**BỘ GIÁO DỤC VÀ ĐÀO TẠO**

**TRƯỜNG ĐẠI HỌC CẦN THƠ**

**KHOA CÔNG NGHỆ THÔNG TIN & TRUYỀN THÔNG**

****

**LUẬN VĂN TỐT NGHIỆP ĐẠI HỌC**

**NGÀNH CÔNG NGHỆ THÔNG TIN**

**Đề tài**

**WEBSITE ĐẶT PHÒNG TRỰC TUYẾN CHO KHÁCH DU LỊCH**

**TÍCH HỢP CHATBOT**

**Sinh viên: Dương Hồ Dũ**

**Mã số: B1706570**

**Khóa: K43**

**Cần Thơ, 04/2021BỘ GIÁO DỤC VÀ ĐÀO TẠO**

**TRƯỜNG ĐẠI HỌC CẦN THƠ**

**KHOA CÔNG NGHỆ THÔNG TIN & TRUYỀN THÔNG**

****

**LUẬN VĂN TỐT NGHIỆP ĐẠI HỌC**

**NGÀNH CÔNG NGHỆ THÔNG TIN**

**Đề tài**

**WEBSITE ĐẶT PHÒNG TRỰC TUYẾN CHO KHÁCH DU LỊCH**

**TÍCH HỢP CHATBOT**

**Giáo viên hướng dẫn: Sinh viên: Dương Hồ Dũ**

**TS. Lâm Nhựt Khang Mã số: B1706570**

**Ths. Bùi Đăng Hà Phương Khóa: K43**

**Cần Thơ, 04/2021**

**NHẬN XÉT CỦA GIÁO VIÊN HƯỚNG DẪN**

Cần Thơ, ngày …tháng … năm …

Giáo viên hướng dẫn

**LỜI CAM ĐOAN**

.

Cần Thơ, ngày ….. tháng ……… năm 2021

**LỜI CẢM ƠN**

Cần Thơ, ngày ….. tháng …… năm 2021

Sinh viên thực hiện

# MỤC LỤC

[MỤC LỤC iv](#_Toc80302112)

[DANH MỤC HÌNH ẢNH vi](#_Toc80302113)

[DANH MỤC BIỂU BẢNG vii](#_Toc80302114)

[DANH MỤC TỪ VIẾT TẮT viii](#_Toc80302115)

[TÓM TẮT ix](#_Toc80302116)

[ABSTRACT x](#_Toc80302117)

[CHƯƠNG 1. GIỚI THIỆU TỔNG QUAN 1](#_Toc80302118)

[1.1. Đặt vấn đề 1](#_Toc80302119)

[1.2. Nghiên cứu liên quan 2](#_Toc80302120)

[1.3. Website và Ứng dụng 3](#_Toc80302121)

[1.4. Chatbot 3](#_Toc80302122)

[1.5. Phân loại bình luận 3](#_Toc80302123)

[1.6. Mục tiêu đề tài 3](#_Toc80302124)

[1.7. Đối tượng và phạm vi nghiên cứu 3](#_Toc80302125)

[1.8. Phương pháp nghiên cứu 4](#_Toc80302126)

[1.9. Nội dung nghiên cứu 4](#_Toc80302127)

[1.10. Bố cục 4](#_Toc80302128)

[CHƯƠNG 2. XÂY DỰNG HỆ THỐNG 5](#_Toc80302129)

[2.1. Thư viện Reactjs 5](#_Toc80302130)

[2.2. Bootstrap 5](#_Toc80302131)

[2.3. Ngôn ngữ lập trình PHP 6](#_Toc80302132)

[2.4. Hệ quản trị cơ sở dữ liệu MYSQL 6](#_Toc80302133)

[2.5. RESTful API 7](#_Toc80302134)

[2.6. Lý thuyết liên quan Service AI 8](#_Toc80302135)

[2.6.1. Ngôn ngữ lập trình Python 8](#_Toc80302136)

[2.6.2. \*\*Sau khi tìm hiểu em sẽ cập nhật sau 8](#_Toc80302137)

[2.7. Mô tả hệ thống 9](#_Toc80302138)

[2.8. Yêu cầu chức năng 9](#_Toc80302139)

[2.8.1. Các chức năng chính 9](#_Toc80302140)

[2.8.2. Chức năng dành cho Admin 10](#_Toc80302141)

[2.8.3. Chức năng dành cho Host 11](#_Toc80302142)

[2.8.4. Chức năng dành cho member 11](#_Toc80302143)

[2.8.5. Các chức năng khác 12](#_Toc80302144)

[2.8.6. Thanh toán VNPAY 12](#_Toc80302145)

[2.8.7. Tích hợp Google map 12](#_Toc80302146)

[2.9. Thiết kế thành phần dữ liệu 12](#_Toc80302147)

[2.10. Sơ đồ Usecase 12](#_Toc80302148)

[2.11. Mô hình dữ liệu mức quan niệm CDM 13](#_Toc80302149)

[2.12. Mô hình dữ liệu mức vật lý PDM 13](#_Toc80302150)

[2.13. Sơ đồ chức năng 19](#_Toc80302151)

[CHƯƠNG 3. HỆ THỐNG HỖ TRỢ 20](#_Toc80302152)

[3.1. Chatbot 20](#_Toc80302153)

[3.1.1. Tổng quan về chatbot 20](#_Toc80302154)

[3.1.2. Framework Rasa 20](#_Toc80302155)

[3.1.3. Xây dựng bộ dữ liệu NLU 20](#_Toc80302156)

[3.1.4. Xây dựng các câu trả lời liên quan 20](#_Toc80302157)

[3.1.5. Xây dựng dữ liệu đối thoại 20](#_Toc80302158)

[3.1.6. Huấn luyện mô hình 20](#_Toc80302159)

[3.2. Hệ thống phân loại bình luận 20](#_Toc80302160)

[3.2.1. Tổng quan về bài toán phân lớp 20](#_Toc80302161)

[3.2.2. Ứng dụng bài toán vào đề tài 20](#_Toc80302162)

[3.2.3. Đánh giá kết quả thực nghiệm 20](#_Toc80302163)

[3.2.4. Giới thiệu về LSTM hoặc SVM 20](#_Toc80302164)

[3.3. Tích hợp 20](#_Toc80302165)

[3.4. Thảo luận kết quả đạt được 20](#_Toc80302166)

[CHƯƠNG 4. GIỚI THIỆU CHƯƠNG TRÌNH 21](#_Toc80302167)

[4.1. Giao diện website 21](#_Toc80302168)

[4.2. Giao diện ứng dụng 21](#_Toc80302169)

[4.3. Chatbot 21](#_Toc80302170)

[4.4. Hệ thống phân loại bình luận 21](#_Toc80302171)

[4.5. Hệ thống tích hợp (//các em đổi tên sau) 21](#_Toc80302172)

[CHƯƠNG 5. KẾT LUẬN 22](#_Toc80302173)

[5.1. Kết luận 22](#_Toc80302174)

[5.2. Hướng pháp triển 22](#_Toc80302175)

[TÀI LIỆU THAM KHẢO 23](#_Toc80302176)

[PHỤ LỤC 1: MÔ HÌNH DỮ LIỆU MỨC VẬT LÝ 24](#_Toc80302177)

[PHỤ LỤC 2: BẢNG MÔ TẢ PHƯƠNG THỨC 25](#_Toc80302178)

[PHỤ LỤC 3: CÁC RÀNG BUỘC THAM CHIẾU 26](#_Toc80302179)

[PHỤ LỤC 4: LƯỢC ĐỒ CƠ SỞ DỮ LIỆU 27](#_Toc80302180)

[PHỤ LỤC 5: HÌNH ẢNH CHƯƠNG TRÌNH 28](#_Toc80302181)

# DANH MỤC HÌNH ẢNH

[Hình 1.1 Website trivago 2](#_Toc86496588)

[Hình 1.2 Website Luxstay 2](#_Toc86496589)

[Hình 2.1 RESTful API 7](#_Toc86496590)

[Hình 2.2 Hình ảnh Rasa 8](#_Toc86496591)

[Hình 2.3 Mô hình Client-Server 9](#_Toc86496592)

[Hình 2.4 Mô hình thực thể mức quan niệm 14](#_Toc86496593)

[Hình 2.5 Sơ đồ chức năng 22](#_Toc86496594)

[Hình 3.1 Minh họa về mạng ANN [8] 25](#_Toc86496595)

[Hình 3.2 Sơ đồ luồng dữ liệu của Rasa [10] 26](#_Toc86496596)

[Hình 3.3 Ý định “ask\_about\_website” 27](#_Toc86496597)

[Hình 3.4 Ý định “find\_room\_with\_convenient” 27](#_Toc86496598)

# DANH MỤC BIỂU BẢNG

[Bảng 2.1 Thực thể MEMBER 14](#_Toc86525467)

[Bảng 2.2 Thực thể HOST 15](#_Toc86525468)

[Bảng 2.3 Thực thể PROPOSAL 16](#_Toc86525469)

[Bảng 2.4 Thực thể CONVENIENT 16](#_Toc86525470)

[Bảng 2.5 Thực thể ROOMS 17](#_Toc86525471)

[Bảng 2.6 Thực thể BOOKING\_SCHEDULE 18](#_Toc86525472)

[Bảng 2.7 Thực thể TRANSACTION 19](#_Toc86525473)

[Bảng 2.8 Thực thể COMMENT 20](#_Toc86525474)

[Bảng 2.9 Thực thể EVALUATE 20](#_Toc86525475)

[Bảng 2.10 Thực thể EVALUATE 21](#_Toc86525476)

[Bảng 3.1 Bảng thống kê các ý định trong đề tài 29](#_Toc86525477)

[Bảng 3.2 Bảng thống kê các hành động trong đề tài 33](#_Toc86525478)

# DANH MỤC TỪ VIẾT TẮT

//sắp theo thứ tự ABC

| **Từ viết tắt** | **Mô tả** |
| --- | --- |
| AI | Artificial Intelligence |
| API | Application Programming Interface |
| CSS | Cascading Style Sheets |
| CSDL | Cở sở dữ liệu |
| DOM | Document Object Model |
| DEV | Developer |
| ML | Machine Learning |
| NLU | Nature Language Understanding |
| NLP | Natural Language Processing |
| UI | User Interface |
| PHP | Hypertext Pre-processor |

# TÓM TẮT

# ABSTRACT

# GIỚI THIỆU TỔNG QUAN

Giới thiệu sơ lực về đề tài nghiên cứu, các thành phần của đề tài, nội dung và phương pháp nghiên cứu.

## Đặt vấn đề

Ngành nghề du lịch tại Việt Nam hiện nay đang phát triển rất mạnh mẽ, với khi hậu nhiệt đới ôn hòa Việt Nam là đất nước rất thuận lợi để phát triển ngành du lịch. Cụ thể nhiều địa điểm du lịch nổi tiếng trong nước được biết đến như danh lam thắng cảnh Vịnh Hạ Long – Ninh Bình, Cố Đô Huế, bãi biển Nha Trang – Khánh Hòa hay thành phố mộng mơ mát mẻ tọa lạc tại tỉnh Lâm Đồng – Đà Lạt, đảo Phú Quốc – Kiên Giang hay tại trung tâm miền Nam – Thành phố Hồ Chí Minh cũng có nhiều nơi tham quan nổi tiếng tuyệt đẹp và ghi nhiều dấu ấn lịch sử đáng chú ý như Nhà thờ Đức Bà tại trung tâm quận 1, … Do đó ngành công nghiệp du lịch đã đóng góp rất nhiều vào tổng doanh thu GDP hằng năm của nước ta.

