dữ liệu (Database Management System - DBMS). SQL được sử dụng để truy vấn, thao tác, và quản lý dữ liệu trong cơ sở dữ liệu quan hệ (relational database).

SQL cung cấp các câu lệnh để thao tác với cơ sở dữ liệu như tạo, sửa đổi hoặc xóa bảng dữ liệu, thêm, cập nhật hoặc xóa dữ liệu trong bảng, truy vấn dữ liệu từ bảng, tạo và quản lý các ràng buộc dữ liệu, và thực hiện các thao tác liên quan đến việc truy xuất và xử lý dữ liệu.

SQL là một ngôn ngữ chuẩn được sử dụng rộng rãi trong các hệ quản trị cơ sở dữ liệu như MySQL, PostgreSQL, Oracle, SQL Server, SQLite và nhiều hệ quản trị cơ sở dữ liệu khác..

### Giới thiệu về Microsoft SQL Server



Hình . Microsoft SQL Server

Microsoft SQL Server[3] là một hệ quản trị cơ sở dữ liệu quan hệ (Relational Database Management System - RDBMS) do Microsoft phát triển và cung cấp. Nó là một trong những hệ quản trị cơ sở dữ liệu phổ biến và mạnh mẽ được sử dụng trong nhiều ứng dụng và môi trường kinh doanh.

### Giới thiệu về Ngrok



Hình . Ngrok

Ngrok[4] là một công cụ proxy công cộng và phần mềm túnel mã nguồn mở. Nó cho phép bạn tạo ra một kết nối an toàn giữa máy cục bộ của bạn và mạng Internet thông qua một kết nối ngược qua tường lửa (reverse tunneling). Điều này cho phép bạn truy cập vào các dịch vụ đang chạy trên máy cục bộ của bạn từ bất kỳ đâu trên Internet mà không cần cấu hình phức tạp hoặc thiết lập các quy tắc tường lửa.

### Rasa



Hình . Rasa

Rasa[5] là một nền tảng mã nguồn mở để xây dựng các ứng dụng trò chuyện thông minh. Nó cung cấp một bộ công cụ và thư viện cho việc phát triển các hệ thống trò chuyện AI, bao gồm cả chatbot và trợ lý ảo.

Cấu trúc files của rasa bao gồm:

* actions/\_\_init\_\_.py: Một file rỗng giúp python định vị hành động của bạn.
* actions/actions.py: Tệp này được sử dụng để tạo các hành động tùy chỉnh. Trong trường hợp bạn muốn gọi một máy chủ bên ngoài hoặc tìm nạp dữ liệu API bên ngoài, bạn có thể xác định hành động của mình tại đây.
* config.yml: File này định nghĩa cấu hình của NLU và core model. Nếu bạn đang sử dụng bất kỳ mô hình nào bên ngoài mô hình NLU, bạn phải xác định đường dẫn tại đây.
* Pipeline: một chuỗi các thành phần xử lý dữ liệu và tích hợp trong quá trình xử lý thông điệp người dùng. Mỗi thành phần trong pipeline đóng vai trò trong việc biến đổi và chuẩn bị dữ liệu cho các bước tiếp theo trong quá trình xử lý trò chuyện
* Policies: một chuỗi các thành phần xử lý dữ liệu và tích hợp trong quá trình xử lý thông điệp người dùng. Mỗi thành phần trong pipeline đóng vai trò trong việc biến đổi và chuẩn bị dữ liệu cho các bước tiếp theo trong quá trình xử lý trò chuyện
* credentials.yml: Tệp này được sử dụng để lưu trữ thông tin đăng nhập để kết nối với các dịch vụ bên ngoài như Facebook Messenger, Slack, v.v.
* data/nlu.md: Trong tệp này, chúng tôi xác định ý định của mình (người dùng có thể yêu cầu bot làm gì? ). Những ý định này sau đó được sử dụng trong đào tạo mô hình NLU.
* data/stories.md: Stories là cuộc trò chuyện mẫu giữa người dùng và bot dưới dạng ý định, phản hồi và hành động. Các câu chuyện Rasa là một dạng dữ liệu đào tạo được sử dụng để đào tạo các mô hình quản lý đối thoại của Rasa.
* domain.yml: Tệp này liệt kê các ý định khác nhau (những điều bạn mong đợi từ người dùng) với các phản hồi và hành động của bot mà nó có thể thực hiện.
* intents: khai báo các chủ ý của người dùng
* actions: khai báo các hành động thường quy ước là utter\_nameaction
* responses: định nghĩa các action sẽ thực hiện nội dung được quy định
* endpoints.yml: Điều này xác định chi tiết để kết nối các kênh như Slack, FB messenger, v.v. để lưu trữ dữ liệu trò chuyện trong cơ sở dữ liệu trực tuyến như Redis, v.v.
* models/<name>.tar.gz: mô hình ban đầu, tất cả các mô hình được đào tạo được lưu trữ trong thư mục mô hình. Để đào tạo lại mô hình, chúng tôi sử dụng lệnh đào tạo rasa.

#### Rasa NLU

Rasa NLU [6] là thành phần xử lý ngôn ngữ tự nhiên của Rasa, được sử dụng để hiểu và phân tích các thông điệp người dùng. Nó giúp trích xuất yêu cầu (intent) và các thực thể (entities) từ văn bản đầu vào, cho phép hệ thống hiểu được ý định của người dùng và thông tin liên quan.

* Intent Recognition (Nhận dạng ý định): Rasa NLU cho phép bạn xác định ý định (intent) của người dùng từ văn bản đầu vào. Ý định thể hiện mục đích hoặc ý muốn của người dùng trong cuộc trò chuyện. Ví dụ, trong câu "Tôi muốn đặt một cuốn sách", ý định có thể là "Đặt sách".
* Entity Extraction (Trích xuất thực thể): Rasa NLU giúp trích xuất các thực thể (entities) từ văn bản người dùng. Thực thể là các thông tin cụ thể trong văn bản, như tên, địa chỉ, ngày tháng, số lượng, v.v. Ví dụ, trong câu "Tôi muốn đặt một cuốn sách về lịch sử", thực thể có thể là "sách" và "lịch sử".
* Preprocessing (Tiền xử lý): Rasa NLU cung cấp các công cụ và kỹ thuật để tiền xử lý dữ liệu ngôn ngữ tự nhiên trước khi thực hiện nhận dạng ý định và trích xuất thực thể. Điều này bao gồm các quy trình như tokenization (phân tách thành từ), stemming (rút gọn từ), stop word removal (loại bỏ từ dừng), v.v.
* Machine Learning Models (Mô hình học máy): Rasa NLU sử dụng các mô hình học máy để nhận dạng ý định và trích xuất thực thể từ dữ liệu huấn luyện. Các mô hình phổ biến được sử dụng trong Rasa NLU là mô hình máy học có giám sát như Support Vector Machines (SVM), Recurrent Neural Networks (RNN), và Transformers.
* Training Data (Dữ liệu huấn luyện): Để xây dựng mô hình Rasa NLU, bạn cần cung cấp dữ liệu huấn luyện. Đây là tập các câu huấn luyện có nhãn, trong đó mỗi câu được gắn kết với ý định và các thực thể tương ứng. Dữ liệu huấn luyện này được sử dụng để huấn luyện các mô hình nhận dạng ý định và trích xuất thực thể.

#### Rasa Core

Rasa Core [7] là thành phần quản lý luồng trò chuyện và quyết định hành động trong hệ thống trò chuyện. Rasa Core sử dụng các chính sách (policies) để xác định hành động phù hợp dựa trên trạng thái hiện tại của cuộc trò chuyện và ý định của người dùng. Nó cho phép xây dựng các luồng trò chuyện phức tạp và điều chỉnh hành vi của hệ thống trò chuyện dựa trên phản hồi từ người dùng.

* Dialogue Management (Quản lý cuộc trò chuyện): Rasa Core quản lý và điều khiển luồng cuộc trò chuyện giữa hệ thống trò chuyện và người dùng. Nó xác định trạng thái hiện tại của cuộc trò chuyện, lưu trữ thông tin và quyết định hành động tiếp theo dựa trên trạng thái và các thông điệp đầu vào.
* Action Selection (Lựa chọn hành động): Rasa Core xác định hành động cần thực hiện dựa trên trạng thái hiện tại của cuộc trò chuyện và các tùy chọn khác. Nó sử dụng các chính sách (policies) để quyết định hành động phù hợp dựa trên các quy tắc và chiến lược xác định trước.
* Policies (Chính sách): Policies xác định cách Rasa Core quyết định hành động tiếp theo dựa trên trạng thái hiện tại và các tùy chọn khác. Có nhiều chính sách khác nhau có thể được sử dụng, bao gồm "Memoization" (ghi nhớ trạng thái và hành động trước đó), "Fallback" (xử lý các tình huống không rõ ràng hoặc không thể nhận dạng), "Rule-based" (quy tắc dựa trên các tình huống cụ thể), và "Machine Learning-based" (dựa trên mô hình học máy).
* Training Data (Dữ liệu huấn luyện): Để xây dựng mô hình Rasa Core, bạn cần cung cấp dữ liệu huấn luyện. Đây là tập các cuộc trò chuyện huấn luyện có nhãn, trong đó mỗi cuộc trò chuyện bao gồm các trạng thái, hành động và phản hồi tương ứng. Dữ liệu huấn luyện này được sử dụng để huấn luyện các mô hình quyết định hành động của Rasa Core.

### Facebook Developer



Hình . Facebook Develepers

Facebook Developer[9] là một nền tảng và tập hợp các công cụ, tài liệu và tài nguyên được cung cấp bởi Facebook để hỗ trợ các nhà phát triển xây dựng ứng dụng và tích hợp với các sản phẩm của Facebook. Nền tảng này cung cấp các API (Application Programming Interface) và SDK (Software Development Kit) cho phép nhà phát triển tạo ra các ứng dụng, tích hợp tính năng của Facebook vào ứng dụng của mình và sử dụng dịch vụ của Facebook như đăng nhập bằng tài khoản Facebook, chia sẻ nội dung, gửi tin nhắn, phân tích dữ liệu và quảng cáo.

Facebook Developer cung cấp các công cụ và tài nguyên để giúp nhà phát triển xây dựng ứng dụng chất lượng, tương tác với cộng đồng phát triển, và tìm hiểu về các hướng dẫn và tiêu chuẩn của Facebook. Các nhà phát triển có thể tạo ứng dụng di động, ứng dụng web, chatbot, trò chơi và nhiều ứng dụng khác sử dụng các công nghệ của Facebook.

Facebook Developer cũng cung cấp một hệ thống quản lý ứng dụng, cho phép nhà phát triển đăng ký, quản lý và quản lý các ứng dụng của mình trên nền tảng Facebook.

