

ĐẠI HỌC QUỐC GIA TP. HỒ CHÍ MINH
TRƯỜNG ĐẠI HỌC CÔNG NGHỆ THÔNG TIN
KHOA CÔNG NGHỆ PHẦN MỀM

TRẦN HỮU DANH

KHÓA LUẬN TỐT NGHIỆP
TÌM HIỂU CÔNG NGHỆ CONVERSATIONAL CHATBOT
ỨNG DỤNG TRONG XÂY DỰNG TEXT-BASED GAME

KỸ SƯ NGÀNH KỸ THUẬT PHẦN MỀM

TP. HỒ CHÍ MINH, 2016

ĐẠI HỌC QUỐC GIA TP. HỒ CHÍ MINH
TRƯỜNG ĐẠI HỌC CÔNG NGHỆ THÔNG TIN
KHOA CÔNG NGHỆ PHẦN MỀM

TRẦN HỮU DANH – 12520054

KHÓA LUẬN TỐT NGHIỆP
TÌM HIỂU CÔNG NGHỆ CONVERSATIONAL CHATBOT
ỨNG DỤNG TRONG XÂY DỰNG TEXT-BASED GAME

KỸ SƯ NGÀNH KỸ THUẬT PHẦN MỀM

GIẢNG VIÊN HƯỚNG DẪN
ThS. ĐINH NGUYỄN ANH DŨNG

TP. HỒ CHÍ MINH, 2016

DANH SÁCH HỘI ĐỒNG BẢO VỆ KHÓA LUẬN

Hội đồng chấm khóa luận tốt nghiệp, thành lập theo Quyết định số

..... ngày của Hiệu trưởng Trường Đại học Công
nghệ Thông tin.

1. – Chủ tịch.
2. – Thư ký.
3. – Ủy viên.
4. – Ủy viên.

LỜI CẢM ƠN

Trước hết em xin chân thành gửi lời cảm ơn đến thầy Đinh Nguyễn Anh Dũng, giảng viên khoa Công Nghệ Phần Mềm, Trường Đại Học Công Nghệ Thông Tin. Từ những ngày đầu tiên học lập trình Game, thầy luôn là người đã truyền cho em nhiều kinh nghiệm cũng như những bài học về động lực cực kỳ hữu ích. Thầy là người định hướng, đóng góp và sửa chữa nhiều chi tiết sai sót giúp đề tài được phát triển đúng hướng. Em hoàn thành được khóa luận là nhờ sự nhắc nhở, đôn đốc và tận tâm từ Thầy.

Phần tiếp theo em xin lòng biết ơn của mình đến tất cả Thầy, Cô giảng viên khoa Công Nghệ Phần Mềm, cũng như những thầy cô đã giảng dạy em suốt quãng thời gian học tập ở trường. Những kiến thức nền tảng là hết sức quan trọng, những kiến thức này giúp em hoàn thiện được bản thân, với nghề nghiệp hiện tại và hiện thực khóa luận này.

Cám ơn tập thể anh em developer công ty XCT đã có một môi trường thật tốt để em làm việc trong quá trình làm khóa luận.

Cảm ơn bố mẹ đã sinh ra và nuôi dạy con, đã chăm sóc và tạo điều kiện để con theo mong muốn trở thành lập trình viên của mình.

Mặc dù đã hết sức cố gắng hoàn thành đề tài khóa luận nhưng vẫn khó thể tránh khỏi những thiếu sót.

Em kính mong nhận được sự góp ý, hỗ trợ của quý Thầy, Cô để hoàn thiện hơn.

Sinh viên thực hiện

Trần Hữu Danh

TP. HCM, ngày.....tháng.....năm.....

NHẬN XÉT KHÓA LUẬN TỐT NGHIỆP
(CỦA CÁN BỘ HƯỚNG DẪN)

Tên khóa luận:

TÌM HIỂU CÔNG NGHỆ CONVERSATIONAL CHATBOT
ỨNG DỤNG TRONG XÂY DỰNG TEXT-BASED GAME

Nhóm SV thực hiện:	Cán bộ hướng dẫn/phản biện:
Trần Hữu Danh 12520054	ThS. Đinh Nguyễn Anh Dũng

Đánh giá Khóa luận

1. Về cuốn báo cáo:

Số trang	_____	Số chương	_____
Số bảng số liệu	_____	Số hình vẽ	_____
Số tài liệu tham khảo	_____	Sản phẩm	_____

Một số nhận xét về hình thức cuốn báo cáo:

.....