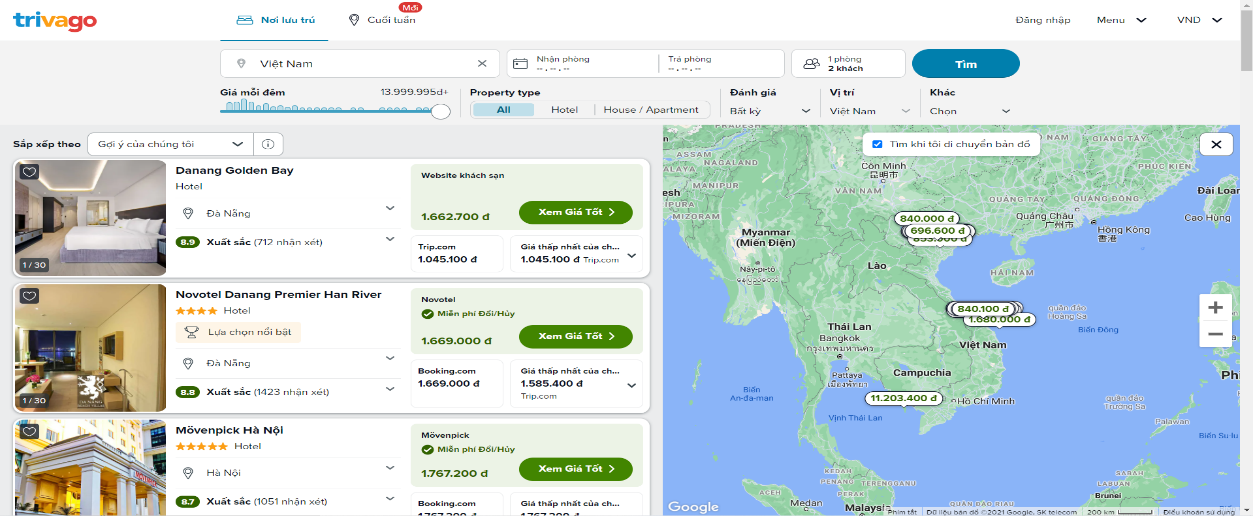
Ngành du lịch phát triển kéo theo thu nhập của hộ dân tại các địa điểm du lịch cũng phát triển theo, mức sống cao hơn từ đó nhiều bất động sản như nhà nghỉ, khách sạn, villa, resort hay mới mẻ hơn gần đây là mini house, … được xây dựng để cung cấp nhu cầu du lịch của người dân trong nước và bạn bè Quốc tế khi ghé thăm Việt Nam.

Những nhu cầu tăng cao của ngành du lịch đã đặt ra 2 bài toán chính, một là có rất nhiều khách sạn, homestay mọc lên vậy làm sao người đi du lịch có thể tìm chọn nơi ở phù hợp với mình; hai là làm sao những người chủ khách sạn, homestay có thể giới thiệu dịch vụ của mình đến tay người sử dụng, nhất là những người ở xa hoặc khách nước ngoài. Để trả lời cho 2 câu hỏi trên thì chúng ta cần phải có một hệ thống mà trên đó người dùng có thể lựa chọn nơi ở hợp lý, thuận tiện, phù hợp với mình mỗi khi đi du lịch; chủ doanh nghiệp thông qua hệ thống có thể quảng cáo dịch vụ của mình, quản lý các thông số của doanh nghiệp mình.

Với những vấn đề trên thì website đặt phòng trực tuyến cho khách du lịch đã ra đời. Nơi mọi người có thể truy cập để biết được thông tin nhiều khách sạn, homestay, v.v… tại nơi họ muốn đi du lịch. Với những tính năng rất thiết thực và nhanh chóng, người dùng có thể xem thông tin nơi ở, hình ảnh, giá phòng cụ thể và liên lạc với host để đặt phòng thông qua trang hệ thống. Ngoài ra hệ thống còn giới thiệu nhiều địa điểm nổi tiếng tại các nơi phát triển ngành du lịch đến với dùng sử dụng trang web với tinh thần tiết kiệm thời gian, công sức cho người thích khám phá địa điểm du lịch. Mang đến trãi nghiệm thú vị, thao tác dễ dàng và nhanh chóng cho người dùng.

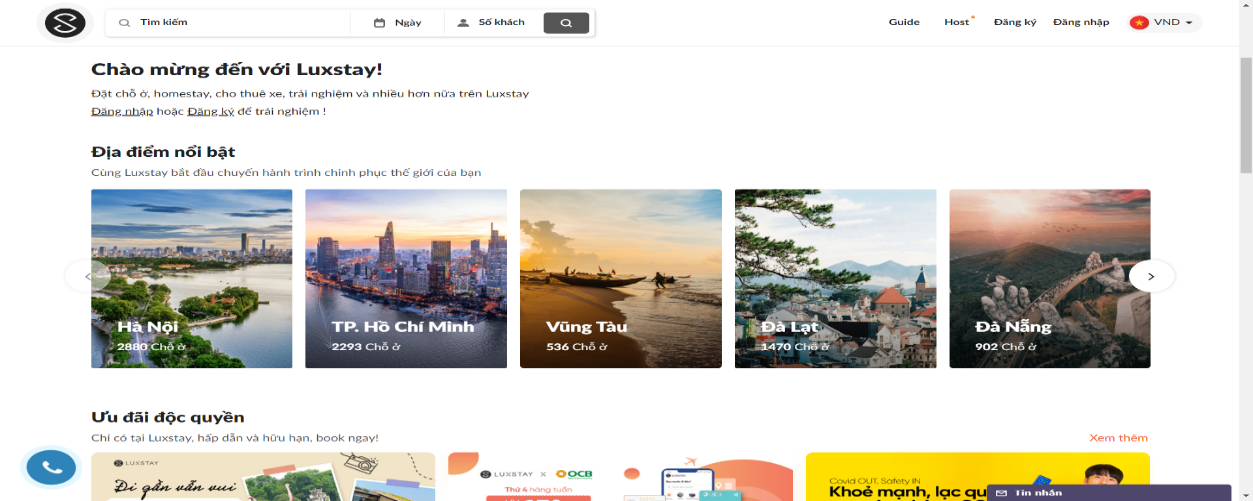
## Nghiên cứu liên quan

Trivago là một ông lớn trong trong việc tìm kiếm, đặt phòng du lịch trên toàn thế giới. Theo thống kê Trivago có khoản 90 triệu lượt truy cập từ 52 quốc gia. Việc sử dụng các trang web như thế này giúp người dùng có thể dễ dàng chọn được nơi nghỉ du lịch phù hợp với nhu cầu và khả năng của mình, nó cũng giúp những người chủ doanh nghiệp dễ dàng quảng bá dịch vụ của mình đến người dùng hơn.



Hình . Website trivago

Một website khác cũng hoạt động trong lĩnh vực này đó chính là Luxstay, một niềm tự hào của Việt Nam về việc phát triển hệ thống hoạt động trên nền tảng “home-sharing”. Không chỉ dừng lại ở cho thuê nhà [homestay](https://homeaz.vn/homestay-tag4052/), Luxstay còn cung cấp các dịch vụ có liên quan đến hoạt động bất động sản, cho phép các chủ nhà kinh doanh thông qua căn hộ của mình, đồng thời mang tới chất lượng căn hộ ngắn hạn tốt nhất cho những người thuê nhà. Hiện tại, Luxstay đã “có mặt” ở TP.HCM, Hà Nội và các thành phố lớn trên khắp cả nước.



**Hình 1.2 Website Luxstay**

## Chatbot

Ngày nay việc sử dụng trợ lý ảo – chatbot là một việc không còn khó khăn. Những chatbot này có thể giúp chúng ta trả lời được những câu hỏi cơ bản từ phía khách hàng, giúp tiết kiệm thời gian cho người vận hành hệ thống cũng như tăng trải nghiệm cho khách hàng. Những chatbot nổi tiếng hiện nay có thể kể đến như Siri của Apple, Contana của Windows, Google Assistant của Google, …

## Phân loại bình luận

Phân loại bình luận cũng là một vấn đề được quan tâm hiện nay. Bình luận trên các trang web xã hội hiện nay rất phổ biến, người dùng có thể viết những câu cảm nhận của mình về dịch vụ, sản phẩm mà họ được trải nghiệm. Những câu bình luận đó cũng chính là thứ mà người cung cấp dịch vụ cần nắm bắt vì đó là tâm lý khách hàng. Tuy nhiên việc đọc hết tất cả các bình luận là một việc gần như bất khả thi, do đó hệ thống phân loại bình luận ra đời để giúp người cung cấp dịch vụ nắm bắt rõ hơn về nhu cầu và tâm lý của khách hàng.

## Mục tiêu đề tài

Xây dựng website mà trên đó người dùng có thể xem, đánh giá, đặt phòng du lịch trực tuyến, thanh toán trực tuyến. Ngoài ra website còn hỗ trợ các chủ doanh nghiệp (khách sạn, homestay, …) trong việc cho thuê phòng, xác nhận đặt phòng online, quản lý doanh thu của mình. Bên cạnh đó website còn tích hợp thêm một chatbot có thể trả lời các câu hỏi liên quan tới hệ thống.

## Đối tượng và phạm vi nghiên cứu

Hệ thống có 4 đối tượng chính:

+ Người quản trị website (Admin)

+ Chủ các doanh nghiệp (Host)

+ Người dùng thông thường (Guest)

+ Người dùng thành viên (Member)

Các vấn đề nghiên cứu:

+ Xây dựng website để phục vụ cho 4 nhóm người dùng ở trên.

+ Tìm hiểu về chatbot, các hệ thống gợi ý hiện có.

+ Tìm hiểu về cách vận hành của một single-page application.

+ Cách tích hợp chatbot vào website.

## Phương pháp nghiên cứu

Tham gia trao đổi, tham khảo ý kiến với giao viên hoặc những người đang thực hiện các dự án công nghệ thông tin về cách vận hành, kiến trúc, ngôn ngữ lập trình xu hướng để xây dựng một web-application hiện nay. Tìm kiếm các tài liệu liên quan đến chatbot: tạo chatbot, huấn luyện chatbot và cách tích hợp chatbot vào website.

## Nội dung nghiên cứu

Tìm hiểu thư viện Reactjs để xây dựng front-end cho trang web.

Tìm hiểu về ngôn ngữ lập trình PHP xây dựng back-end cho trang web.

Tìm hiểu về hệ quản trị cơ sở dữ liệu Mysql cho việc lưu trữ và truy vấn dữ liệu.

Tìm hiểu về XAMPP để tạo máy chủ ảo phục vụ cho việc xây dựng và kiểm thử trang web.

Tìm hiểu về ngôn ngữ lập trình Python, các package, thư viện cần thiết để xây dựng chatbot.

## Bố cục

Nội dung của quyển luận văn bao gồm 05 chương:

* Chương 1- Giới thiệu tổng quan: giới thiệu sơ lược về đề tài nghiên cứu và mục tiêu của đề tài.
* Chương 2 – Xây dựng hệ thống: giới thiệu về cơ sở lý thuyết của các thành phần cấu tạo nên hệ thống và các chức năng chính của hệ thống.
* Chương 3- Hệ thống hỗ trợ: giới thiệu về chatbot và cách ứng dụng của Rasa chatbot vào website.
* Chương 4- Giới thiệu chương trình: thành quả của đề tài, giao diện và chức năng của đề tài.
* Chương 5- Kết luận: tổng kết kết quả đạt được của nghiên cứu và đề xuất hướng phát triển cho đề tài.