## Đặc tả nhóm chức năng

### UC1 Yêu cầu thông tin môn học

|  |  |  |  |
| --- | --- | --- | --- |
| **UC1** | | **Yêu cầu thông tin môn học** | **Độ phức tạp: Cao** |
| **Mô tả** | | User tiến hành đặt yêu cầu về thông tin môn học với trợ lý ảo CVHT | |
| **Tác nhân** | | User | |
| **Tiền điều kiện** | | Nhập thông tin môn học cần tra cứu | |
| **Hậu điều kiện** | **Thành công** | Trợ lý ảo phản hồi chi tiết thông tin môn học được yêu cầu | |
| **Lỗi** | Trợ lý ảo yêu cầu User kiểm tra lại thông tin môn học | |
| **ĐẶC TẢ CHỨC NĂNG** | | | |
| **Luồng sự kiện chính** | | | |
| **Luồng chính**   * User gửi một yêu cầu thông tin môn học kèm theo tên môn học đến trợ lý ảo CVHT * Trợ lý ảo xác nhận yêu cầu từ User * Phân loại yêu cầu của User là thông tin môn học * Trợ lý ảo sau khi phân loại được yêu cầu thì tiến hành trích xuất dữ liệu tên môn học từ yêu cầu của User * Trợ lý ảo kiểm tra tên môn học vừa nhận được * Nếu hợp lệ truy cập Database trích xuất thông tin môn học và phản hồi thành công, nếu không thì tiến hành luồng phát sinh bên dưới | | | |
| **Luồng sự kiện phát sinh** | | | |
| **Nhập sai tên môn học / Không nhập tên môn học** | | | |
| * Trợ lý ảo CVHT yêu cầu User kiểm tra lại thông tin môn học * User có thể yêu cầu lại thông tin môn học với tên môn học đã nhập đúng | | | |

### UC2 Yêu cầu thông tin môn học có thể đăng ký với điều kiện cung cấp

|  |  |  |  |
| --- | --- | --- | --- |
| **UC2** | | **Yêu cầu thông tin môn học có thể đăng ký với điều kiện cung cấp** | **Độ phức tạp: Cao** |
| **Mô tả** | | User tiến hành gửi yêu cầu với những điều kiện cung cấp thì sẽ có thể ĐKH những môn học gì | |
| **Tác nhân** | | User | |
| **Tiền điều kiện** | | Nhập thông tin điều kiện | |
| **Hậu điều kiện** | **Thành công** | Trợ lý ảo phản hồi lại những môn học có thể ĐKH | |
| **Lỗi** | Yêu cầu kiểm tra lại thông tin | |
| **ĐẶC TẢ CHỨC NĂNG** | | | |
| **Luồng sự kiện chính** | | | |
| **Luồng chính:**   * User gửi yêu cầu đến CVHT * CVHT sẽ phân loại yêu cầu và trích xuất dữ liệu từ yêu cầu của User * CVHT kiểm tra thông tin đã trích xuất được * Nếu hợp lệ thực hiện truy cập Database lấy thông tin các môn học có điều kiện tương ứng với điều kiện do User cung cấp phản hồi thông tin các môn học có thể ĐKH, nếu không hợp lệ thực hiện luồng phát sinh | | | |
| **Luồng sự kiện phát sinh** | | | |
| **Nhập sai thông tin điều kiện** | | | |
| * User nhập thông tin sai dẫn đến CVHT không thể truy xuất dữ liệu từ Database * CVHT yêu cầu User kiểm tra lại thông tin * User có thể gửi lại yêu cầu với thông tin chính xác | | | |

### UC3 Yêu cầu thông tin lộ trình học gợi ý cho sinh viên

|  |  |  |  |
| --- | --- | --- | --- |
| **UC3** | | **Yêu cầu thông tin lộ trình học gợi ý cho sinh viên** | **Độ phức tạp: Cao** |
| **Mô tả** | | User gửi yêu cầu lộ trình học của năm hoặc của kỳ cụ thể của năm cụ thể tới CVHT | |
| **Tác nhân** | | User | |
| **Tiền điều kiện** | | Thông tin năm học hoặc kỳ học năm học cần tra cứu | |
| **Hậu điều kiện** | **Thành công** | Phản hồi danh sách môn học | |
| **Lỗi** | Yêu cầu User kiểm tra lại thông tin cung cấp | |
| **ĐẶC TẢ CHỨC NĂNG** | | | |
| **Luồng sự kiện chính** | | | |
| **Luồng chính**   * User gửi yêu cầu cho CVHT * CVHT phân loại xác định loại yêu cầu của User và tiến hành trích xuất dữ liệu năm học hoặc kỳ học năm học từ yêu cầu * CVHT kiểm tra thông tin vừa trích xuất * Nếu hợp lệ tiến hành truy cập Database dựa vào năm học và kỳ học lấy ra danh sách các môn học sau đó phản hồi lại phía User * Nếu không tiến hành luồng phát sinh | | | |
| **Luồng sự kiện phát sinh** | | | |
| **Nhập sai năm học / Nhập kỳ học nhưng không có năm học** | | | |
| * User cung cấp kỳ học nhưng không có năm học dẫn đến CVHT không thể biết kỳ học của năm học nào nên không thể phản hồi * CVHT phản hồi yêu cầu User kiểm tra lại thông tin | | | |

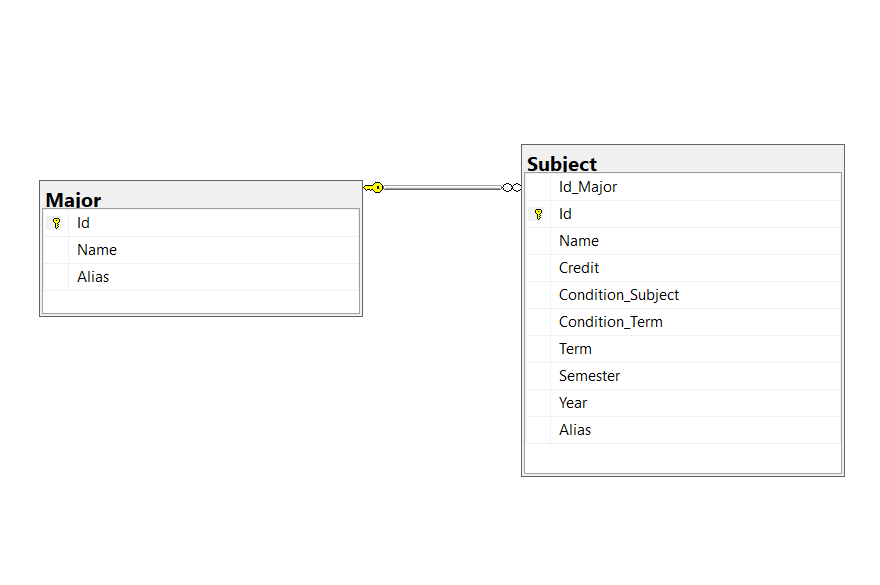
### UC4 Yêu cầu giải đáp một số vấn đề liên quan ĐKH hoặc CTĐT

|  |  |  |  |
| --- | --- | --- | --- |
| **UC4** | | **Yêu cầu giải đáp một số vấn đề liên quan ĐKH** | **Độ phức tạp: Trung bình** |
| **Mô tả** | | User yêu cầu CVHT gửi đáp một số vấn đề liên quan CTĐT hoặc ĐKH | |
| **Tác nhân** | | User | |
| **Tiền điều kiện** | | Không có | |
| **Hậu điều kiện** | **Thành công** | CVHT phản hồi giải đáp thắc mắc của User | |
| **Lỗi** | Không có | |
| **ĐẶC TẢ CHỨC NĂNG** | | | |
| **Luồng sự kiện chính/Kịch bản chính** | | | |
| **Luồng chính**   * User yêu cầu CVHT giải đáp thắc mắc * CVHT phân loại yêu cầu giải đáp và dựa vào yêu cầu để phản hồi giải đáp User | | | |

# Thiết kế cơ sở dữ liệu và tạo dữ liệu huấn luyện

## Thiết kế Cơ sở dữ liệu

### Sơ đồ Diagram của Database



Hình . Diagram

### Danh sách các bảng

| **STT** | **Tên bảng** | **Mô tả** |
| --- | --- | --- |
| **1** | Major | Danh sách lưu thông tin các ngành học |
| **2** | Subject | Danh sách lưu thông tin các môn học |

### Chi tiết các bảng

#### Major

|  |  |  |  |
| --- | --- | --- | --- |
| **STT** | **Tên trường** | **Kiểu dữ liệu** | **Ý nghĩa** |
| **1** | Id | NVARCHAR(255) | Mã định danh của chuyên ngành |
| **2** | Name | NVARCHAR(255) | Tên của chuyên ngành |
| **3** | Alias | NVARCHAR(255) | Bí danh của chuyên ngành |

#### Subject

|  |  |  |  |
| --- | --- | --- | --- |
| **STT** | **Tên trường** | **Kiểu dữ liệu** | **Ý nghĩa** |
| **1** | Id\_Major | NVARCHAR (255) | Mã định danh của chuyên ngành |
| **2** | Id | NVARCHAR (255) | Mã môn học |
| **3** | Name | NVARCHAR(255) | Tên môn học |
| **4** | Credit | INT | Số tín chỉ của môn học |
| **5** | Condition\_Subject | NVARCHAR(255) | Môn học điều kiện tiên quyết |
| **6** | Condition\_Term | INT | Số tín chỉ tiên quyết |
| **7** | Term | INT | Mã số loại học phần |
| **8** | Semester | INT | Kỳ học gợi ý nên học |
| **9** | Year | INT | Năm học gợi ý nên học |
| **10** | Alias | NVARCHAR(255) | Bí danh môn học |

## Tạo dữ liệu huấn luyện

Giới thiệu Folder data trong Rasa:

* nlu.yml: File này bao gồm các thành phần sau:
* intent: ý định hay còn gọi là loại yêu cầu của User trong Machine Learning có thể hiểu là tương tự như class hay nhãn.
* examples: nơi ta liệt kê các loại câu hỏi của intent tức nhiều hình thức câu từ khác nhau nhưng đều mang chung một ý nghĩa
* stories.yml:
* story: định nghĩa câu chuyện, đoạn hội thoại mà ta muốn trợ lý ảo đi theo hay có thể hiểu là kịch bản được viết ra để điều khiển câu chuyện
* steps: các bước của kịch bản được xây dựng từ trước
* intent: ý định của người dùng
* actions: trợ lý ảo trả lời
* rules.yml: các luật được chúng ta đặt ra bắt buộc trợ lý ảo phải tuân theo khi gặp một intent cụ thể của User

Tiếp theo chúng ta tìm hiểu đến một thành phần quan trong khác của Rasa trong file domain.yml:

* Actions: Liệt các custom actions - một thành phần quan trọng khác mà ta sẽ nói đến ở phần sau
* Intent: Liệt kê các intent của User
* Respone: Trợ lý ảo thực hiện phản hồi
* Slots: là các vị trí lưu trữ thông tin trong chatbot. Mỗi slot tương ứng với một kiểu dữ liệu cụ thể như ngày tháng, chuỗi ký tự hoặc số. Khi chatbot tương tác với người dùng, các giá trị entities được trích xuất sẽ được gán vào các slots tương ứng. Việc lưu trữ các giá trị entities trong slots giúp chatbot duy trì thông tin của người dùng trong suốt quá trình hội thoại, cho phép nó theo dõi và sử dụng các thông tin này để thực hiện các hành động phù hợp
* Entities: là các thành phần quan trọng trong một câu truy vấn hoặc thông điệp, đại diện cho các giá trị cụ thể của dữ liệu mà chatbot cần trích xuất từ đó. Entities có thể là các thông tin như tên, địa chỉ, thời gian, số lượng, và nhiều hơn nữa. Khi bạn xác định và trích xuất entities từ câu truy vấn, chatbot có thể sử dụng thông tin này để hiểu ý định của người dùng và cung cấp các phản hồi phù hợp

Ở chúng ta sẽ xây dựng dữ liệu huấn luyện file data/nlu.yml

### Dữ liệu cho intent: Yêu cầu thông tin môn học

Ví dụ:

|  |  |  |
| --- | --- | --- |
| **STT** | **Data** | **Slot** |
| 1 | Thông tin [CĐTN: Phát triển phần mềm](subject) | Subject – Text lưu chữ tên môn học |
| 2 | Thêm thông tin về chuyên đề [CĐTN: Cơ sở dữ liệu](subject) là gì? | subject |
| 3 | Tôi cần cụ thể chuyên đề [CĐTN: Cơ sở dữ liệu](subject) | subject |
| 4 | Mình cần thông tin chuyên đề [CĐTN: Phát triển phần mềm](subject) | subject |