.....

.....

.....

.....

.....

.....

2. Về nội dung nghiên cứu:

.....

.....

.....

.....

3. Về chương trình ứng dụng:

.....

.....

.....

.....

.....

.....

4. Về thái độ làm việc của sinh viên:

.....

.....

.....

.....

Đánh giá chung:

.....

Điểm từng sinh viên:

Trần Hữu Danh : /10

Người nhận xét

(Ký tên và ghi rõ họ tên)

ThS. Đinh Nguyễn Anh Dũng

ĐHQG TP. HỒ CHÍ MINH
TRƯỜNG ĐẠI HỌC
CÔNG NGHỆ THÔNG TIN

CỘNG HÒA XÃ HỘI CHỦ NGHĨA VIỆT NAM
Độc Lập - Tự Do - Hạnh Phúc

TP. HCM, ngày.....tháng.....năm.....

NHẬN XÉT KHÓA LUẬN TỐT NGHIỆP
(CỦA CÁN BỘ PHẢN BIỆN)

Tên khóa luận:

TÌM HIỂU CÔNG NGHỆ CONVERSATIONAL CHATBOT
ỨNG DỤNG TRONG XÂY DỰNG TEXT-BASED GAME

Nhóm SV thực hiện:

Cán bộ hướng dẫn/phản biện:

Trần Hữu Danh 12520054

Đánh giá Khóa luận

1. Về cuốn báo cáo:

Số trang _____ Số chương _____

Số bảng số liệu _____ Số hình vẽ _____

Số tài liệu tham khảo _____ Sản phẩm _____

Một số nhận xét về hình thức cuốn báo cáo:

.....

.....

.....

.....

.....

.....

.....

2. Về nội dung nghiên cứu:

.....

.....

.....

.....

.....

3. Về chương trình ứng dụng:

.....

.....

.....

.....

.....

4. Về thái độ làm việc của sinh viên:

.....

.....

.....

.....

.....

Đánh giá chung:

Điểm từng sinh viên:

Trần Hữu Danh : /10

Người nhận xét

(Ký tên và ghi rõ họ tên)

ĐỀ CƯƠNG CHI TIẾT

TÊN ĐỀ TÀI: “TÌM HIỂU CÔNG NGHỆ CONVERSATIONAL CHATBOT ỨNG DỤNG TRONG XÂY DỰNG TEXT-BASED GAME
Cán bộ hướng dẫn: ThS. Đinh Nguyễn Anh Dũng
Thời gian thực hiện: Từ ngày 01/09/2016 đến ngày 30/12/2016
Sinh viên thực hiện: Trần Hữu Danh - 12520054
Nội dung đề tài: Mục tiêu Nghiên cứu, tiếp cận công nghệ Conversational Chatbot Ứng dụng công nghệ Conversational Chatbot làm game Text-based Game Phạm vi Xây dựng Webhook trên nền tảng Facebook Messenger để ứng dụng phát triển Text-based Game Đối tượng Người dùng facebook, muốn chơi game thể loại giải đố chữ Đề tài còn làm tài liệu cho cộng đồng mã nguồn mở cùng phát triển Text-based Game trên Facebook messenger Phương pháp thực hiện Tìm hiểu, khái quát các nền tảng Conversational Chatbot đang có trên thị trường Tìm hiểu công nghệ python, flask, NLP để phát triển ứng dụng. Kết quả mong đợi Game chạy được ổn định

<p>Để tương tác với người chơi thông qua text</p> <p>Chatbot có thể xử lý nhiều trường hợp từ người dùng.</p> <p>Có thể tự tạo và custom nhiều map chơi khác nhau</p>		
Kế hoạch thực hiện:		
STT	Mô tả công việc	Người thực hiện
1	Tìm hiểu nền tảng, công nghệ Conversational Chatbot đang hiện có trên thế giới	Trần Hữu Danh
2	Phân tích ưu, khuyết điểm và lựa chọn công nghệ sử dụng cho đề tài	Trần Hữu Danh
3	Tìm hiểu Python 3.5, Flask, WebHook, Facebook Messenger Bot	Trần Hữu Danh
4	Viết Core Game	Trần Hữu Danh
5	Tìm hiểu NLP và ứng dụng trong Python	Trần Hữu Danh
6	Thiết kế cốt truyện Game	Trần Hữu Danh
7	Kiểm tra, hoàn thiện sản phẩm	Trần Hữu Danh
8	Cài đặt, kiểm thử	Trần Hữu Danh
9	Viết báo cáo	Trần Hữu Danh
<p>Xác nhận của CBHD</p> <p>(Ký tên và ghi rõ họ tên)</p> <p>ThS. Đinh Nguyễn Anh Dũng</p>		<p>TP.HCM, ngày 30 tháng 9 năm 2016</p> <p>Sinh viên</p> <p>(Ký tên và ghi rõ họ tên)</p> <p>Trần Hữu Danh</p>