# XÂY DỰNG HỆ THỐNG

Giới thiệu cơ sở lý thuyết về các thành phần cấu tạo nên hệ thống, yêu cầu và các chức năng của hệ thống

## Thư viện Reactjs

Reactjs là một thư viện của JavaScript, dùng để tạo User Interface (UI – giao diện người dùng) được phát hành bởi facebook. Reactjs được phát triển để giúp người lập trình có thể tạo ra UI cho website một cách dễ dàng và linh động. Reactjs sử dụng cấu trúc của JSX để chia các thành phần của website thành các component nhỏ hơn, điều này sẽ giúp component được tái sử dụng một cách dễ dàng ở một trang khác. Ngoài ra, việc định nghĩa state để lưu trữ dữ liệu trong mỗi component cũng sẽ khiến nội dung của component trở nên linh hoạt hơn, nếu state bị thay đổi thì nội dung của component đó cũng sẽ thay đổi theo. Mỗi khi một dữ liệu bị thay đổi, thay vì phải load lại cả trang thì Reactjs chỉ cần load lại những component chứa dữ liệu bị thay đổi, vì thế sẽ giúp trang web trở nên mượt mà hơn, nhanh hơn. Một điểm nổi bật khác là Reactjs mang lại hiệu suất tốt hơn với virtual DOM (Virtual Document Object Model – Mô hình đối tượng tài liệu), cho phép người dùng xây dựng các virtual DOMs và lưu chúng lại trong bộ nhớ. Khi component cần render lại thì virtual DOM sẽ được so sánh với DOM thật trong bộ nhớ để so sánh sự khác biệt từ đó thay thế chúng trong DOM thật. [1]

## Bootstrap

Bootstrap là một CSS (Cascading Style Sheets) frame-work mã nguồn mở được phát triển bởi Mark Otto và Jacob Thornton tại Twitter giúp lập trình viên thiết kế và tùy chỉnh cách bố trí cho trang web một cách dễ dàng. Ngày nay bootstrap được sử dụng rộng rãi trong việc sắp xếp bố cục và trang trí trang web, và gần như nó đã trở thành tiêu chuẩn của một lập trình viên front-end.

Bootstrap bao gồm các mẫu thiết kế dựa trên HTML và CSS như typography, forms, buttons, tables, navigation, modals, image carousels, … cũng như các plugin JavaScript tùy chọn. Bootstrap cũng cung cấp cho bạn khả năng tạo ra các responsive designs một cách dễ dàng. Responsive web design tức là website sẽ tự động điều chỉnh kích thước để hiển thị vừa vặn trên tất cả các thiết bị, từ điện thoại nhỏ đến desktops lớn. [2]

## Ngôn ngữ lập trình PHP

PHP (Personal Home Page hay Hypertext Pre-processor) là một loại ngôn ngữ lập trình kịch bản hay một loại mã lệnh chủ yến được dùng để viết các kịch bản cho máy chủ. Ngôn ngữ PHP thông thường phù hợp với việc **lập trình web** vì nó có thể dễ dàng kết nối với các website có sử dụng HTML để chạy trên những trình duyệt web. Chính vì vậy, ngôn ngữ lập trình này luôn được coi là khá dễ học. Và PHP cũng trở thành ngôn ngữ web dev (Developer – người phát triển phần mềm) phổ biến nhất mà các lập trình viên sẽ cần phải học nếu muốn phát triển theo hướng web.

Ưu điểm của PHP là mã nguồn mở nên người dùng có thể dễ dàng cài đặt và sử dụng. PHP tương thích với nhiều ngôn ngữ, trình duyệt web, bảo mật cao thích hợp cho việc xử lý dữ liệu phí server. PHP trở nên phổ biến nhờ vào việc có thể sinh ra các tài liệu XML, hình ảnh, tập tin PDF, v.v… [3]

## Hệ quản trị cơ sở dữ liệu MYSQL

MySQL là [hệ quản trị cơ sở dữ liệu](https://vi.wikipedia.org/wiki/H%E1%BB%87_qu%E1%BA%A3n_tr%E1%BB%8B_c%C6%A1_s%E1%BB%9F_d%E1%BB%AF_li%E1%BB%87u) [mã nguồn mở](https://vi.wikipedia.org/wiki/Ph%E1%BA%A7n_m%E1%BB%81m_ngu%E1%BB%93n_m%E1%BB%9F) phổ biến nhất thế giới được phát hành bởi Oracle và được các lập trình viên rất ưa chuộng trong quá trình phát triển ứng dụng. Vì MySQL là hệ quản trị cơ sở dữ liệu tốc độ cao, ổn định và dễ sử dụng, có tính khả chuyển, hoạt động trên nhiều hệ điều hành cung cấp một hệ thống lớn các hàm tiện ích rất mạnh. Với tốc độ và tính bảo mật cao, MySQL rất thích hợp cho các ứng dụng có truy cập CSDL trên internet. Người dùng có thể tải về MySQL miễn phí từ trang chủ. MySQL có nhiều phiên bản cho các hệ điều hành khác nhau như Windows, Linux, MacOS X, …

MySQL là một trong những ví dụ rất cơ bản về Hệ Quản trị Cơ sở dữ liệu quan hệ sử dụng Ngôn ngữ truy vấn có cấu trúc (SQL).

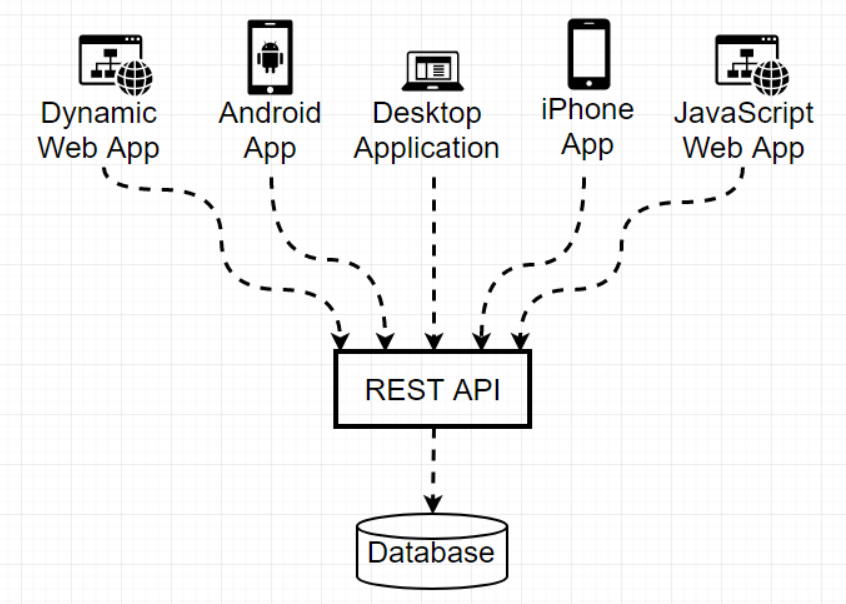
MySQL được sử dụng cho việc bổ trợ NodeJs, [PHP](https://vi.wikipedia.org/wiki/PHP), [Perl](https://vi.wikipedia.org/wiki/Perl), và nhiều ngôn ngữ khác, làm nơi lưu trữ những thông tin trên các trang web viết bằng NodeJs, PHP hay Perl, … [4] [5]

## RESTful API

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

REST (REpresentational State Transfer) là một kiểu kiến trúc được sử dụng trong việc giao tiếp giữa các máy tính (máy tính cá nhân và máy chủ của trang web) trong việc quản lý các tài nguyên trên internet. REST được sử dụng rất nhiều trong việc phát triển các ứng dụng Web Services sử dụng giao thức HTTP trong giao tiếp thông qua mạng internet. Các ứng dụng sử dụng kiến trúc REST này thì sẽ được gọi là ứng dụng phát triển theo kiểu RESTful.

API (Application Programming Interface) là tập hợp các quy tắc hay các cơ chế mà dựa vào đó các ứng dụng khác nền tảng có thể tương tác với nhau. API có thể trả về kiều dữ liệu tương ứng với ứng dụng yêu cầu như JSON hay XML. [6]



Hình . RESTful API

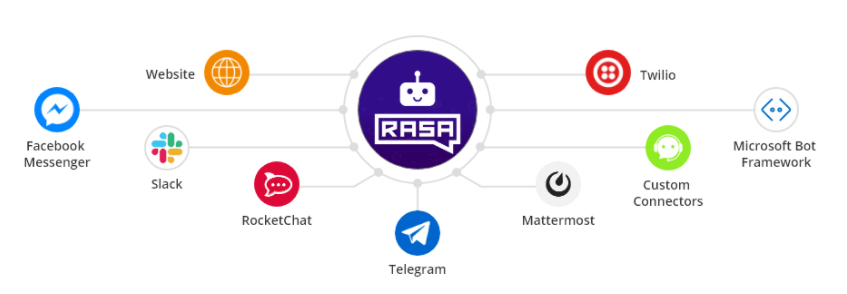
## Lý thuyết liên quan Service AI

### Ngôn ngữ lập trình Python

[Python](https://quantrimang.com/python) là ngôn ngữ lập trình hướng đối tượng, cấp cao, mạnh mẽ, được tạo ra bởi Guido van Rossum và được cho ra mắt lần đầu vào năm 1991. Nó dễ dàng để tìm hiểu và đang nổi lên như một trong những ngôn ngữ lập trình nhập môn tốt nhất cho người lần đầu tiếp xúc với ngôn ngữ lập trình. Python hoàn toàn tạo kiểu động và sử dụng cơ chế cấp phát bộ nhớ tự động. Python có [cấu trúc dữ liệu](https://quantrimang.com/cau-truc-du-lieu-va-giai-thuat) cấp cao mạnh mẽ và cách tiếp cận đơn giản nhưng hiệu quả đối với lập trình hướng đối tượng. Cú pháp lệnh của Python là điểm cộng vô cùng lớn vì sự rõ ràng, dễ hiểu và cách gõ linh động làm cho nó nhanh chóng trở thành một ngôn ngữ lý tưởng để viết script và phát triển ứng dụng trong nhiều lĩnh vực, ở hầu hết các nền tảng.

Điểm cộng của python là cú pháp đơn giản, gần gũi với ngôn ngữ tự nhiên. Vì thế nó rất được ưu chuộng bởi các lập trình viên hiện nay. Chính vì các điểm mạnh của ngôn ngữ lập trình này mà nó được ưu tiên lựa chọn để thực hiện các đự án AI.

### Rasa Framework



Hình . Hình ảnh Rasa

Rasa là một framework máy học mã nguồn mở được viết bằng ngôn ngữ python được dùng để trả lời các câu hỏi đơn giản của người dùng.

Rasa bao gồm Rasa NLU (Natural Language Understanding) dùng để hiểu ý định của người dùng và Rasa CORE dùng để nhận biết ý định của người dùng và đưa ra phản hồi tương ứng với ý định đó.

Rasa server sẽ nhận yêu cầu POST từ người dùng theo dạng json chứa tin nhắn text từ người dùng và client-id. Sau khi Rasa xử lý tin nhắn đó sẽ gửi trả về kết quả chứa tin nhắn trả lời hoặc các hình ảnh, link,… theo dạng json thông qua API.