### Dữ liệu cho intent: Yêu cầu thông tin môn học với điều kiện tiên quyết

Ví dụ:

|  |  |  |
| --- | --- | --- |
| **STT** | **Data** | **Slot** |
| 1 | Hoàn thành các môn học [Đồ họa](list), [An toàn thông tin](list), [An toàn mạng](list) | list – Một danh sách chứa các môn |
| 2 | [101 tín chỉ](credit) và qua các môn [Kiến trúc máy tính](list) và [Nguyên lý hệ điều hành](list) | credit – Text lưu trữ số tín chỉ  list |
| 3 | Tớ học môn [Học máy](list) | list |
| 4 | Đã qua môn [Học máy](list) và [90 tín](credit) | list  credit |
| 5 | Tôi có thể học được những gì khi đã học các môn [Tiếng Trung 1](list), [Tiếng Ý 1](list), [Tiếng Pháp 2](list)? | list |

### Dữ liệu cho intent: Yêu cầu lộ trình học

Ví dụ:

|  |  |  |
| --- | --- | --- |
| STT | Data | Slot |
| 1 | Với năm [2](year) kỳ [2](semester), tôi cần chọn những môn học nào? | year – Text  semester - Text |
| 2 | Cho tôi xin gợi ý về môn học phù hợp với năm [bốn](year) và kỳ [ba](semester) của tôi được không? | year – Text  semester - Text |
| 3 | Năm [3](year) học gì? | year – Text |
| 4 | Tôi học [2 năm rưỡi](ysem) nên học gì? | ysem – Text lưu trữ năm và học kỳ |
| 5 | Tôi học được [nửa năm](ysem) nên học gì? | ysem |

### Dữ liệu cho intent: Yêu cầu giải đáp một số thắc mắc CTĐT hoặc ĐKH

Ví dụ:

|  |  |  |
| --- | --- | --- |
| STT | Data | Slot |
| 1 | Tại sao học phần [Lập trình hướng đối tượng](name\_subject) không mở cửa đăng ký học cho sinh viên? | name\_subject – Text lưu trữ tên môn học |
| 2 | Vì sao sinh viên không thể xem học phần A trong lịch học cá nhân dù đã đăng ký thành công? |  |
| 3 | Vì sao sinh viên nhóm 1, 2 không được phép đăng ký học cùng nhóm 3 dù học phần này đã mở trong thời khóa biểu của nhóm 1, 2? |  |
| 4 | Để tránh việc trễ hạn và sắp xếp tài chính, sinh viên nhóm 1, 2 khi học hoặc thi lại cùng nhóm 3 nên thực hiện đóng học phí và lệ phí thi lại các học phần theo quy định của trường. |  |
| 5 | Sinh viên có điểm tổng kết học phần Tiếng Anh sơ cấp 2 (GE102) < 4, nhưng sau đó đã học, thi và được cấp chứng chỉ tiếng Anh quốc tế thì có được quy đổi điểm học phần tiếng Anh sơ cấp 1, 2 không? |  |

# Thiết kế Actions

## Giới thiệu chung

### Custom Actions

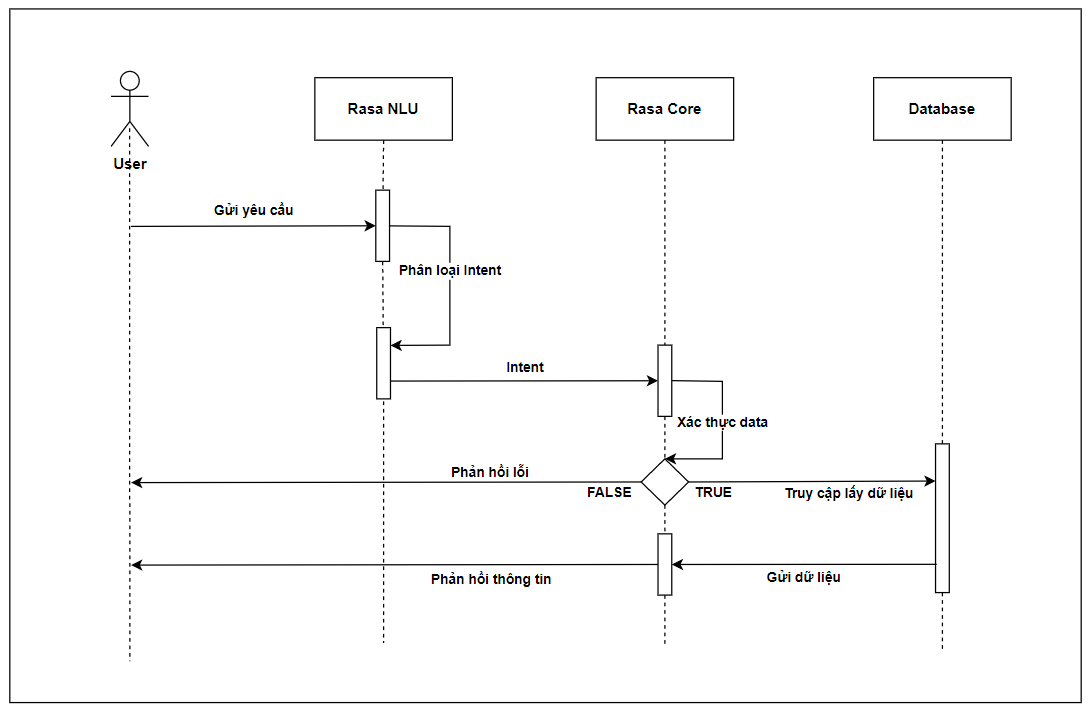
Custom Actions trong Rasa là các hành động tùy chỉnh mà bạn có thể định nghĩa để trợ lý ảo của bạn thực hiện trong quá trình hội thoại. Khi trợ lý ảo nhận được thông điệp từ người dùng, nó sẽ đánh giá ý định và trích xuất các Entities. Sau đó, nó có thể thực hiện một hoặc nhiều Custom Actions để cung cấp phản hồi phù hợp hoặc thực hiện các tác vụ khác nhau.

Custom Actions cho phép bạn tương tác với các API bên ngoài, cơ sở dữ liệu hoặc bất kỳ hệ thống nào khác để lấy thông tin, cập nhật dữ liệu hoặc thực hiện hành động phức tạp. Bạn có thể xử lý logic kinh doanh, truy vấn cơ sở dữ liệu, gọi API bên ngoài, gửi email và thực hiện nhiều tác vụ khác thông qua Custom Actions.

Custom Actions được định nghĩa trong file actions/actions.py.

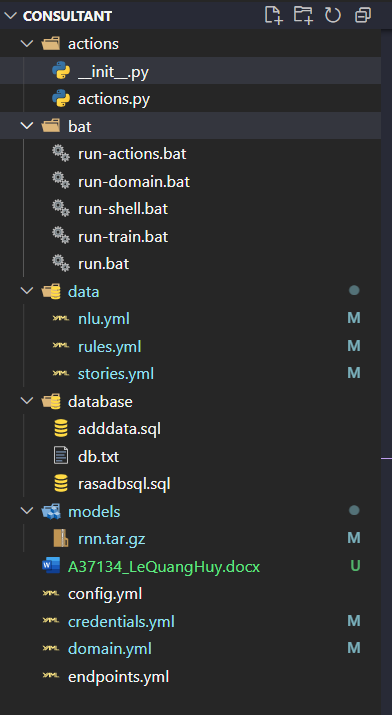
### Thiết kế chức năng

* Sơ đồ tuần tự chung cho các chức năng



Hình . Sơ đồ tuần tự chức năng giải đáp yêu cầu của User

## Cấu trúc thư mục



Hình . Cấu trúc thư mục Rasa

## Thuật toán trong Custom Actions

### Mục tiêu

Chúng ta cần phải linh hoạt trong câu trả lời ví dụ như khi nhận câu “Xin chào” thì trợ lý sẽ trả lời “Chào bạn tôi có thể giúp gì được cho bạn?”. Đây là trường hợp cơ bản vì khi xác định được “Xin chào” là Intent greet câu phản hồi của trợ lý ảo có thể cố định ví dụ như trên. Nhưng trong trường hợp đặc biệt hơn như “Cho tôi thông tin môn Học máy” trợ lý ảo sẽ trả lời “Thông tin môn Học máy là…” nhưng với câu “Hãy cho tôi chi tiết môn Giải tích 1” câu trả lời sẽ nhận được là “Thông tin môn Giải tích 1 là…” như ta có thể thấy tên môn học ở đây là biến thay đổi tuỳ vào câu hỏi và thông tin môn học được lấy từ Database vậy câu phản hồi của trợ lý ảo trong trường hợp này là không thể để cố định.

Để giải quyết vấn đề này chúng ta đã biết đến khái niệm Custom Actions trong Rasa đã được giới thiệu ở trên.

### Giải pháp

* Sử dụng Custom Actions trong Rasa
* Để giải quyết vấn đề trên và sử dụng Custom Actions chúng ta trước tiên cần:

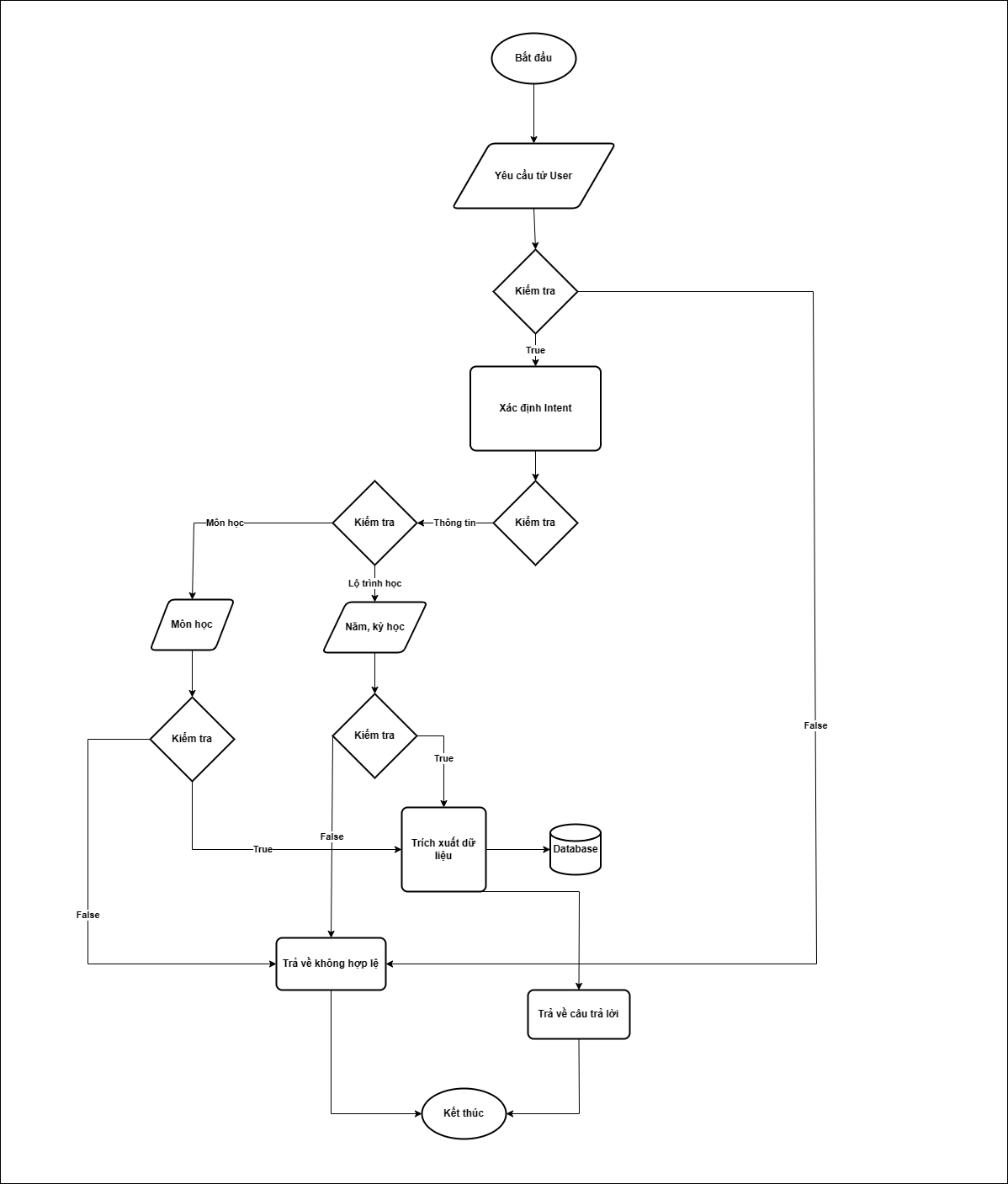
#### Phân tích dữ liệu đầu vào và đầu ra