MỤC LỤC

Chương 1.	TỔNG QUAN	3
1.1.	Lịch sử Conversational Chatbot	3
1.1.1.	Chatbot là gì?.....	3
1.1.2.	Sự phát triển của Chatbot	5
1.1.3.	Một số nền tảng Chatbot đang có trên thị trường.....	7
1.2.	Bối cảnh nghiên cứu	10
1.3.	Động lực nghiên cứu	11
1.4.	Giới hạn đề tài	12
Chương 2.	CƠ SỞ CÔNG NGHỆ	13
2.1.	Python	13
2.1.1.	Python là gì?	13
2.1.2.	Tại sao chọn Python?	13
2.1.3.	Một số tính chất của Python	14
2.2.	Flask.....	15
2.3.	Webhook.....	19
2.4.	Facebook Messenger Platform	20
2.5.	Natural Language ToolKit.....	20
2.5.1.	Natural Language ToolKit là gì?.....	20
2.5.2.	Why Natural Language ToolKit?	22
2.5.3.	ChatterBot.....	22
2.6.	JSON.....	24
Chương 3.	PHÂN TÍCH THIẾT KẾ	26
3.1.	Tổng quan chức năng	26

3.2.	Kiến trúc tổng quát	26
3.3.	Game pyZork	27
3.3.1.	Yêu cầu	27
3.3.2.	Cốt truyện	27
3.3.3.	GamePlay	28
3.3.4.	Phân tích, thiết kế Game.....	37
3.3.5.	Sơ đồ Flow Game	57
3.3.6.	Các cấu trúc dữ liệu trong Game	60
3.3.7.	Xử lý dữ liệu người chơi	61
3.3.8.	Xử lý ngôn ngữ tự nhiên	63
Chương 4.	CÀI ĐẶT MINH HỌA.....	65
4.1.	Yêu cầu	65
4.2.	Hướng dẫn cài đặt.....	65
4.3.	Cách chơi game	68
Chương 5.	KẾT LUẬN, HƯỚNG PHÁT TRIỂN	70
5.1.	Kết luận.....	70
5.1.1.	Đánh giá chung.....	70
5.1.2.	Bài học.....	70
5.2.	Hướng phát triển.....	71
	TÀI LIỆU THAM KHẢO.....	72

DANH MỤC HÌNH VẼ

Hình vẽ 1.1: Chatbot là gì?	3
Hình vẽ 1.2: Mô tả ví dụ Chatbot hỗ trợ kiểm tra tình hình giao thông	5
Hình vẽ 1.3: Facebook Chatbot.....	8
Hình vẽ 1.4: Skype Chatbot	9
Hình vẽ 1.5: Telegram Chatbot.....	10
Hình vẽ 1.6: Tầm ảnh hưởng của ChatBot so với ứng dụng khác	11
Hình vẽ 2.1: Code của Python đơn giản, gần với ngôn ngữ nói.	14
Hình vẽ 2.2: Hello World with Flask.....	15
Hình vẽ 2.3: Routing in Flask	15
Hình vẽ 2.4: Variable Rules in Flask	16
Hình vẽ 2.5: Unique URLs / Redirection Behavior in Flask	16
Hình vẽ 2.6: URL Building in Flask	16
Hình vẽ 2.7: HTTP Methods in Flask	17
Hình vẽ 2.8: Request Object in Flask	17
Hình vẽ 2.9: File Upload in Flask	17
Hình vẽ 2.10: File Upload bảo mật trong Flask.....	18
Hình vẽ 2.11: Reading Cookie in Flask	18
Hình vẽ 2.12: Strong cookie in Flask.....	18
Hình vẽ 2.13: Redirect in Flask.....	18
Hình vẽ 2.14: Error Handle in Flask	18
Hình vẽ 2.15: Session in Flask.....	19
Hình vẽ 2.16: Logging in Flask.....	19
Hình vẽ 2.17: Facebook Messenger cho Facebook developer.....	20
Hình vẽ 2.18: NLTK hỗ trợ Tokenize và tách các tag từ text input	21
Hình vẽ 2.19: Entities trong NLTK	21
Hình vẽ 2.20: Parse tree trong NLTK	22
Hình vẽ 2.21: Thư viện ChatterBot.....	23
Hình vẽ 2.22: Quy trình hoạt động của ChatterBot	24