Rasa thích hợp triển khai trên nhiều nền tảng như: Facebook messenger, Website, Telegram, Slack,…

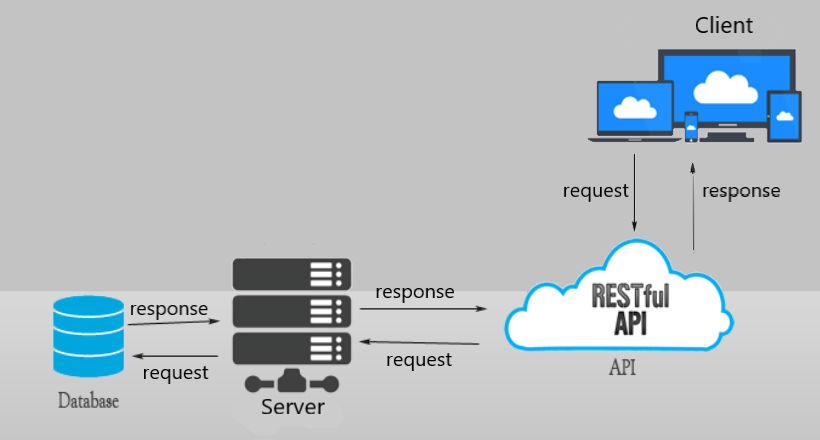
### MySQL Connector

MySQL Connector được sử dụng trong các trường hợp bot cần truy xuất các thông tin từ cở sở dữ liệu mysql.

## Mô tả hệ thống

Hệ thống được xây dựng theo mô hình Client-Server:

* Client là web browser có nhiệm vụ hiển thi giao diện người dùng, gửi các yêu cầu truy vấn, cập nhật, thêm, xóa dữ liệu lên cho server thông qua API. Nhận các dữ liệu được trả về, xử lý và hiển thị lên cho người dùng.
* Server có nhiệm vụ tiếp nhận yêu cầu từ phía client, xử lý và truy xuất dữ liệu trong CSDL, trả kết quả về cho client theo định dạng JSON.



Hình . Mô hình Client-Server

## Yêu cầu chức năng

### Các chức năng chính

**- Đăng ký thành viên**: Người dùng sẽ truy cập vào trang đăng ký thành viên và điền đầy đủ thông tin vào form đăng ký. Dữ liệu này sẽ được client kiểm tra tính hợp lệ và gửi lên server. Sau đó server sẽ thêm dữ liệu và trả về thông báo cho client. Nếu tài khoản được thêm thành công thì client sẽ thông báo là “Đăng ký tài khoản thành công!”, ngược lại thì client sẽ thông báo “Đăng ký tài khoản thất bại”. Ngay sau đó người dùng có thể sử dụng tài khoản mà mình đăng ký thành công để đăng nhập vào hệ thống.

**- Đăng ký trở thành host**: Người dùng sẽ truy cập vào trang đăng ký doanh nghiệp và điền đầy đủ thông tin vào form đăng ký. Dữ liệu này sẽ được client kiểm tra tính hợp lệ và gửi lên server. Sau đó server sẽ thêm dữ liệu và trả về thông báo cho client. Nếu tài khoản được thêm thành công thì client sẽ thông báo là “Đăng ký tài khoản thành công!”, ngược lại thì client sẽ thông báo “Đăng ký tài khoản thất bại”. Nếu đăng ký thành công thì người dùng sẽ phải chờ admin duyệt tài khoản và cấp mật khẩu cho mình thì mới có thể đăng nhập vào hệ thống.

**- Đăng nhập**: người dùng sẽ truy cập vào trang đăng nhập để, nhập tên đăng nhập và mật khẩu. Tên đăng của người dùng sẽ được gửi lên server và yêu cầu lấy mật khẩu của người dùng đó. Trong trường hợp không có kết quả trả về hoặc mật khẩu khác với mật khẩu người dùng nhập vào thì client sẽ thông báo “Bạn đã nhập sai tên đăng nhập hoặc mật khẩu!”. Ngược lại nếu người dùng đó là admin thì sẽ chuyển đến trang admin, nếu người dùng đó là host thì sẽ chuyển đến trang quản lý của host, nếu người dùng đó là member thì sẽ chuyển về trang chủ.

**- Đăng xuất**: đăng xuất tài khoản ra khỏi hệ thống, xóa tất cả cả dữ liệu của người dùng khỏi web browser.

### Chức năng dành cho Admin

Để sử dụng các chức năng của admin thì phải đăng nhập bằng tài khoản admin.

- **Xem được thống kê doanh thu của toàn bộ hệ thống**: khi admin sử dụng này thì server sẽ gửi về tất cả các dữ liệu về giao dịch của toàn bộ hệ thống. Client sẽ dựa vào dữ liệu được trả về để tính toán và hiển thị thống kê cho admin các chỉ số như: tổng doanh thu, tăng trưởng hoặc tụt giảm so với tháng trước, tổng số giao dịch của hệ thống trong tháng, …

- **Quản lý người dùng**: khi admin sử dụng chức năng này thì server sẽ gửi về tất cả các thông tin của host cũng như của member. Client sẽ dựa vào các dữ liệu được trả về để hiển thị lên cho admin. Admin có thể xem, khóa tài khoản hoặc xóa các tài khoản hiện đang có trong hệ thống.

- **Cập nhật các tiện tích**: khi admin sử dụng chức năng này thì server sẽ gửi về tất cả các tiện ích của toàn bộ hệ thống. Client sẽ dựa vào dữ liệu được trả về để hiển thị lên cho admin. Admin có thể thêm, sửa, xóa tên hoặc logo của các tiện ích của hệ thống.

- **Xử lý yêu cầu trở thành host**: khi admin sử dụng chức năng này thì server sẽ trả về tất cả các yêu cầu để trở thành host. Client sẽ dựa vào các dữ liệu được trả về để hiển thị lên cho admin. Admin có thể duyệt hoặc bỏ qua yêu cầu; nếu admin duyệt thì admin sẽ tiến hành cấp mật khẩu cho host, các dữ liệu này sẽ được gửi lên server thêm tài khoản host, khi thêm thành công thì server sẽ tự động gửi email tới host để thông báo yêu cầu được duyệt và cấp mật khẩu cho host đó; nếu admin bỏ qua yêu cầu thì server sẽ xóa yêu cầu khỏi CSDL.

- **Quản lý bình luận**: khi admin sử dụng chức năng này thì server sẽ trả về tất cả các bình luận của hệ thống sắp xếp theo ngày gửi mới nhất. Client sẽ dựa vào các dữ liệu được trả về để hiển thị lên cho admin. Admin có thể xóa các bình luận không phù hợp.

### Chức năng dành cho Host

Để sử dụng các chức năng của host thì phải đăng nhập bằng tài khoản host.

- **Thêm phòng**: khi host truy cập vào trang này và nhập tất cả các thông tin về phòng mới mà mình sẽ thêm vào, sau đó dữ liệu sẽ được gửi lên server để thêm vào CSDL và gửi trả về phản hồi cho client. Nếu thêm thành công thì client sẽ chuyển đến trang quản lý phòng của host, nếu thêm thất bại thì client sẽ thông báo “Thêm phòng thất bại!”.

- **Cập nhật thông tin host**: khi host truy cập vào trang này thì server sẽ trả về thông tin cá nhân của host, client sẽ dựa vào dữ liệu được trả về để hiển thị lên giao diện host. Tại đây host có thể chỉnh sửa tất cả các thông tin như: Tên doanh nghiệp, email, mật khẩu, logo, …

- **Xác nhận đặt phòng**: khi host truy cập vào trang này thì server sẽ trả về thông tin của tất cả các giao dịch chờ host xác nhận, client sẽ dựa vào dữ liệu được trả về để hiển thị lên giao diện host. Tại đây host có thể chọn phòng để xác nhận giao dịch của khách hàng hoặc cũng có thể hủy giao dịch.

- **Quản lý phòng**: khi host truy cập vào trang này thì server sẽ trả về dữ liệu về tất cả các phòng của host đó, client sẽ dựa vào dữ liệu được trả về để hiển thị lên giao diện host. Tại đây host có thể chỉnh sửa phòng, check-in, check-out,…

- **Thanh toán**: sau khi host check-out một phòng nào đó thì toàn bộ thông tin về quá trình sử dụng phòng đó sẽ được hiển thị lên form thanh toán. Các dữ liệu này sẽ được gửi lên server để hoàn thành việc thanh toán.

- **Thống kê giao dịch của host**: khi host truy cập vào trang này thì server sẽ trả về thông tin giao dịch của host, client sẽ sử dụng những dữ liệu này để hiển thị lên giao diện các thông tin như: doanh thu, tổng các giao dịch, tăng trưởng, …

- **Thên chi nhánh**: khi host truy cập vào trang này thì client sẽ hiện lên form để thêm chi nhánh mới. Sau khi điền đầy đủ thông tin để yêu cầu thêm chi nhánh mới thì dữ liệu sẽ được gửi lên server và chờ admin duyệt.

### Chức năng dành cho member

Để sử dụng các chức năng của member thì phải đăng nhập bằng tài khoản member.

- **Cập nhật thông tin member**: khi người dùng sử dụng chức năng này thì server sẽ trả về các thông tin cá nhân của người dùng đó, client sẽ dựa vào dữ liệu được trả về để hiển thị giao diện. Tại đây người dùng có thể chỉnh sửa các thông tin của mình.

- **Đặt phòng**: người dùng sẽ chọn một phòng sau đó tiến hành đặt phòng. Người dùng sẽ điền thông tin vào form đặt phòng và chọn phương thức thanh toán. Sau khi chuyển tiền cọc xong, người dùng sẽ đợi host xác nhận và hoàn thanh đặt phòng và nhận về code check-in. Người dùng có thể sử dụng code này để check-in và check-out phòng.

- **Xem lịch sử giao dịch**: khi người dùng sử dụng chức năng này thì server sẽ trả về toàn bộ lịch sử giao dịch của người dùng đó, client sẽ dựa vào dữ liệu được trả về để hiển thị lên giao diện. Tại đây người dùng có thể xem nhưng không thể điều chỉnh lịch sử giao dịch.

- **Bình luận**: người dùng có thể nhập bình luận các nhân của mình bên dưới phòng mà họ chọn, người dùng cũng có thể kèm theo hình ảnh bên trong bình luận của mình.

- **Đánh giá**: người dùng có thể đánh giá dịch vụ của phòng theo thang điểm từ 1 đến 5.

- **Báo cáo bình luận**: người dùng có thể báo cáo các bình luận mà mình cho là không phù hợp. Khi báo cáo thì phải cần có lý do báo cáo.

- **Tương tác với chatbot**: người dùng có thể tương tác với chatbot với những câu hỏi liên quan tới hệ thống như, giá thành, vị trí, cách thanh toán, …

### Các chức năng khác

- **Tìm kiếm thông tin theo yêu cầu**: người dùng sẽ cho biết như cầu của mình như giá, vị trí, đánh giá, ... Server sẽ tìm kiếm và trả về danh sách các host gần đúng với nhu cầu của người dùng nhất.

- **Gợi ý các địa điểm**: khi người dùng sử dụng chức năng này thì client sẽ yêu cầu sử dụng vị trí hiện tại của người dùng và tiến hành gửi tọa độ của người dùng lên server để tìm kiếm. Kết quả trả về là danh sách các địa điểm gần với người dùng nhất.