* Đầu vào - Input:
* *Yêu cầu của User*: Là câu yêu cầu đưa vào của User để dành cho CVHT, phân loại ra Intent của User
* *Dữ liệu từ yêu cầu của User*: Từ yêu cầu của User CVHT ta có được Intent sau đó sẽ lọc ra các Keyword và lấy ra dữ liệu từ yêu cầu của User
* Đầu ra - Output: *Thông tin giải đáp cho yêu cầu của User.*
* Thông tin có thể là môn học, giải đáp thắc mắc hoặc thông tin lộ trình học.

#### Thiết kế thuật toán

**Thuật toán:**

* Bước 1: Tiếp nhận yêu cầu từ User.
* Bước 2: Dự đoán Intent của yêu cầu, nếu yêu cầu hợp lệ thì chuyển bước 3 nếu không hợp lệ thì chuyển bước 12.
* Bước 3: Từ xác định được Intent nếu là loại yêu cầu lộ trình học thì chuyển bước 4, dạng hỏi thông tin môn học chuyển bước 7, những dạng câu hỏi cần giải đáp thắc mắc chuyển bước 9.
* Bước 4: Từ yêu cầu loại ra các thông tin như năm học, kỳ học hoặc môn học.
* Bước 5: Kiểm tra năm học hợp lệ hoặc có năm học và kỳ học hợp lệ thì chuyển bước 6 trong trường hợp sai chuyển bước 12.
* Bước 6: Truy cập cơ sở dữ liệu truy xuất dữ liệu tương ứng với lộ trình học yêu cầu chuyển bước 11.
* Bước 7: Kiểm tra thông tin môn học vừa được đề cập, nếu hợp lệ chuyển bước 8 trường hợp còn loại chuyển bước 12.
* Bước 8: Truy cập cơ sở dữ liệu truy xuất dữ liệu tương ứng với thông tin môn học được yêu cầu chuyển bước 11.
* Bước 9: Với dạng yêu cầu là giải đáp một số thông tin khác cần xác định Intent, chuyển bước 10.
* Bước 10: Dựa vào Intent đã xác định trả lời thắc mắc của User, chuyển bước 11.
* Bước 11: Từ các câu trả lời đã có được in ra màn hình cho User.
* Bước 12: In ra yêu cầu không hợp cho User.



Hình . Sơ đồ khối thuật toán nhận diện yêu cầu và trả về phản hồi

# Machine learning

## Giới thiệu chung

Học máy là một lĩnh vực của trí tuệ nhân tạo liên quan đến việc nghiên cứu và xây dựng các kĩ thuật cho phép các hệ thống "học" tự động từ dữ liệu để giải quyết những vấn đề cụ thể. Các thuật toán học máy xây dựng một mô hình dựa trên dữ liệu mẫu, được gọi là dữ liệu huấn luyện, để đưa ra dự đoán hoặc quyết định mà không cần được lập trình chi tiết về việc đưa ra dự đoán hoặc quyết định này. Ví dụ như các máy có thể "học" cách phân loại thư điện tử xem có phải thư rác (spam) hay không và tự động xếp thư vào thư mục tương ứng.[10]

Các loại giải thuật:

* Học có giám sát: là một kĩ thuật của ngành học máy để xây dựng một hàm (function) từ dữ liệu huấn luyện. Dữ liệu huấn luyện bao gồm các cặp gồm đối tượng đầu vào (thường dạng vec-tơ), và đầu ra mong muốn.[11]
* Học không có giám sát: là một phương pháp của ngành học máy nhằm tìm ra một mô hình mà phù hợp với các quan sát.[1] Nó khác biệt với học có giám sát ở chỗ là đầu ra đúng tương ứng cho mỗi đầu vào là không biết trước. Trong học không có giám sát, một tập dữ liệu đầu vào được thu thập. Học không có giám sát thường đối xử với các đối tượng đầu vào như là một tập các biến ngẫu nhiên. Sau đó, một mô hình mật độ kết hợp sẽ được xây dựng cho tập dữ liệu đó.[12]
* Học tăng cường: là một lĩnh vực con của học máy, nghiên cứu cách thức một agent trong một môi trường nên chọn thực hiện các hành động nào để cực đại hóa một khoản thưởng (reward) nào đó về lâu dài. Các thuật toán học tăng cường cố gắng tìm một chiến lược ánh xạ các trạng thái của thế giới tới các hành động mà agent nên chọn trong các trạng thái đó.[13]

Chúng ta sẽ sử dụng giải thuật học có giám sát.

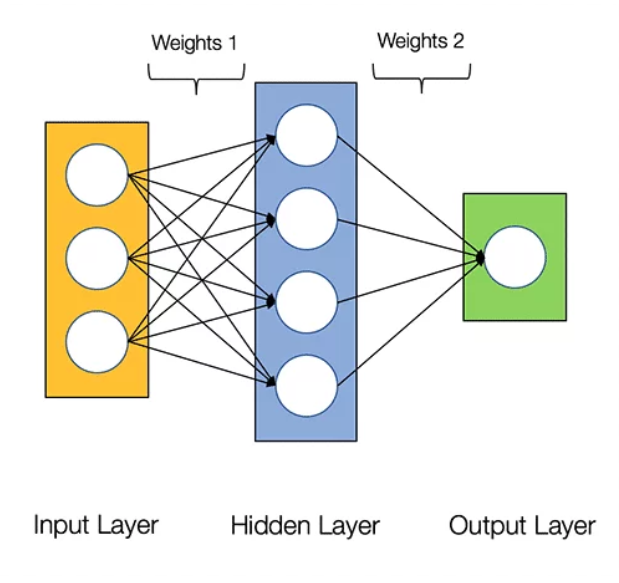
## Mô hình Neural Network

### Giới thiệu mô hình

Neural Network (Mạng nơ-ron nhân tạo), đây là một chuỗi những thuật toán được đưa ra để tìm kiếm các mối quan hệ cơ bản trong tập hợp các dữ liệu. Thông qua việc bắt bước cách thức hoạt động từ não bộ con người. Nói cách khác, mạng nơ ron nhân tạo được xem là hệ thống của các tế bào thần kinh nhân tạo. Đây thường có thể là hữu cơ hoặc nhân tạo về bản chất.[14]

Mạng Neural Network là sự kết hợp của những tầng perceptron hay còn gọi là perceptron đa tầng. Và mỗi một mạng Neural Network thường bao gồm 3 kiểu tầng là:

* Tầng input layer (tầng vào): Tầng này nằm bên trái cùng của mạng, thể hiện cho các đầu vào của mạng.
* Tầng output layer (tầng ra): Là tầng bên phải cùng và nó thể hiện cho những đầu ra của mạng.
* Tầng hidden layer (tầng ẩn): Tầng này nằm giữa tầng vào và tầng ra nó thể hiện cho quá trình suy luận logic của mạng.



Hình . Kiến trúc Neural Network

Lưu ý: Mỗi một Neural Network chỉ có duy nhất một tầng vào và 1 tầng ra nhưng lại có rất nhiều tầng ẩn.

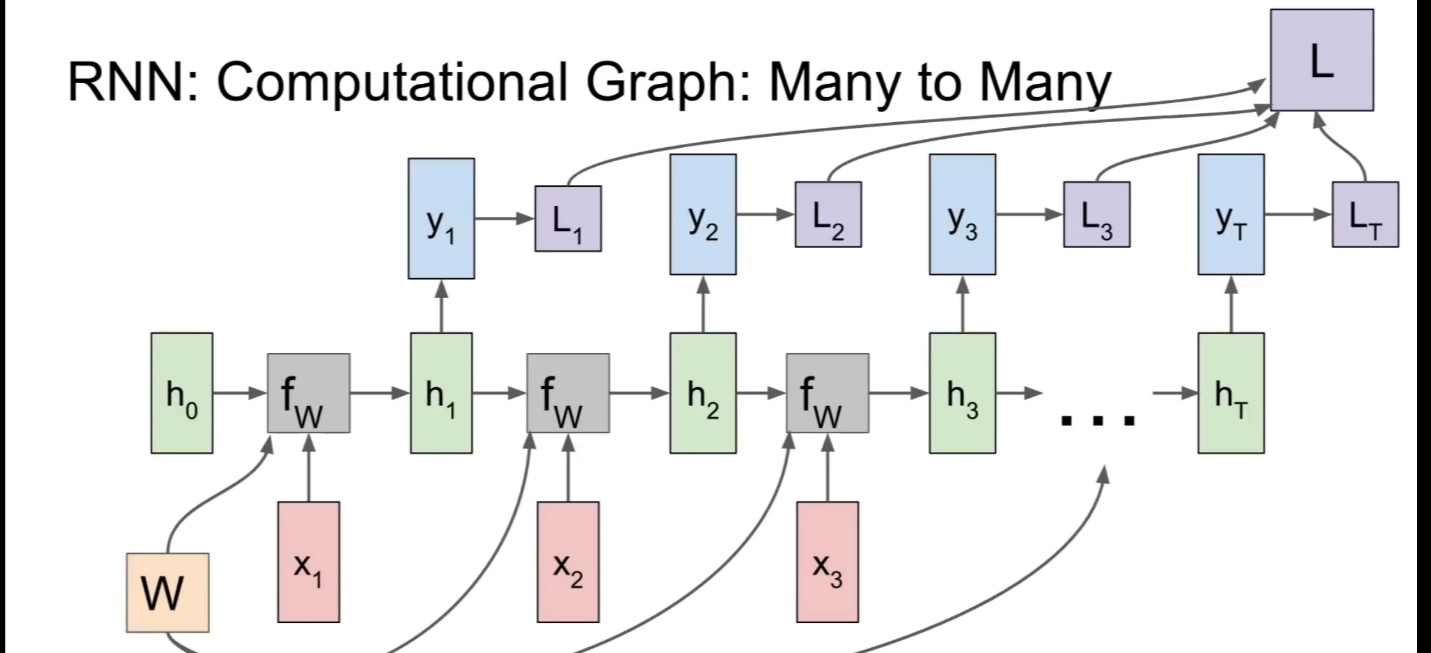
Với mạng Neural Network thì mỗi nút mạng là một sigmoid nơron nhưng chúng lại có hàm kích hoạt khác nhau. Thực tế, người ta thường sử dụng có cùng loại với nhau để việc tính toán thuận lợi hơn. Tại mỗi tầng, số lượng nút mạng có thể khác nhau còn tùy vào bài toán hoặc cách giải quyết.

### Mô hình Recurrent Neural Network

Recurrent Neural Network hay còn gọi là mạng hồi quy RNN, một lớp của NN, nơi kết nối giữa các nút để tạo thành đồ thị có hướng dọc theo một trình tự thời gian. Điều này cho phép mạng thể hiện hành vi động tạm thời.[15]

Phân loại RNN:[16]

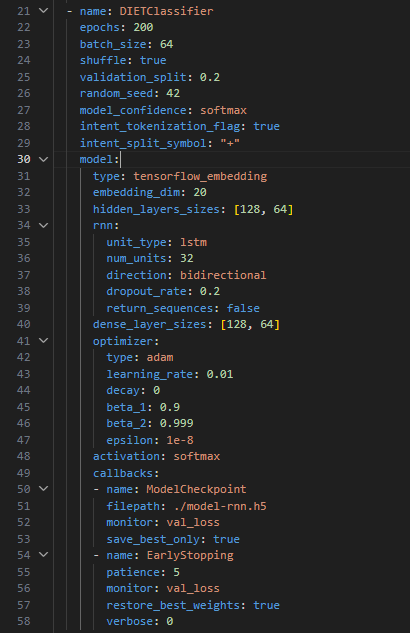
* One to one: mẫu bài toán cho NN và và Convolutional Neural Network (CNN), 1 input và 1 output, ví dụ 1 ảnh đầu vào và đầu ra là 1 nhãn cho ảnh.
* One to many: mẫu bài toán có 1 input nhưng nhiều output, ví dụ: bài toán caption cho ảnh, input là 1 ảnh nhưng output là nhiều chữ mô tả cho ảnh đấy, dưới dạng một câu.
* Many to one: bài toán có nhiều input nhưng chỉ có 1 output, ví dụ bài toán phân loại hành động trong video, input là nhiều ảnh (frame) tách ra từ video, ouptut là hành động trong video.
* Many to many: bài toán có nhiều input và nhiều output, ví dụ bài toán dịch từ tiếng anh sang tiếng việt, input là 1 câu gồm nhiều chữ: “I love Vietnam” và output cũng là 1 câu gồm nhiều chữ “Tôi yêu Việt Nam”.



Hình . Mô hình RNN Many to many

Trong bài toán này, chúng ta sẽ tập trung vào Many to many.

Xây dựng Model trong bài:



Hình . Mô hình RNN

Mô hình trong Rasa có tên là "DIETClassifier" và được cấu hình với các tham số như sau:

* epochs: Số lượng vòng lặp huấn luyện của mô hình là 200.
* batch\_size: Kích thước của mỗi batch được sử dụng trong quá trình huấn luyện là 64.
* shuffle: Dữ liệu huấn luyện sẽ được xáo trộn sau mỗi epoch.
* validation\_split: Tỷ lệ phần trăm dữ liệu sẽ được sử dụng cho việc đánh giá mô hình trong quá trình huấn luyện là 20%.
* random\_seed: Một giá trị được sử dụng để khởi tạo ngẫu nhiên và đảm bảo sự nhất quán của quá trình huấn luyện.
* model\_confidence: Hàm softmax được sử dụng để tính toán độ tin cậy của các dự đoán của mô hình.

Mô hình được xây dựng bằng cách sử dụng kiến trúc "tensorflow\_embedding" với các tham số sau:

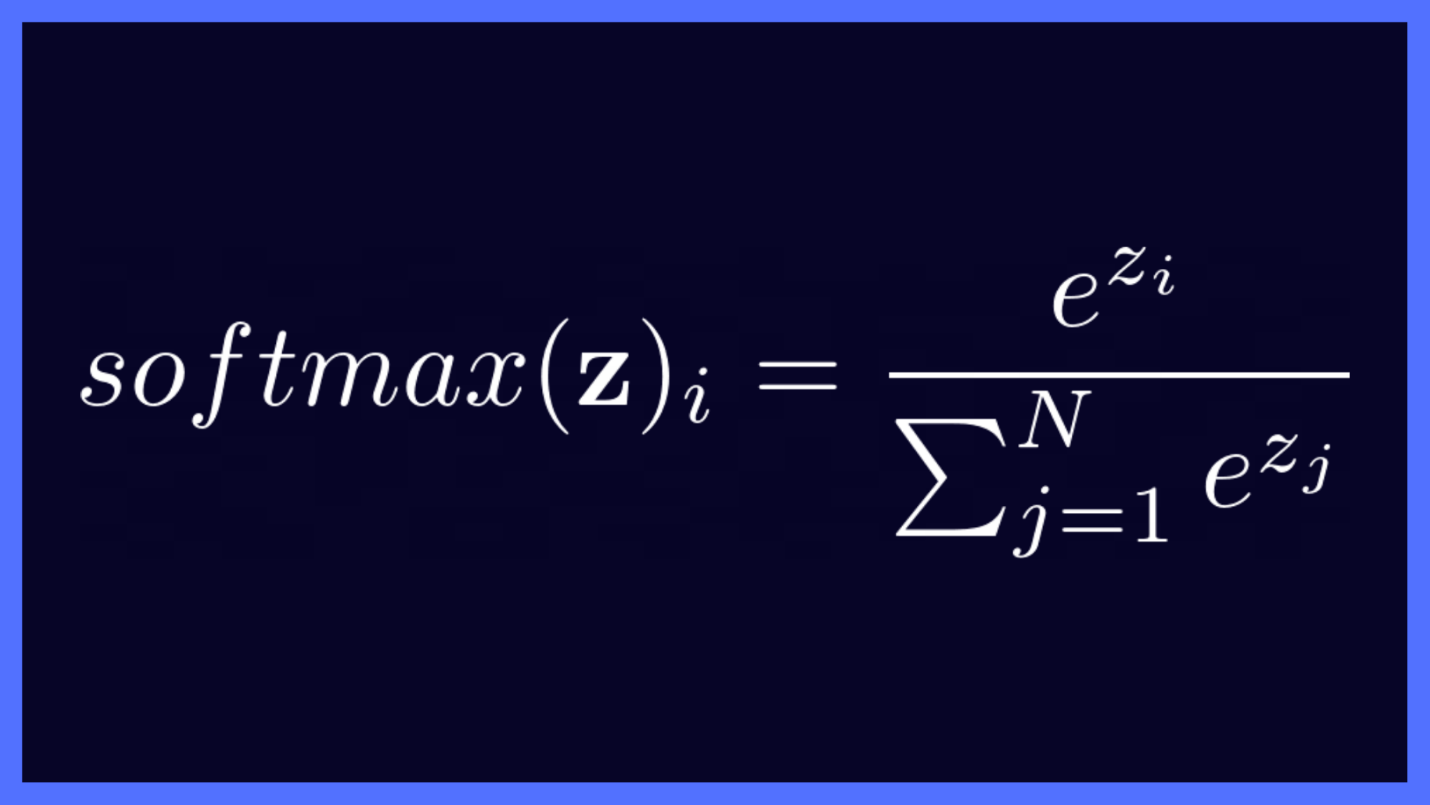
* embedding\_dim: Kích thước của không gian nhúng từ (word embeddings) là 20.
* hidden\_layers\_sizes: Kích thước của các tầng ẩn (hidden layers) là [128, 64].
* rnn: mạng RNN sử dụng với các tham số sau:
* unit\_type: Loại đơn vị RNN được sử dụng là LSTM (Long Short-Term Memory).
* num\_units: Số lượng đơn vị trong mỗi tầng LSTM là 32.
* direction: Mô hình sử dụng mạng LSTM song hướng (bidirectional) để học biểu diễn từ cả văn bản đầu vào và đầu ra.
* dropout\_rate: Tỷ lệ dropout là 0.2, giúp tránh overfitting trong quá trình huấn luyện.
* return\_sequences: Mô hình chỉ trả về kết quả ở output cuối cùng của LSTM.
* dense\_layer\_sizes: Kích thước của các tầng fully connected (dense layers) sau khi mạng RNN đã được thực hiện là [128, 64].
* optimizer: Một thuật toán tối ưu hóa Adam được sử dụng với các tham số sau:

learning\_rate: Tốc độ học của thuật toán Adam là 0.01.

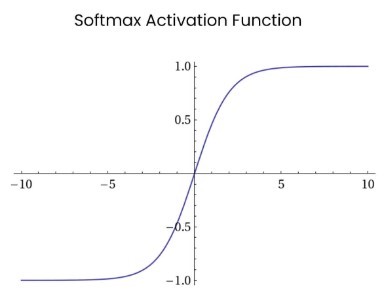
decay: Hệ số giảm tốc độ học sau mỗi epoch là 0.

beta\_1 và beta\_2: Các hệ số điều chỉnh trong thuật toán Adam là 0.9 và 0.999 tương ứng.

* epsilon: Một hằng số nhỏ được sử dụng để tránh chia cho 0.
* activation: Hàm kích hoạt softmax được sử dụng cho tầng đầu ra của mô hình.



Hình . Công thức của hàm Softmax



Hình . Đồ thị của hàm Softmax

Mô hình được cấu hình với các callbacks sau:

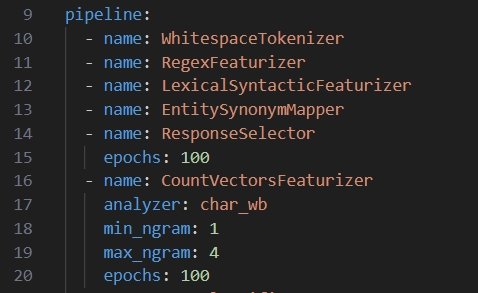
* ModelCheckpoint: Lưu trữ trọng số của mô hình vào tệp tin "./model-rnn.h5" sau mỗi epoch nếu chỉ số val\_loss (hàm mất mát trên tập validation) tốt hơn ở epoch trước đó.
* EarlyStopping: Dừng quá trình huấn luyện nếu không có sự cải thiện trong val\_loss sau 5 epoch liên tiếp. Trọng số tốt nhất được khôi phục và sử dụng để đưa ra dự đoán cuối cùng.

### Các thành phần khác trong Pipelines và Policies

Pipelines và Policies[17]

#### Pipelines

Trong Rasa, pipeline là một danh sách các thành phần được sử dụng để xử lý và trích xuất thông tin từ đầu vào ngôn ngữ tự nhiên và dự đoán intent, thực thể, và hành động tương ứng. Các thành phần trong pipeline hoạt động theo thứ tự từ trên xuống dưới, với đầu ra của thành phần trước được đưa vào làm đầu vào cho thành phần tiếp theo.



Hình . Pipelines phần 1

WhitespaceTokenizer: Đây là một tokenizer đơn giản, dùng để tách các từ trong câu dựa trên khoảng trắng. Ví dụ: "Hello world" sẽ được tách thành ["Hello", "world"].

RegexFeaturizer: Đây là một phần tử được sử dụng để trích xuất đặc trưng từ văn bản sử dụng các biểu thức chính quy. Nó có thể sử dụng các biểu thức chính quy để tìm kiếm và đánh dấu các mẫu, ví dụ như đánh dấu các từ có chứa số, các từ in hoa, các từ có ký tự đặc biệt, v.v.