- **Đăng nhập thông qua bên thứ ba**: hệ thống cho phép sử dụng tài khoản Facebook hoặc Google để đăng nhập vào hệ thống với vai trò member.

### Thanh toán PAYPAL

Paypal là một dịch vụ tài chính trung gian quốc tế, cho phép người dùng thanh toán và chuyển tiền quốc tế qua mạng internet một cách nhanh chóng, dễ dàng. Paypal được xem là một “ví điện tử mang tầm quốc tế” có thể liên kết với các tài khoản ngân hàng giúp cho việc thanh toán điện tử trở nên vô cùng dễ dàng. Người dùng có thể sử dụng ví Paypal để thanh toán các khoản phí nếu hệ thống đó có liên kết với Paypal.

### Thanh toán VNPAY

Vnpay là tên thương hiệu của Công ty Cổ phần Giải pháp Thanh toán Việt Nam, chuyên cung cấp các giải pháp thanh toán điện tử chủ yếu thông qua liên kết với các tài khoản ngân hàng. Vnpay giúp người dùng thanh toán qua Cổng thanh toán Vnpay-QR. Hiểu một cách đơn giản, bạn có thể thanh toán online cho các hàng hóa dịch vụ qua Vnpay bằng mã QR (Quick Response code) nhờ tính năng QR Pay được tích hợp sẵn với tài khoản ngân hàng.

### Tích hợp bản đồ

Hệ thống sẽ sử dụng thử viện react-leaflet để hiển thị bản đồ trong việc chọn địa chỉ khi đăng kí. Khi người dùng nhập địa chỉ vào input text hoặc click trực tiếp lên bảng đồ thì hệ thống sẽ gọi api có sẵn là “https://api.opencagedata.com/geocode” để lấy về kinh độ và vĩ đọ ứng với địa chỉ mà người dùng nhập. Sau đó hệ thống sẽ lấy kinh độ và vĩ độ đó để hiển thị lên bản đồ thống qua thư viện react-leaflet.

## Thiết kế thành phần dữ liệu

## Sơ đồ Usecase

## Mô hình dữ liệu mức quan niệm CDM



Hình . Mô hình thực thể mức quan niệm

## Mô hình dữ liệu mức vật lý PDM

|  |  |  |  |
| --- | --- | --- | --- |
| **STT** | **Tên thuộc tính** | **Kiểu dữ liệu** | **Mô tả** |
| 1 | ID\_MEMBER | INTEGER | Mã của member |
| 2 | FULLNAME | VARCHAR(50) | Họ tên của member |
| 3 | ACCOUNT | VARCHAR(50) | Tài khoản của member |
| 4 | PASSWORD | VARCHAR(32) | Mật khẩu của member |
| 5 | PHONE | VARCHAR(10) | Số điện thoại member |
| 6 | ADDRESS | VARCHAR(100) | Địa chỉ của member |
| 7 | LATITUDE | DOUBLE | Kinh độ của member dựa trên địa chỉ |
| 8 | LONGTITUDE | DOUBLE | Vĩ độ của member dựa trên địa chỉ |
| 9 | REGISTRATION | DATETIME | Ngày đăng kí tài khoản |

Bảng . Thực thể MEMBER

|  |  |  |  |
| --- | --- | --- | --- |
| **STT** | **Tên thuộc tính** | **Kiểu dữ liệu** | **Mô tả** |
| 1 | ID\_HOST | INTEGER | Mã của host |
| 2 | FULLNAME\_HOST | VARCHAR(50) | Họ tên của host |
| 3 | COMPANY\_NAME | VARCHAR(100) | Tên doanh nghiệp |
| 4 | EMAIL\_HOST | VARCHAR(50) | Email của host |
| 5 | PASSWORD\_HOST | VARCHAR(32) | Mật khẩu của host |
| 6 | PAYPAL\_ID | VARCHAR(200) | Client-id của tài khoản paypal |
| 7 | BANK\_NUMBER | VARCHAR(20) | Số tài khoản ngân hàng của host |
| 8 | BRANCH\_NAME | VARCHAR(200) | Tên ngân hàng của số tài khoản host |
| 9 | OWNER\_ACCOUNT | VARCHAR(100) | Tên chủ tài khoản ngân hàng |
| 10 | PHONE\_HOST | INTEGER | Số điện thoại của host |
| 11 | ADDRESS\_HOST | VARCHAR(100) | Địa chỉ của doanh nghiệp |
| 12 | LOGO\_HOST | VARCHAR(100) | Địa chỉ lưu trữ logo của host trên server |
| 13 | LATITUDE | DOUBLE | Kinh độ của doanh nghiệp dựa trên địa chỉ |
| 14 | LONGTITUDE | DOUBLE | Vĩ độ của doanh nghiệp dựa trên địa chỉ |

Bảng . Thực thể HOST

|  |  |  |  |
| --- | --- | --- | --- |
| **STT** | **Tên thuộc tính** | **Kiểu dữ liệu** | **Mô tả** |
| 1 | ID\_PROPOSAL | INTEGER | Mã của yêu cầu |
| 2 | FULLNAME | VARCHAR(50) | Họ tên người gửi yêu cầu |
| 3 | COMPANY | VARCHAR(100) | Tên doanh nghiệp |
| 4 | EMAIL | VARCHAR(50) | Email của người gửi yêu cầu |
| 5 | PHONE | INTEGER | Số điện thoại của người gửi yêu cầu |
| 6 | ADDRESS | VARCHAR(100) | Địa chỉ của doanh nghiệp |
| 7 | LATITUDE | DOUBLE | Kinh độ của doanh nghiệp dựa trên địa chỉ |
| 8 | LONGTITUDE | DOUBLE | Vĩ độ của doanh nghiệp dựa trên địa chỉ |
| 9 | STATE | VARCHAR(10) | Trạng thái: đăng kí mới hoặc thêm chi nhánh |

Bảng . Thực thể PROPOSAL

|  |  |  |  |
| --- | --- | --- | --- |
| **STT** | **Tên thuộc tính** | **Kiểu dữ liệu** | **Mô tả** |
| 1 | ID\_CONVENIENT | INTEGER | Mã của tiện ích |
| 2 | NAME\_CONVENIENT | VARCHAR(50) | Tên của tiện ích |
| 3 | LOGO\_CONVENIENT | VARCHAR(100) | Địa chỉ lưu trữ logo của tiện ích trên server |

Bảng . Thực thể CONVENIENT

|  |  |  |  |
| --- | --- | --- | --- |
| **STT** | **Tên thuộc tính** | **Kiểu dữ liệu** | **Mô tả** |
| 1 | ID\_ROOM | INTEGER | Mã của phòng |
| 2 | NAME\_ROOM | VARCHAR(50) | Tên của phòng |
| 3 | TYPE\_ROOM | INTEGER | Loại của phòng |
| 4 | CAPACITY | INTEGER | Số lượng người trong một phòng |
| 5 | PRICE | INTEGER | Giá của phòng cho 24 giờ |
| 6 | IMAGES\_ROOM | VARCHAR(1000) | Các địa chỉ lưu trữ các hình ảnh của phòng trên server |
| 7 | CONVENIENTS | VARCHAR(200) | Chuỗi các mã của các tiện ích tương ứng của phòng |
| 8 | ID\_HOST | INTEGER | Mã của host (khóa ngoại liên kết với thực thể HOST) |

Bảng . Thực thể ROOMS

|  |  |  |  |
| --- | --- | --- | --- |
| **STT** | **Tên thuộc tính** | **Kiểu dữ liệu** | **Mô tả** |
| 1 | ID\_BOOKING | INTEGER | Mã của lịch đặt |
| 2 | TYPE\_ROOM | INTEGER | Mã của loại phòng |
| 3 | FULLNAME | VARCHAR(100) | Họ và tên của người đặt phòng |
| 4 | PHONE | VARCHAR(10) | Số điện thoại của người đặt phòng |
| 5 | CHECKIN\_DATE | TIMESTAMP | Ngày, giờ nhận phòng |
| 6 | CHECKOUT\_DATE | TIMESTAMP | Ngày, giờ trả phòng |
| 7 | DEPOSIT | INTEGER | Tiền đặt cọc |
| 8 | CODE | VARCHAR(10) | Mã code để check-in |
| 9 | STATE | VARCHAR(20) | Trạng thái của lịch đặt phòng |
| 10 | ID\_ROOM | INTEGER | Mã của phòng được đặt (khóa ngoại liên kết với thực thể ROOM) |
| 11 | ID\_MEMBER | INTEGER | Mã của thành viên được đặt (khóa ngoại liên kết với thực thể MEMBER) |
| 12 | ID\_HOST | INTEGER | Mã của host được đặt (khóa ngoại liên kết với thực thể HOST) |

Bảng . Thực thể BOOKING\_SCHEDULE

|  |  |  |  |
| --- | --- | --- | --- |
| **STT** | **Tên thuộc tính** | **Kiểu dữ liệu** | **Mô tả** |
| 1 | ID\_TRANSACTION | INTEGER | Mã của giao dịch |
| 2 | FULLNAME | VARCHAR(100) | Họ tên của khách hàng thực hiện giao dịch |
| 3 | PHONE | VARCHAR(10) | Số điện thoại khách hàng thực hiện giao dịch |
| 4 | CHECKIN\_DATE | DATETIME | Ngày, giờ nhận phòng |
| 5 | CHECKOUT\_DATE | DATETIME | Ngày, giờ trả phòng |
| 6 | TOTAL\_PAYMENT | INTEGER | Tổng tiền phải trả |
| 7 | DEPOSIT | INTEGER | Số tiền đã đặt cọc |
| 8 | STATE | VARCHAR(20) | Trạng thái của giao dịch |
| 9 | ID\_ROOM | INTEGER | Mã của phòng được đặt (khóa ngoại liên kết với thực thể ROOM) |
| 10 | ID\_MEMBER | INTEGER | Mã của thành viên được đặt (khóa ngoại liên kết với thực thể MEMBER) |

Bảng . Thực thể TRANSACTION

|  |  |  |  |
| --- | --- | --- | --- |
| **STT** | **Tên thuộc tính** | **Kiểu dữ liệu** | **Mô tả** |
| 1 | ID\_COMMENT | INTEGER | Mã của bình luận |
| 2 | CONTENT | VARCHAR(1000) | Nội dung của bình luận |
| 3 | IMAGE | VARCHAR(1000) | Địa chỉ lưu trữ các hình ảnh của bình luận trên server |
| 4 | DATE\_CREATE | TIMESTAMP | Ngày giờ gửi comment |
| 5 | ID\_ROOM | INTEGER | Mã của phòng được đặt (khóa ngoại liên kết với thực thể ROOM) |
| 6 | ID\_MEMBER | INTEGER | Mã của thành viên được đặt (khóa ngoại liên kết với thực thể MEMBER) |

Bảng . Thực thể COMMENT

|  |  |  |  |
| --- | --- | --- | --- |
| **STT** | **Tên thuộc tính** | **Kiểu dữ liệu** | **Mô tả** |
| 1 | ID\_EVALUATE | INTEGER | Mã của đánh giá |
| 2 | POINT | INTEGER | Điểm đánh giá |
| 3 | ID\_ROOM | INTEGER | Mã của phòng được đặt (khóa ngoại liên kết với thực thể ROOM) |
| 4 | ID\_MEMBER | INTEGER | Mã của thành viên được đặt (khóa ngoại liên kết với thực thể MEMBER) |

Bảng . Thực thể EVALUATE

|  |  |  |  |
| --- | --- | --- | --- |
| **STT** | **Tên thuộc tính** | **Kiểu dữ liệu** | **Mô tả** |
| 1 | ID\_REPORT | INTEGER | Mã của báo cáo |
| 2 | REASON | VARCHAR(500) | Lý do báo cáo |
| 3 | ID\_COMMENT | INTEGER | Mã của bình luận được đặt (khóa ngoại liên kết với thực thể COMMENT) |
| 4 | ID\_MEMBER | INTEGER | Mã của thành viên được đặt (khóa ngoại liên kết với thực thể MEMBER) |

Bảng . Thực thể EVALUATE

## Sơ đồ chức năng

Hình . Sơ đồ chức năng

# HỆ THỐNG HỖ TRỢ

Giới thiệu về chatbot và cách ứng dung của Rasa chatbot vào website

## Chatbot

Chatbot được định nghĩa là chương trình máy tính dựa trên AI mô phỏng các cuộc hội thoại của con người. Chúng cũng được gọi là trợ lý kỹ thuật ảo có thể hiểu các khả năng của con người. Những con bot sẽ diễn giải và xử lý các yêu cầu của người dùng và đưa ra câu trả lời cho họ một cách nhanh chóng. Chatbot không chỉ thông minh về việc hiểu và trả lời các câu hòi mà còn có thể truy xuất dữ liệu từ CSDL.