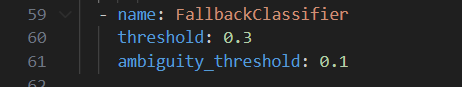
LexicalSyntacticFeaturizer: Đây là một phần tử để trích xuất đặc trưng từ thông tin từ vựng và cú pháp của câu. Nó có thể sử dụng các kỹ thuật như POS tagging (gán nhãn phần từ), dependency parsing (phân tích cú pháp phụ thuộc), sentiment analysis (phân tích cảm xúc) để tạo ra các đặc trưng dựa trên thông tin ngôn ngữ.

EntitySynonymMapper: Đây là một phần tử được sử dụng để ánh xạ các đồng nghĩa của các thực thể. Nó có thể dùng để gán nhãn đồng nghĩa cho các từ hoặc cụm từ có cùng ý nghĩa, giúp cải thiện quá trình hiểu và xử lý ngôn ngữ tự nhiên.

ResponseSelector: Đây là một phần tử được sử dụng để chọn phản hồi phù hợp từ một tập hợp các câu trả lời. Nó có thể dựa trên các đặc trưng từ văn bản đầu vào để dự đoán câu trả lời tốt nhất cho câu hỏi hoặc yêu cầu của người dùng. Trong trường hợp này, mô hình được huấn luyện trong 100 epoch để tối ưu hóa quá trình chọn phản hồi.

CountVectorsFeaturizer là một phần tử trong pipeline xử lý ngôn ngữ tự nhiên (NLP) được sử dụng để trích xuất đặc trưng từ văn bản đầu vào. Dưới đây là cấu hình chi tiết cho CountVectorsFeaturizer:

* name: Tên của phần tử CountVectorsFeaturizer.
* analyzer: Phương pháp phân tích (analyzer) được sử dụng để tách từ và xây dựng các n-gram. Trong trường hợp này, char\_wb được sử dụng, nghĩa là từ sẽ được chia thành các ký tự và chỉ xây dựng n-gram nếu nó xuất hiện trong từ hoặc ở đầu hoặc cuối từ. Ví dụ: từ "example" sẽ được chia thành các ký tự ['e', 'x', 'a', 'm', 'p', 'l', 'e'] và tạo ra các n-gram ['e', 'x', 'a', 'm', 'p', 'l', 'e', 'ex', 'xa', 'am', 'mp', 'pl', 'le', 'exa', 'xam', 'amp', 'mpl', 'ple', 'exam', 'xamp', 'ampl', 'mple'].
* min\_ngram: Số lượng tối thiểu của n-gram để xây dựng. Trong trường hợp này, giá trị là 1, cho nghĩa là chỉ xây dựng các ký tự đơn lẻ.
* max\_ngram: Số lượng tối đa của n-gram để xây dựng. Trong trường hợp này, giá trị là 4, cho nghĩa là xây dựng các n-gram từ ký tự đơn lẻ đến 4 ký tự.
* epochs: Số lượng vòng lặp huấn luyện của CountVectorsFeaturizer. Trong trường hợp này, giá trị là 100.



Hình . Pipelines phần 2

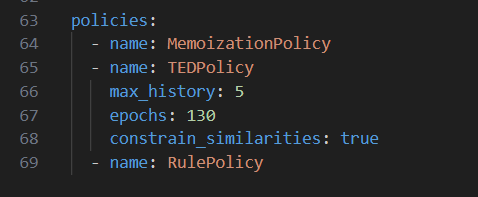
FallbackClassifier là một thành phần được sử dụng để xử lý các trường hợp không chắc chắn hoặc mơ hồ trong việc phân loại intent. Thông qua việc đặt ngưỡng (threshold) và ngưỡng mơ hồ (ambiguity threshold), FallbackClassifier quyết định liệu một câu được xem là mơ hồ hay không và xử lý nó một cách phù hợp.

* threshold: 0.3: Đây là ngưỡng được sử dụng để xác định liệu một câu được phân loại intent hay không. Nếu xác suất phân loại intent cao hơn ngưỡng này, câu sẽ được coi là được phân loại thành công. Nếu xác suất phân loại intent thấp hơn ngưỡng này, FallbackClassifier sẽ được kích hoạt.
* ambiguity\_threshold: 0.1: Đây là ngưỡng mơ hồ được sử dụng để xác định liệu một câu có mức độ mơ hồ cao hay không. Nếu xác suất phân loại intent tương đối gần nhau (chênh lệch nhỏ hơn ngưỡng mơ hồ), câu sẽ được xem là mơ hồ và FallbackClassifier sẽ thực hiện các hành động liên quan đến việc xử lý mơ hồ.

Khi một câu không được phân loại intent thành công hoặc được xem là mơ hồ, FallbackClassifier sẽ thực hiện các hành động như trả về intent fallback (intent dự phòng) hoặc thực hiện các hành động xử lý mơ hồ khác, ví dụ như yêu cầu người dùng cung cấp thêm thông tin hoặc giúp đỡ.

#### Policies

Trong Rasa, các thành phần "policies" là các thành phần quyết định trong việc xử lý hoạt động của trợ lý ảo. Chúng xác định cách trợ lý ảo phản hồi và tương tác với người dùng dựa trên dữ liệu huấn luyện và các quy tắc được định nghĩa trước.



Hình . Policies

MemoizationPolicy: Đây là một chính sách đơn giản dựa trên việc nhớ các hành động đã được thực hiện trước đó. Nó lưu trữ các đường dẫn hành động trước đó trong bộ nhớ và thực hiện các hành động tương ứng nếu tìm thấy một tình huống tương tự trong lịch sử tương tác. Chính sách này hữu ích trong việc xử lý các trường hợp đã được huấn luyện một cách rõ ràng và cần phản hồi nhanh chóng.

TEDPolicy: Đây là một chính sách sử dụng mô hình học sâu "Transformer Embedding Dialogue" (TED) để dự đoán intent và hành động dựa trên lịch sử tương tác. max\_history là số lượng câu trước đó được sử dụng để xây dựng bộ nhớ cho mô hình TED. epochs chỉ định số lần huấn luyện mô hình. constrain\_similarities là một cờ để hạn chế độ tương đồng của các vector nhúng (embeddings) được sử dụng trong mô hình.

RulePolicy: Đây là một chính sách dựa trên quy tắc được định nghĩa trước để xác định các hành động dựa trên mẫu và điều kiện. Quy tắc được định nghĩa theo cú pháp của Rasa và được sử dụng để phân loại intent và thực hiện hành động tương ứng.

Các thành phần "policies" này được sắp xếp theo thứ tự ưu tiên từ trên xuống dưới trong pipeline xử lý của Rasa. Khi một câu được nhận diện, Rasa sẽ thực hiện các chính sách theo thứ tự để xác định hành động phù hợp để trả lời hoặc tương tác với người dùng.

# Cài đặt và thực nghiệm

## Cài đặt

Bước 1: Training model bằng rasa train –fixed-model-name tenmodel.

Bước 2: Chạy lệnh rasa run actions và rasa run.

Bước 3: Chạy ngrok http 5005.

Bước 4: Truy cập Facebook Developer khởi tạo ứng dụng khởi tạo ứng dụng.

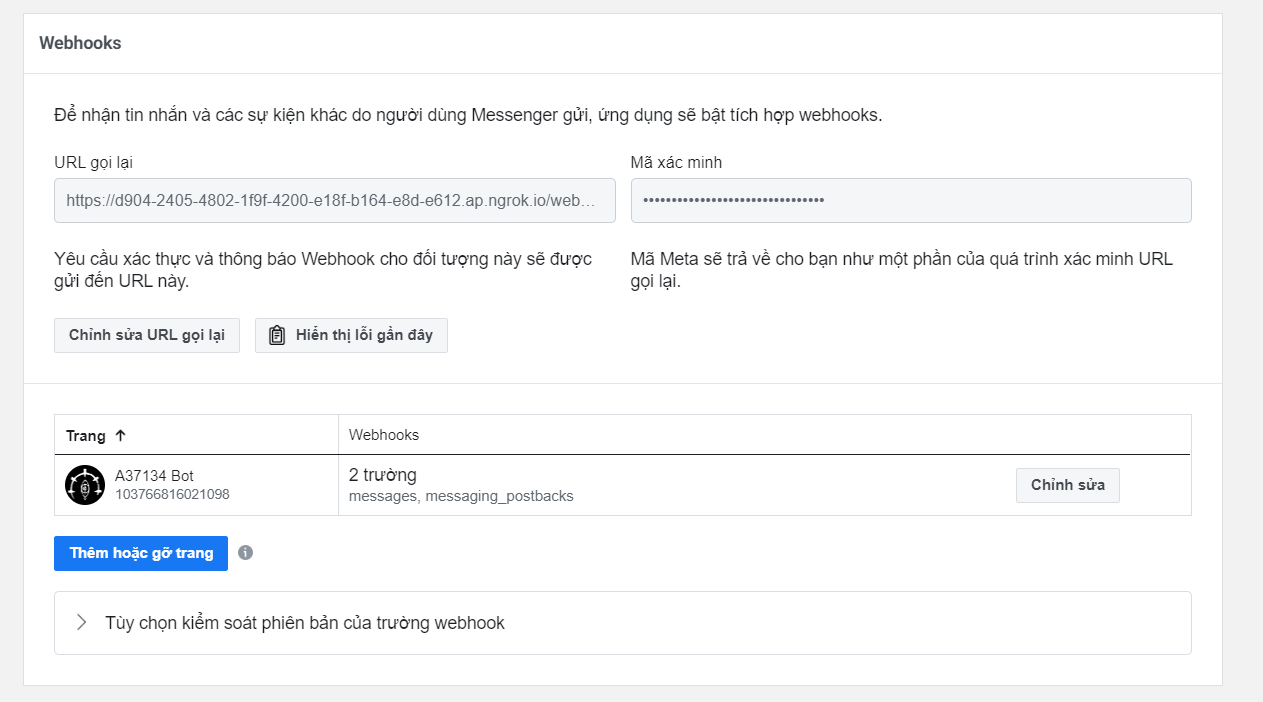
Bước 5: Chọn messenger chọn cài đặt.

Bước 6: Tại mục mã truy cập tiến hành tạo một Page mới cần thêm 2 trường messages và messaging\_postbacks để bot có thể trả lời tin nhắn.



Hình . Giao diện mã truy cập

Bước 7: Khởi tạo Webhooks.

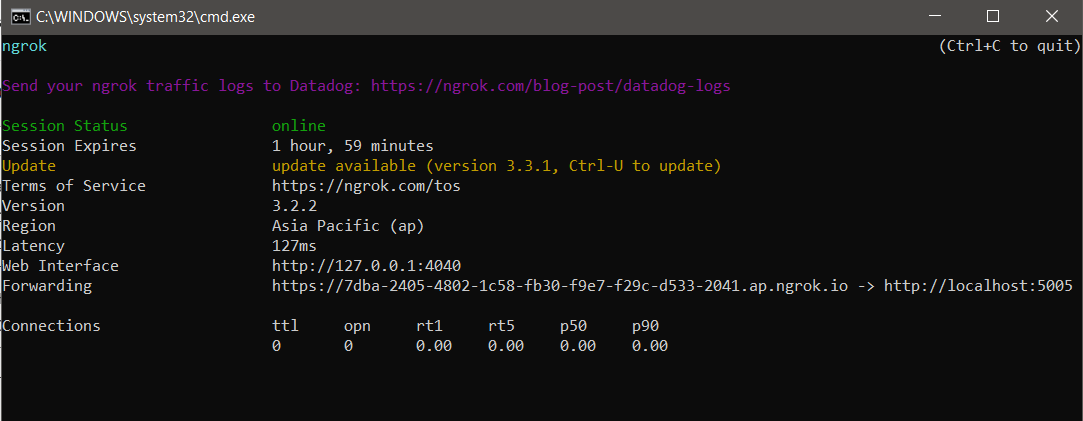


Hình . Giao diện Webhooks

Bước 8: Lấy mã (token) của ứng dụng trong bước 7 và dán vào page-access-token trong file credentials.yml.

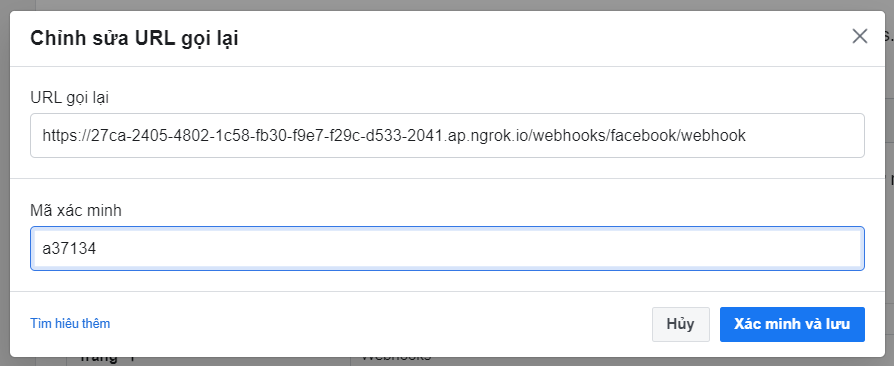
Bước 9: Chọn chỉnh sửa URL gọi lại ở giao diện Webhooks.

Bước 10: Khi chạy ngrok ở bước 3 ta sẽ được:



Hình . Khởi tạo ngrok

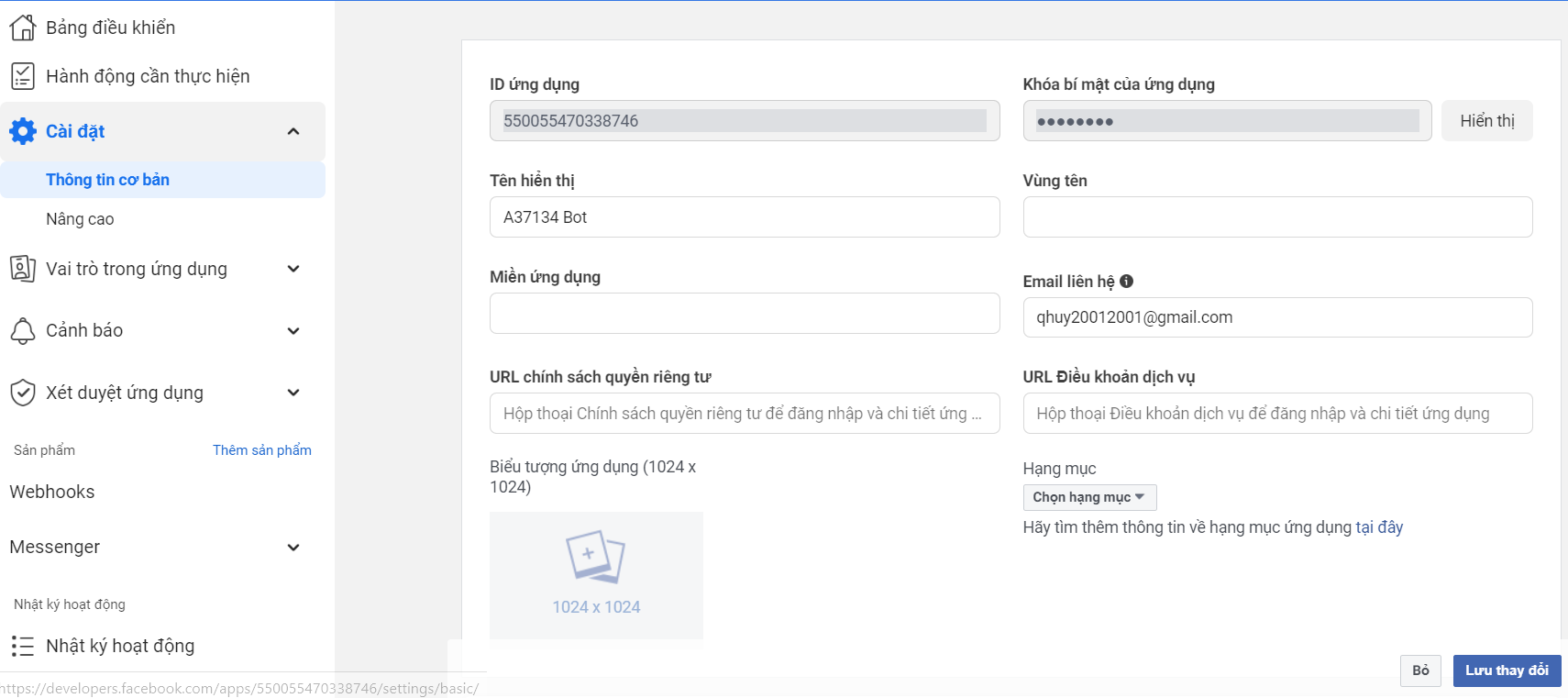
Dán đường link ở Forwarding vào giao diện chỉnh sửa URL mục URL gọi lại và thêm /webhooks/facebook/webhook vào cuối như sau:



Hình . Giao diện chỉnh sửa URL gọi lại

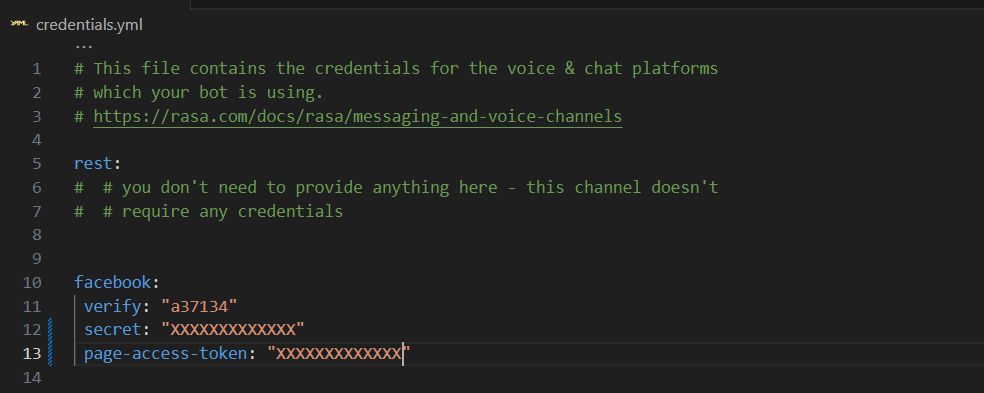
Mã xác minh sẽ được lấy từ verify trong file credentials.yml sẽ do chúng ta tự định nghĩa. Chọn xác minh và lưu.

Bước 11: Tạo trang cài đặt chọn thông tin cơ bản lấy khoá bí mật của ứng dụng sau đó dán vào secret trong file credentials.yml.



Hình . Giao diện cài đặt thông tin cơ bản

Bước 12: Sau khi làm xong ta sẽ có file credentials.yml như sau:

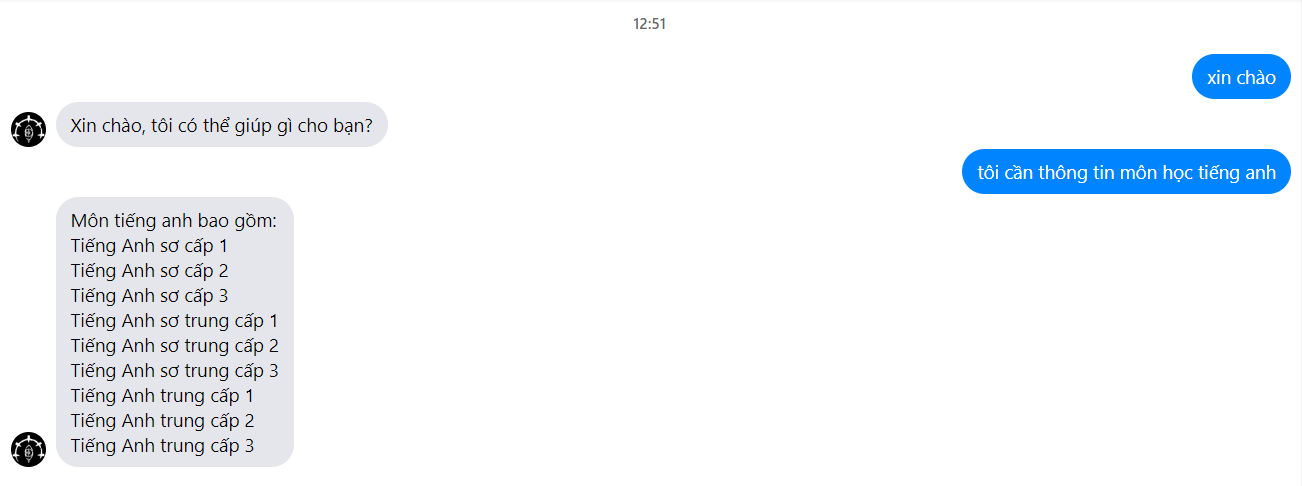


Hình . File credentials.yml

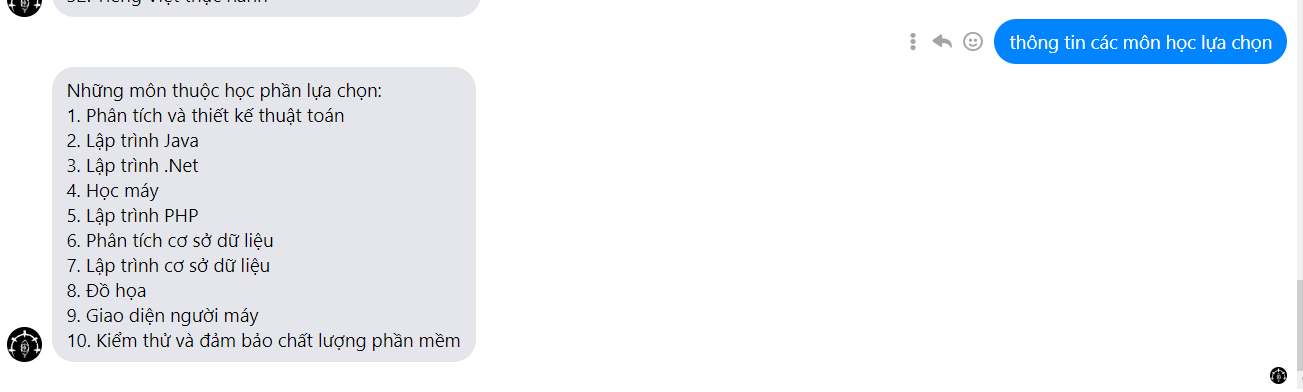
Bước 13: Truy cập vào Page bằng tài khoản Facebook cá nhân sau đó nhắn tin để thử nghiệm.