### Tổng quan về chatbot

1. **Giới thiệu về chatbot**

Chatbot chính là hệ thống các Bot ở trong các trạng thái trực tuyến, trên các website hoặc trên các nền tảng chat khác của social media để “chat tự động” với người dùng. Chatbot tương tác với chúng ta như một hệ thống trả lời tin nhắn nhanh chóng, tự động. Bằng cách xây dựng, giả lập các mô hình tương tác, kịch bản tương tác của con người sử dụng phương pháp trong Machine Learning (ML), hệ thống chatbot có thể tự học, tự hiểu các câu hỏi, nhu cầu của người dùng và đưa ra các phản hồi sao cho phù hợp. [7]

Các khải niệm được sử dụng trong chatbot:

+ Intent: là ý định của người dùng trong câu nói.

+ Entity: là các thực thể bổ nghĩa cho ý định.

+ Stop words: là các từ xuất nhiện nhiều trong câu nói nhưng không mang ý nghĩa quan trọng

Ví dụ: “Tôi muốn xem các khách sạn tốt ở Đà Lạt”

Intent của của câu trên là muốn biết thông tin chi tiết về các khách sạn hoặc nơi lưu trú khi đi du lịch Đà Lạt.

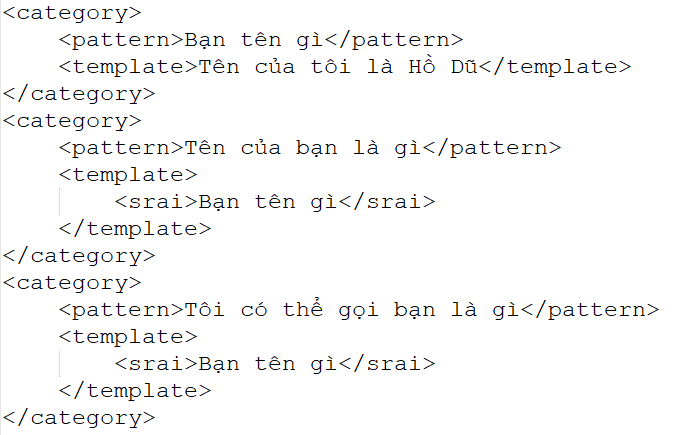
Entiy của câu trên là “Đà Lạt”, một địa danh, địa điểm.

Stop words của câu trên là các từ “Tôi”, “các”, “ở”.

1. **Các phương pháp xây dựng chatbot**

-**Pattern matchers – phát hiện trùng khớp các mẫu câu**: phương pháp này sẽ tìm kiếm các mẫu câu trong câu hỏi của người dùng và xem có khớp với các câu hỏi đã được “huấn luyện” lưu trữ trong hệ thống để đưa ra câu trả lời phù hợp nhất. Phương pháp pattern matchers dựa theo cấu trúc tiêu chuẩn của ngôn ngữ lập trình “Artificial Intelligence Markup Language (AIML). [8]

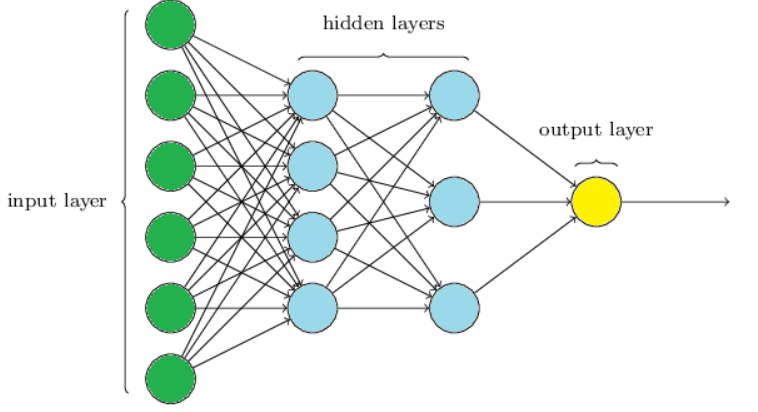
Ví dụ:



Giả sử bạn hỏi “Bạn tên gì” hoặc “Tên của bạn là gì” thì bot cũng sẽ trả lời là “Tên của tôi là Hồ Dũ”.

- **Multinational Navie Bayes – phân lớp các câu**: là thuật toán phổ biến nhất để phân loại văn bản trong mảng xử lý ngôn ngữ tự nhiên – NLP (Natural Language Processing). Cụ thể các mẫu câu có ý nghĩa tương tự nhau sẽ được nhóm chung một lớp (class), tương tự sẽ có nhiều class khác nhau. Chúng ta sẽ có những câu trả lời riêng ứng với từn class. Khi có một câu mới đến thì bot sẽ phân tách câu đó ra thành các từ (word) và so sánh các word trong câu với từng class. Class nào chứa nhiều word trong câu được đưa vào nhất thì bot sẽ đưa ra câu trả lời ứng với class đó. [8]

- **Artificial Neural Network (ANN) – mạng nơ-ron nhân tạo**: ANN là một nhóm các nút (node) được nối với nhau, gần giống với mạng nơ-ron trong não con người, nó có thể hiểu là một hệ thống tính toán được xây dựng dựa trên các mạng nơ-ron sinh học. Phương pháp này là tính toán tỉ lệ chính xác dữ liệu đầu ra từ dữ liệu đầu vào bằng cách tính toán các trọng số cho mỗi kết nối (connection) từ các lần lặp lại trong khi huấn luyện dữ liệu cho chatbot. Mỗi bước huấn luyện đầu ra được xuất ra với độ chính xác cao. Mỗi câu sẽ được phân thành từng từ và mỗi từ chính là dữ liệu đầu vào input đưa vào ANN. Các từ sẽ được kết nối với nhau khi huấn luyện dữ liệu Chatbot, các kết nối này sẽ được đo bằng trọng số cho hàng ngàn lần lặp lại ở các thời điểm khác nhau. Mỗi lần lặp lại trọng số có thể được cải thiện và độ chính xác sẽ cao hơn. [8]



Hình . Minh họa về mạng ANN [8]

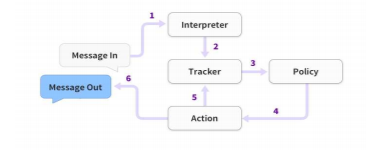
- **Natural Language Processing (NLP) – xử lý ngôn ngữ tự nhiên**: NLP là một nhánh riêng của trí tuệ nhân tạo, nó tập trung vào việt xử lý ngôn ngữ của ngon người. Hiểu được ý nghĩa ngôn ngữ của con người là một điều rất khó, để làm được việc này thì hệ thống NLP phải trải qua các bước như: phân tích hình thái câu, phân tích cú pháp, phân tích ngữ nghĩa, tích hợp văn bản, phân tích thực nghĩa. NLP được ứng dụng vào cuộc sống thực tế như: chatbot, nhận dạng giọng nói, dịch tự động, tìm kiếm thông tin,… [8]

### Framework Rasa

Rasa là một framework máy học mã nguồn mở dùng để xử lý giọng nói, dịch tự động văn bản hoặc tham gia vào các cuộc trò truyện với con người. Rasa sẽ sinh câu trả lời dựa trên ngữ cảnh trước đó của câu input và chọn những từ ngữ thể hiện hành động liên quan đến bối cảnh của câu sắp được sinh ra. Rasa gồm 2 phần chính là Rasa NLU (Nature Language Understanding) và Rasa core. [9]

Rasa NLU được xem như là một cánh tay để lấy đầu vào từ người dùng. Nó là một thư viện để hiểu ngôn ngữ tự nhiên thực hiện hiên phân đoạn ý định và trích xuất thực thể từ câu đầu vào của người dùng và giúp bot hiểu những gì người dùng đang nói. [10]

Rasa core giống như bộ não sẽ đảm nhận quyết định sinh ra câu trả lời dựa trên các Intent. Entity và các thành phần cấu trúc đã được Rasa NLU xử lý. [10]



Hình . Sơ đồ luồng dữ liệu của Rasa [10]

Giải thích về luồng dữ liệu trên:

1.văn bản từ người dùng cuối được cung cấp cho Rasa NLU, tại đây NLU sẽ cho ra output có cấu trúc chứa văn bản gốc, mục đích của câu nói và các thực thể trong câu.

2.trình theo dõi duy trì trạng thái hội thoại và sinh đầu ra có cấu trúc giống trình thông dịch NLU.

3.đầu ra từ trình theo dõi được cung cấp cho chính sách, hoạt động trên trạng thái hiện tại của trình theo dõi.

4.chính sách quyết định hành động thích hợp tiếp theo là thực hiện.

5.nhật ký của hành động đã chọn được trình theo dõi duy trì.

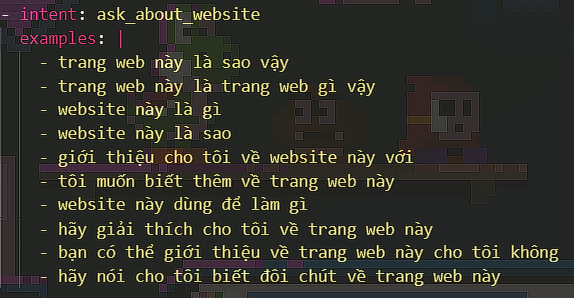
6.phản hồi thích hợp được trả ra cho người dùng.

### Xây dựng bộ dữ liệu NLU

Dữ liệu NLU bao gồm các ý định, mỗi ý định bao gồm tên ý định và các câu đầu vào mẫu có nội dung tương tự nhau để thể hiện cho ý định đó.

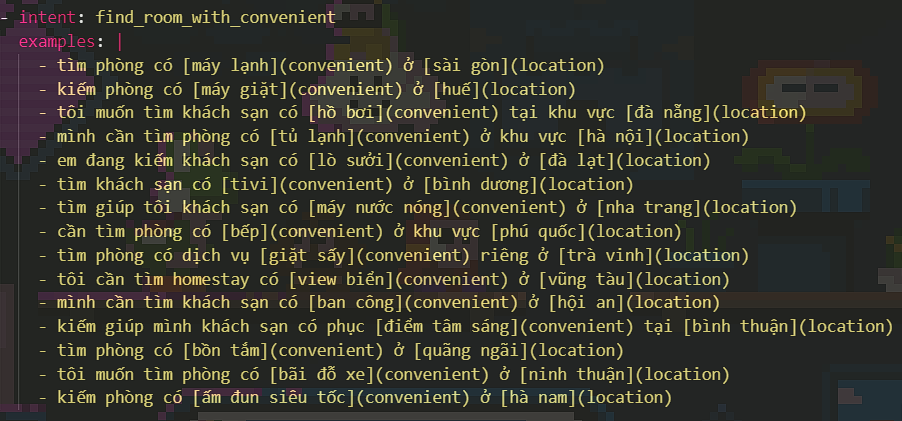
Thực thể là các từ hay các cụm từ dùng để xác định các khái niệm nào đó trong dữ liệu của mình. Mỗi thực thể bao gồm tên thực thể và giá trị của thực thể. Mỗi ý định có thể chưa không hoặc nhiều thực thể khác nhau.

Ví dụ 1 về ý định “ask\_about\_website” không chứa thực thể:



Hình . Ví dụ về ý định “ask\_about\_website”

Ví dụ 2 về ý định “find\_room\_with\_convenient” chứa 2 thực thể [convenient] và [location]:



Hình . Ví dụ về ý định “find\_room\_with\_convenient”

Các ý định trong đề tài:

|  |  |  |
| --- | --- | --- |
| **STT** | **Tên ý định** | **Ý nghĩa** |
| 1 | greet | Các câu chào hỏi |
| 2 | social\_enquiry | Các câu hỏi thăm xã giao |
| 3 | goodbye | Các câu chào tạm biệt |
| 4 | mood\_great | Các câu thể hiện tâm trạng tốt, sự hài lòng |
| 5 | mood\_unhappy | Các câu thể hiện tâm trạng không tốt, sự không hài lòng |
| 6 | ask\_myself | Các câu hỏi về bot |
| 7 | ask\_gender | Các câu hỏi về giới tính của bot |
| 8 | ask\_about\_website | Các câu hỏi về trang web dùng để làm gì |
| 9 | ask\_to\_become\_host | Các câu hỏi về cách để trở thành nhà cung cấp dịch vụ (host) của hệ thống |
| 10 | ask\_host\_role | Các câu hỏi về các chức năng của nhà cung cấp dịch vụ (host) trong hệ thống |
| 11 | ask\_how\_to\_booking | Các câu hỏi về cách đặt phòng |
| 12 | ask\_for\_checkout\_policy | Các câu hỏi về chính sách đặt và trả phòng |
| 13 | ask\_for\_booking\_history | Các câu hỏi về cách xem lịch sử đặt phòng |
| 14 | ask\_for\_room\_confirmed | Các câu hỏi về cách kiểm tra xem giao dịch có được xác nhận hay chưa |
| 15 | ask\_for\_deposit | Các câu hỏi về tiền đặt cọc |
| 16 | ask\_for\_personal \_infomation | Các câu hỏi về cách xem thông tin cá nhân của người dùng |
| 17 | ask\_for\_login | Các câu hỏi về cách đặt phòng |
| 18 | ask\_for\_register | Các câu hỏi về cách đặt kí tài khoản thành viên |
| 19 | ask\_for\_booking\_reject | Các câu hỏi về vấn đề phòng bị từ chối giao dịch |
| 20 | ask\_cannot\_comment | Các câu hỏi tại sao không thể bình luận |
| 21 | ask\_cannot\_vote | Các câu hỏi tại sao không thể đánh giá |
| **STT** | **Tên ý định** | **Ý nghĩa** |
| 22 | ask\_cannot\_open \_personal\_infomation | Các câu hỏi tại sao không thể vào xem thông tin cá nhân của người dùng |
| 23 | ask\_cannot\_check\_room \_available | Các câu hỏi tại sao không thể kiểm tra tình trạng còn phòng |
| 24 | ask\_to\_checkin | Các câu hỏi về cách nhận phòng |
| 25 | ask\_cannot\_login | Các câu hỏi tại sao không thể đăng nhập |
| 26 | ask\_cannot\_register | Các câu hỏi tại sao không thể đăng kí |
| 27 | requirement\_to\_become \_host | Các câu hỏi về yêu cầu để tham gia vào hệ thống với vai trò là nhà cung cấp dịch vụ (host) |
| 28 | find\_room\_in\_location | Các câu hỏi để tìm phòng ở một địa điểm nào đó |
| 29 | find\_room\_hightest\_rate | Các câu hỏi để tìm phòng có đánh giá cao nhất |
| 30 | find\_room\_lowest\_price | Các câu hỏi để tìm phòng có giá thấp nhất |
| 31 | find\_room\_hightest\_price | Các câu hỏi để tìm phòng có giá cao nhất |
| 32 | find\_room\_with\_price \_from\_low\_to\_hight | Các câu hỏi để tìm phòng có giá nằm giữa một khoảng nào đó |
| 33 | find\_room\_with\_price \_from\_hight\_to\_low | Các câu hỏi để tìm phòng có giá nằm giữa một khoảng giá nào đó |
| 34 | find\_room\_with\_price \_lower\_than | Các câu hỏi để tìm phòng có giá nhỏ hơn một giá trị nào đó |
| 35 | find\_room\_with\_name | Các câu hỏi để tìm phòng bằng tên nhà cung cấp dịch vụ |
| 37 | find\_room\_with \_convenient | Các câu hỏi để tìm phòng có các tiện ích như yêu cầu người dùng |

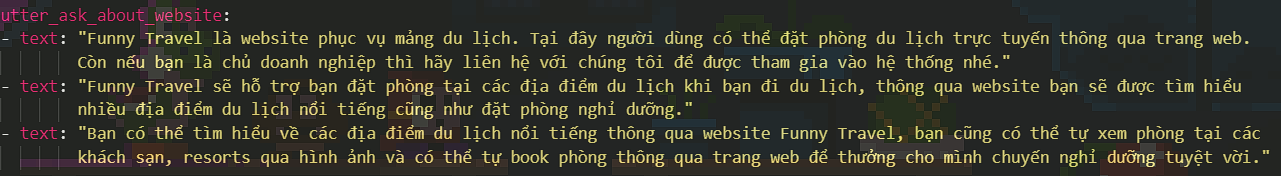
Bảng . Bảng thống kê các ý định trong đề tài

### Xây dựng các câu trả lời liên quan

Ứng với mỗi ý định của người dùng sẽ có một hành động khác nhau để trả lời có các ý định đó. Các hành động trả lời có thể bao gồm văn bản, hình ảnh hoặc liên kết.

Với các ý định không cần truy vấn cơ sở dữ liệu sẽ tạo ra các câu trả lời mẫu, rasa sẽ chọn ngẫu nhiên một trong các câu trả lời mẫu đó để phản hồi cho người dùng.

Ví dụ về hành động “utter\_ask\_about\_website”:



Hình . Ví dụ về hành động “utter\_ask\_about\_website”

Với các ý định phải truy vấn cơ sở dữ liệu rasa sẽ gọi tới api action để thực hiện thực hiện việc truy vấn và sau đó sẽ phản hồi cho người dùng.

Ví dụ về hành động “action\_find\_room\_with\_convenient”:



Hình . Ví dụ về hành động “action\_find\_room\_with\_convenient”

Các hành động tương ứng với các ý định trong đề tài:

|  |  |  |
| --- | --- | --- |
| **STT** | **Tên hành động** | **Ý nghĩa** |
| 1 | Utter\_greet | Trả lời các câu chào hỏi của người dùng |
| 2 | Utter\_social\_enquiry | Trả lời các câu hỏi thăm xã giao của người dùng |
| 3 | Utter\_goodbye | Trả lời các câu chào tạm biệt của người dùng |
| 4 | Utter\_mood\_great | Trả lời các câu thể hiện tâm trạng tốt, sự hài lòng của người dùng |
| 5 | Utter\_mood\_unhappy | Trả lời các câu thể hiện tâm trạng không tốt, sự không hài lòng của người dùng |
| 6 | Utter\_ask\_myself | Trả lời các câu hỏi về bot của người dùng |
| 7 | Utter\_ask\_gender | Trả lời các câu hỏi về giới tính của bot |
| 8 | Utter\_ask\_about \_website | Trả lời các câu hỏi website dùng để làm gì của người dùng |
| 9 | Utter\_ask\_to\_become \_host | Trả lời các câu hỏi về cách tham gia vào hệ thống với vai trò là nhà cung cấp dịch vụ (host) |
| 10 | Utter\_ask\_host\_role | Trả lời các câu hỏi về chức năng của nhà cung cấp dịch vụ (host) trong hệ thống |
| 11 | Utter\_ask\_how\_to \_booking | Trả lời các câu hỏi về cách đặt phòng của người dùng |
| 12 | Utter\_ask\_for\_check \_out\_policy | Trả lời các câu hỏi về chính sách đặt phòng và hủy phòng của người dùng |
| 13 | Utter\_ask\_for\_booking \_history | Trả lời các câu hỏi về cách xem lịch sử đặt phòng của người dùng |
| 14 | Utter\_ask\_for\_deposit | Trả lời các câu hỏi về tiền đặt cọc của người dùng |
| 15 | Utter\_ask\_for\_personal \_infomation | Trả lời các câu hỏi về cách xem thông tin của người dùng |
| 16 | Utter\_ask\_for\_login | Trả lời các câu hỏi về cách đăng nhập của người dùng |
| 17 | Utter\_ask\_for\_register | Trả lời các câu hỏi về cách đăng kí tài khoản thành viên của người dùng |
| **STT** | **Tên hành động** | **Ý nghĩa** |
| 18 | Utter\_ask\_booking \_reject | Trả lời các câu hỏi vì sao giao dịch bị từ chối của người dùng |
| 19 | Utter\_ask\_cannot \_comment | Trả lời các câu hỏi vì sao không thể bình luận của người dùng |
| 20 | Utter\_ask\_cannot\_vote | Trả lời các câu hỏi vì sao không thể đánh giá của người dùng |
| 21 | Utter\_ask\_cannot\_open \_personal\_infomation | Trả lời các câu vì sao không thể truy cập vào trang thông tin cá nhân của người dùng |
| 22 | Utter\_ask\_cannot \_check\_room\_available | Trả lời các câu vì sao không thể kiểm tra tình trạng còn phòng của người dùng |
| 23 | Utter\_ask\_to\_checkin | Trả lời các câu hỏi về cách đặt phòng của người dùng |
| 24 | Utter\_ask\_cannot\_login | Trả lời các câu hỏi vì sao không thể đăng nhập của người dùng |
| 25 | Utter\_ask\_cannot \_register | Trả lời các câu hỏi vì sao không thể đăng kí tài khoản của người dùng |
| 26 | Utter\_requirement\_to \_become\_host | Trả lời các câu về yêu cầu để trở thành nhà cung cấp dịch vụ (host) của người dùng |
| 27 | Action\_find\_room\_in \_location | Trả về kết quả là thông tin các host có phòng ở vị trí mà người dùng muốn hỏi |
| 28 | Action\_find\_room \_hightest\_rate | Trả về kết quả là thông tin các host có phòng với trung bình điểm đánh giá cao nhất |
| 29 | Action\_find\_room \_lowest\_price | Trả về kết quả là thông tin các host có phòng với giá thành thấp nhất |
| 30 | Action\_find\_room \_hightest\_price | Trả về kết quả là thông tin các host có phòng với giá thành cao nhất |
| 31 | Action\_find\_room \_with\_custom\_price | Trả về kết quả là thông tin các host có phòng với giá nằm giữa một khoảng giá nào đó |
| 32 | Action\_find\_room \_with\_price\_lower\_than | Trả về kết quả là thông tin các host có phòng với giá thấp hơn giá mà người dùng muốn hỏi |
| **STT** | **Tên hành động** | **Ý nghĩa** |
| 33 | Action\_find\_room \_with\_name | Trả về kết quả là thông tin các host có gần giống nhất với tên mà người dùng muốn tìm |
| 34 | Action\_find\_room \_with\_convenient | Trả về kết quả là thông tin các host có các tiện ích mà người dùng muốn tìm |
| 35 | Action\_default | Đây là hành động ngoại lệ, khi bot không hiểu ý định của người dùng thì hành động này sẽ được gọi để yêu cầu người dùng nói rõ hơn ý định của mình |

Bảng . Bảng thống kê các hành động trong đề tài

### Xây dựng dữ liệu đối thoại

Xây dựng dữ liệu đối thoại mà một bước quan trọng trong quá trình xây dựng rasa chatbot. Ở bước này chúng ta sẽ xây dựng các cuộc trò truyện mẫu, các trường hợp có thể xảy ra trong cuộc trò truyện giữa người dùng và bot. Dựa trên những cuộc trò truyện mẫu này mà rasa chatbot có thể hiểu được rõ hơn ý định của người dùng và đưa ra câu trả lời phù hợp.