## Thực nghiệm

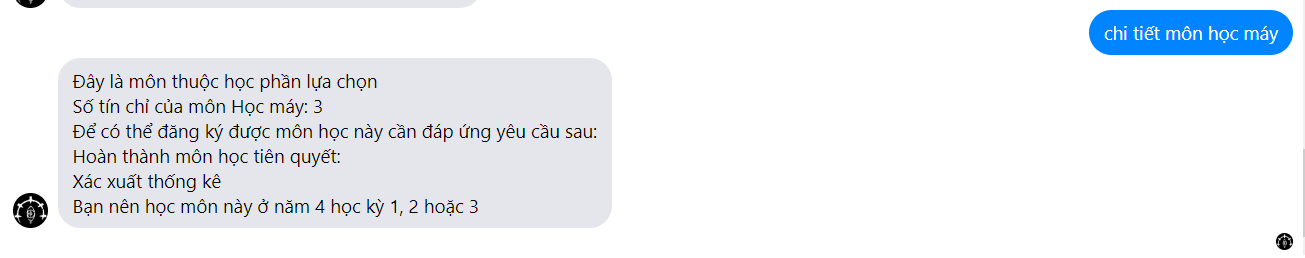
### Yêu cầu thông tin môn học



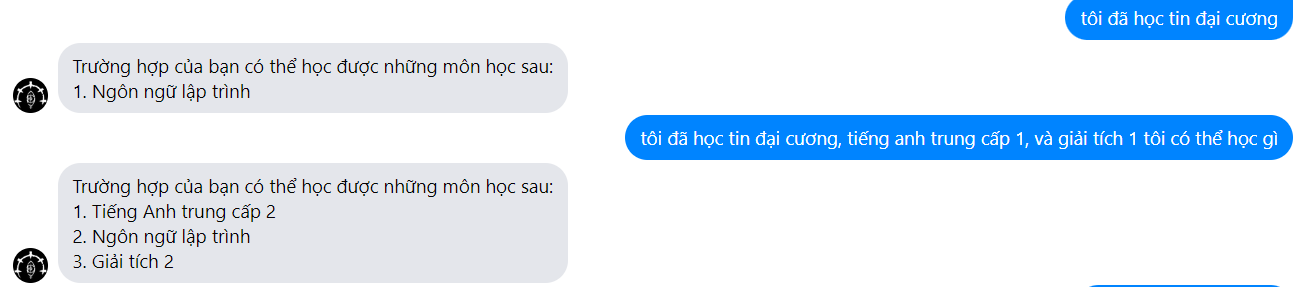
Hình . Liệt kê các môn học có chung tên



Hình . Liệt kê các môn thuộc một học phần

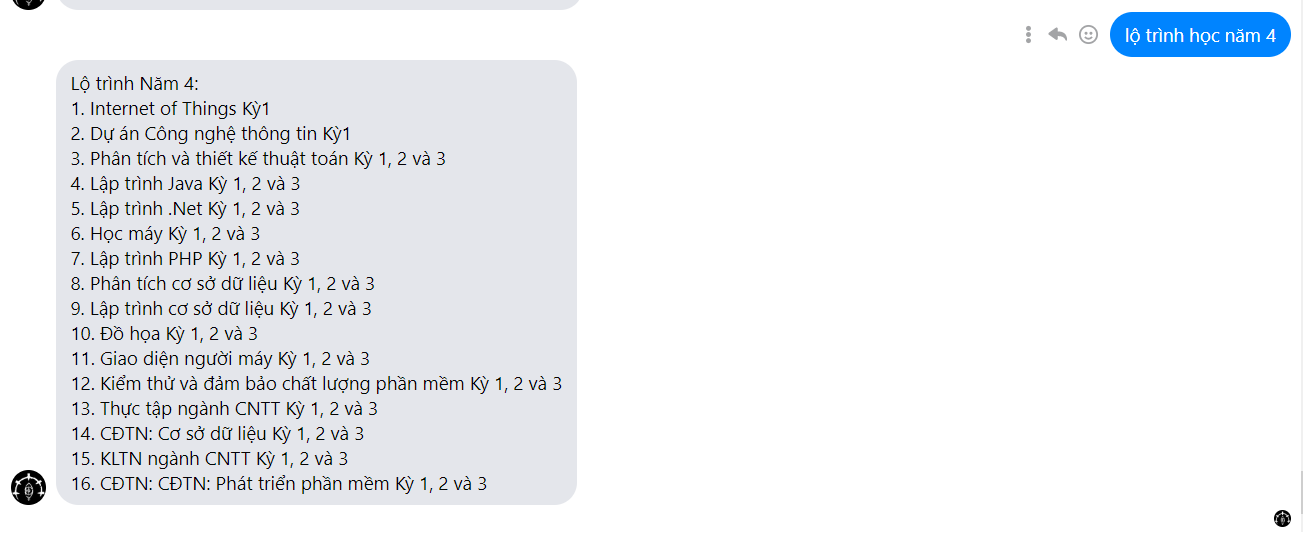


Hình . Chi tiết một môn học

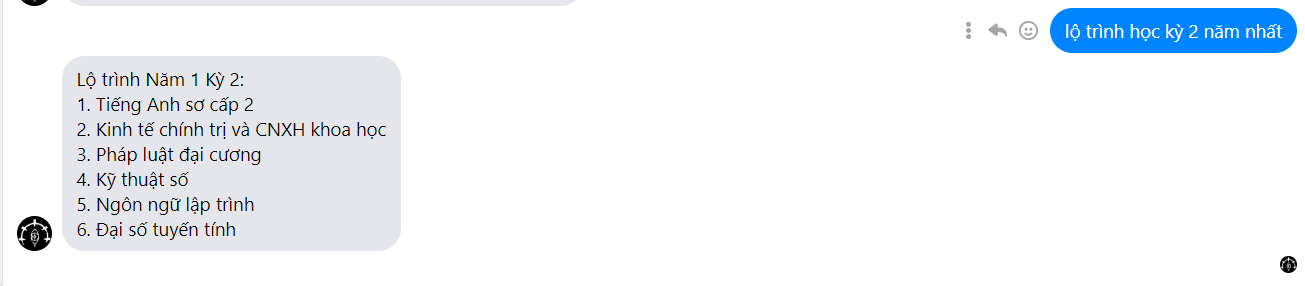


Hình . Yêu cầu các môn có thể học

### Yêu cầu lộ trình học



Hình . Lộ trình học của năm



Hình . Lộ trình học của kỳ và năm

### Hỏi giải đáp một số vấn đề

# KẾT LUẬN

Trên đây là báo cáo Chuyên đề tốt nghiệp với đề tài “Phần mềm quản lý Chương trình đào tạo và tư vấn hoàn thành chương trình đào tạo” tại trường Đại học Thăng Long. Đề tài này là một chủ đề rất hay, thực tế và mang tính ứng dụng cao trong thực tiễn. Để xây dựng phần mềm chúng em đã có cơ hội tìm hiểu về việc phân tích bài toán, phân tích nghiệp vụ cũng như nghiên cứu trau dồi thêm kiến thức về công nghệ, phát triển thuật toán để tối ưu hóa hệ thống, giúp hệ thống vận hành thực tiễn và trơn tru nhất. Dù còn nhiều thiết sót nhưng nhóm em cũng mong dự án có thể cung cấp một cái nhìn tổng quát về phần mềm quản lý CTĐT và tư vấn xét hoàn thành CTĐT.

Thông qua quá trình thực hiện đề tài, chúng em đã biết cách xây dựng một phần mềm theo thứ tự từ đầu đến cuối, trong đó bước đầu là phân tích nghiệp vụ bài toán, bước tiếp theo là đặc tả, phân tích luồng sự kiện, thiết kế cơ sở dữ liệu và thiết kế API, thiết kế và cài đặt chức năng. Ngoài ra, chúng em còn rèn luyện được cho mình rất nhiều kiến thức và các kỹ năng như:

* Hiểu được quy trình nghiệp vụ phù hợp cho quản lý CTĐT và tư vấn xét hoàn thành CTĐT tại trường Đại học Thăng Long.
* Hiểu hơn về phương pháp hướng đối tượng, từ đó áp dụng để phân tích, thiết kế bài toàn.
* Có được các kinh nghiệm thực tế khi được tham gia vào một dự án lớn cụ thể để có thế áp dụng được các kiến thức đã được học vào thực tiễn.
* Kỹ năng làm việc nhóm, quản lý công việc và làm báo cáo.

Thêm vào đó, chúng em cũng tích lũy thêm kiến thức về ngôn ngữ JS, C# hay thư viện như ReactJS, một số ngôn ngữ khác: HTML, CSS, … và thành thạo hơn trong việc sử dụng các công cụ quản lý mã nguồn để có một trang web gần như hoàn chỉnh và cũng là những kiến thức quan trọng để em có thể tiến xa hơn trong tương lai.

Hệ thống của nhóm em mới được cài đặt ở mức đơn giản để mô tả cho các phân tích của bài toán. Để đưa phần mềm vào thực hiện cần có thêm các nghiên cứu công nghệ sử dụng và cách tích hợp dữ liệu vào cho hệ thống như: nhập thông tin bằng file dữ liệu, đưa ra các báo cáo thống kê dưới dạng văn bản, tăng cường khả năng bảo mật cho trang web.

Do khối lượng công việc và thiếu sót về kiến thức nên sản phẩm của nhóm chúng em chưa được hoàn thiện nhất và không tránh khỏi những sai sót không đáng có. Chúng em rất mong nhận được những ý kiến, đóng góp của quý thầy cô để sản phẩm được hoàn thiện hơn.

Hướng phát triển đề tài:

* Nghiên cứu và phát triển kiến thức về các công nghệ đang sử dụng, các công nghệ mới nhằm phục vụ cho việc tối ưu hoá và phát triển hệ thống một cách hoàn chỉnh hơn nữa.
* Nghiên cứu và xây dựng trên nền tảng mobile.
* Áp dụng hệ thống vào sử dụng trong thực tiễn tại các trường đại học.

# TÀI LIỆU THAM KHẢO

[1] Tham khảo cuốn “Hướng dẫn học tập” trường Đại học Thăng Long (2019-2020)

[2] “SQL” <https://en.wikipedia.org/wiki/SQL/>

[3] “Microsoft SQL Server” <https://en.wikipedia.org/wiki/Microsoft_SQL_Server/>

[4] “Ngrok” <https://ngrok.com/product>

[5] “Rasa” <https://rasa.com/docs/rasa/>

[6] “Rasa NLU” <https://rasa.com/docs/rasa/nlu/>

[7] “Rasa Core” <https://rasa.com/docs/rasa/core/>

[9] “Facebook Developer” <https://developers.facebook.com/>

[10] “Machine Learning” <https://vi.wikipedia.org/wiki/H%E1%BB%8Dc_m%C3%A1y>

[11] “Học có giám sát” <https://vi.wikipedia.org/wiki/H%E1%BB%8Dc_c%C3%B3_gi%C3%A1m_s%C3%A1t>

[12] “Học không có giám sát” <https://vi.wikipedia.org/wiki/H%E1%BB%8Dc_kh%C3%B4ng_c%C3%B3_gi%C3%A1m_s%C3%A1t>

[13] “Học tăng cường” <https://vi.wikipedia.org/wiki/H%E1%BB%8Dc_t%C4%83ng_c%C6%B0%E1%BB%9Dng>

[14] “Neural Network” <https://itnavi.com.vn/blog/neural-network-la-gi>

[15] “Recurrent neural network” <https://vi.wikipedia.org/wiki/M%E1%BA%A1ng_th%E1%BA%A7n_kinh_h%E1%BB%93i_quy>

[16] “Phân loại RNN” <https://nttuan8.com/bai-13-recurrent-neural-network/>

[17] “Pipelines và Policies” <https://learning.rasa.com/conversational-ai-with-rasa/pipeline/>