Nói cách khác đây là bước mà chúng ta dạy cho bot biết khi nhận được ý định nào thì sẽ trả lời như thế nào cho phù hợp. Tất nhiên chúng ta cũng không cần phải vét cạn tất cả các câu chuyện, nhưng cũng đừng xây dựng dữ liệu đối thoại ít quá sẽ khiến cho bot kém “thông minh”.

Ví dụ về dữ liệu đối thoại

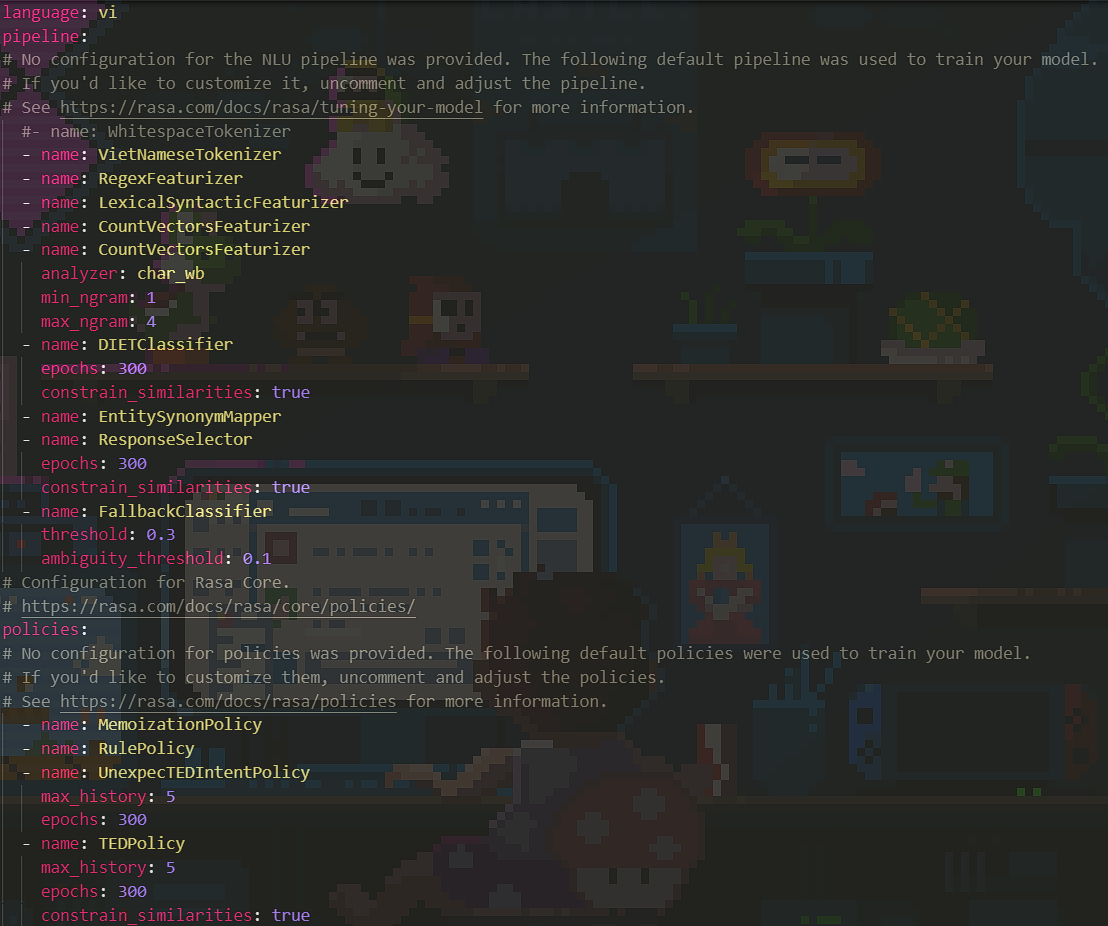


Hình . Ví dụ về dữ liệu đối thoại của rasa chatbot

### Huấn luyện mô hình

Huấn luyện mô hình được xem như là bước mà bot sẽ học tất cả những gì mà người dùng cung cấp mà sẽ lưu chúng lại trong các model. Các model này là cở sở để bot có thể trả lời các câu hỏi khác của người dùng.

Chúng ta có thể tùy chọn cách huấn luyện mô hình bằng cách chỉnh sửa file config.yml



Hình . Cấu hình Rasa để huấn luyện mô hình

Các thành phần để huấn luyện NLU sẽ được khai báo ở phần pipeline. Chúng ta sẽ tiến hành tùy chỉnh các thành phần trong pipeline để chuẩn bị cho việc huấn luyện mô hình. Với mỗi câu đầu vào từ NLU, rasa sẽ thực hiện tuần tự các công việc như tách từ tiếng Việt, trích xuất thực thể, phân loại ý định, lựa chọn phản hồi. Chúng ta cũng đặt lại tham số epochs để rasa được học nhiều hơn giúp cải thiện khả năng “thông minh” của bot.

Các thành phần để huấn luyện core sẽ được khai báo ở phần policies. Các thành phần trong policies được hiểu như là các con đường được bot lựa chọn để đưa ra câu trả lời cho người dùng với độ ưu tiên đi từ trên xuống.

## Hệ thống phân loại bình luận

//Chú ý, chỉ người dùng có tài khoản mới được quyền bình luận. Nếu bình luận là tích cực thì cho phép hiển thị luôn; ngược lại, khi bình luận là tiêu cực thì ẨN bình luận và gửi thông báo bình luận không tốt về cho admin duyệt hoặc xóa

### Tổng quan về bài toán phân lớp

//Trình bày tổng quan về bài toán phân lớp

// Phương pháp thực hiện

// Phương pháp đánh giá

### Ứng dụng bài toán vào đề tài

//Phần này có thể đơn giản dùng Naïve Bayes/ LSTM hoặc mô hình khác

### Đánh giá kết quả thực nghiệm

### Giới thiệu về LSTM hoặc SVM

//có thể dùng LSTM hoặc SVM hoặc deep neural network khác

## Tích hợp

//phần này cô sẽ hướng dẫn sau khi đi được ½ chặn đường

## Thảo luận kết quả đạt được

# GIỚI THIỆU CHƯƠNG TRÌNH

// Viết một đoạn mô tả tóm tắt các em sẽ trình bày những gì trong chương này

Các em chỉ giới thiệu một số hình ảnh đặc trưng nhất, cần phải khoe nhất, CÒN LẠI ĐƯA HẾT VÀO PHỤ LỤC

## Giao diện website

## Giao diện ứng dụng

## Chatbot

## Hệ thống phân loại bình luận

## Hệ thống tích hợp (//các em đổi tên sau)

# KẾT LUẬN

// Viết một đoạn mô tả tóm tắt các em sẽ trình bày những gì trong chương này

## Kết luận

//các em cần trình bày kết quả đạt được, hạn chế … 🡪 tất cả đều phải viết thành đoạn văn hoàn chỉnh, không gạch đầu dòng

## Hướng pháp triển

# TÀI LIỆU THAM KHẢO

|  |  |
| --- | --- |
| [1] | Sanchit Aggarwal, "Modern Web-Development using ReactJS", New Deli, India: Department of Information Technology, Northern India Engineering College, New Delhi, India, 2018. |
| [2] | Jake Spurlock, "Bootstrap", United States of America: O'Reilly Media, Inc., 1005 Gravenstein Highway North, Sebastopol, CA 95472., 2013. |
| [3] | Rasmus Lerdorf; Kevin Tatroe; Bob Kaehms; Ric McGredy;, "Propraming PHP", United States of America: O'Reilly Media, Inc., 1005 Gravenstein Highway North, Sebastopol, CA 95472., 2002. |
| [4] | DuBois Paul, "MySQL", PearSon Education, 2008. |
| [5] | Allan Kent; David Powers; Rachel Andrew;, "Introduction to MySQL and SQL." PHP Web Development with Macromedia Dreamweaver MX 2004, Apress, Berkeley, CA, 2004. 27-59., 2004. |
| [6] | Thai Nguyen, "RESTful: Phần 1 - Tìm hiểu RESTful API," 21 July 2019. [Online]. Available: https://itplusx.info/restful-api-gioi-thieu-phan-1/. |
| [7] | BigDataUni, ""Tổng quan về Chat bot (P.1) Chat bot là gì?," [Online]. Available: https://bigdatauni.com/tin-tuc/tong-quan-ve-chatbot-phan-1-chatbot-la-gi.html. |
| [8] | BigDataUni, ""Tổng quan về Chatbot (P.2) Chatbot hoạt động như thế nào"," [Online]. Available: https://bigdatauni.com/tin-tuc/tong-quan-ve-chatbot-phan-2-chatbot-hoat-dong-nhu-the-nao.html. |
| [9] | ""Introduction to Rasa Open Source"," 12 July 2021. [Online]. Available: https://rasa.com/docs/rasa/. |
| [10] | Rakesh Kumar Sharma; Manoj Joshi;, "IJERT-An Analytical Study and Review of open Source Chatbot framework, RASA", India: National Informatics Center (NIC) India, 06, June-2020. |
| [11] | BigDataUni, "Tổng quan về Chatbot (P.3): Lợi ích của Chatbot," [Online]. Available: https://bigdatauni.com/tin-tuc/tong-quan-ve-chatbot-phan-3-loi-ich-cua-chatbot.html. |

# PHỤ LỤC 1: MÔ HÌNH DỮ LIỆU MỨC VẬT LÝ

// Mô hình dữ liệu mức PDM

# PHỤ LỤC 2: BẢNG MÔ TẢ PHƯƠNG THỨC

# PHỤ LỤC 3: CÁC RÀNG BUỘC THAM CHIẾU

# PHỤ LỤC 4: LƯỢC ĐỒ CƠ SỞ DỮ LIỆU

# PHỤ LỤC 5: HÌNH ẢNH CHƯƠNG TRÌNH