Hình vẽ 3.1: Tổng quan kiến trúc game pyZork	26
Hình vẽ 3.2: CREATE command trong pyZork	29
Hình vẽ 3.3: Danh sách các command hệ thống trong pyZork.....	30
Hình vẽ 3.4: Command lấy thông tin hiện tại của người chơi	30
Hình vẽ 3.5: Inventory rỗng	31
Hình vẽ 3.6: Lấy thông tin về kho đồ của người dùng.	31
Hình vẽ 3.7: Hướng dẫn chơi game pyZork	32
Hình vẽ 3.8: Reset pyZork data.....	32
Hình vẽ 3.9: Người dùng nhập text để di chuyển trong pyZork	33
Hình vẽ 3.10: Dùng các động từ thường để dễ dàng chơi game hơn.....	34
Hình vẽ 3.11: Yêu cầu item để được đi tiếp.....	35
Hình vẽ 3.12: Nhặt vật phẩm yêu cầu để được qua màn	36
Hình vẽ 3.13: Người chơi được đi tiếp sau khi nhặt được vật phẩm yêu cầu.....	37
Hình vẽ 3.14: Mô tả ngữ hành trong pyZork	39
Hình vẽ 3.15: Sơ đồ màn đầu trong pyZork.....	43
Hình vẽ 3.16: Mở bức thư để lấy thông tin.....	44
Hình vẽ 3.17: Cần tìm ra chiếc chìa khóa để mở cửa ra ngoài.	44
Hình vẽ 3.18: Đọc nội dung trong bức thư giới thiệu game	45
Hình vẽ 3.19: Ra bên ngoài ngôi nhà, bắt đầu hành trình.....	45
Hình vẽ 3.20: đi về khu rừng hướng đông	46
Hình vẽ 3.21: Tìm thợ rèn trong ngôi làng	46
Hình vẽ 3.22: Sơ đồ bên trong khu rừng.....	46
Hình vẽ 3.23: Đi từ ngôi làng vào rừng	47
Hình vẽ 3.24: Tìm đến cây đại thụ đang có Wood Treasure	47
Hình vẽ 3.25: Leo lên cây và lấy Wood Treasure.....	48
Hình vẽ 3.26: Kiểm tra trạng thái hiện tại.....	48
Hình vẽ 3.27: Đến bờ sông hướng đông	49
Hình vẽ 3.28: Sơ đồ tại bờ sông.....	49
Hình vẽ 3.29: Cách tìm thấy được Water Treasure và đi đến núi lửa.....	50

Hình vẽ 3.30: Sơ đồ tại núi lửa	50
Hình vẽ 3.31: Cách lấy gem trên núi lửa	51
Hình vẽ 3.32: Đi tìm thợ rèn sau khi có Fire Treasure	51
Hình vẽ 3.33: Sơ đồ tìm thợ rèn.....	52
Hình vẽ 3.34: Rèn thanh kiếm giết rồng bằng Fire Treasure.....	52
Hình vẽ 3.35: Về hướng nam để tìm hang rồng.....	52
Hình vẽ 3.36: Sơ đồ tại hang rồng	53
Hình vẽ 3.37: Cách tiêu diệt rồng	53
Hình vẽ 3.38: Tiêu diệt rồng để lấy Metal Treasure	54
Hình vẽ 3.39: Sơ đồ tại thung lũng	54
Hình vẽ 3.40: Theo hướng Tây Nam để tìm thấy thung lũng.....	55
Hình vẽ 3.41: Xuống đáy thung lũng để tìm Earth Treasure	55
Hình vẽ 3.42: Đào xuống lòng đất để tìm Earth Treasure	55
Hình vẽ 3.43: Kiểm tra đủ số lượng 5 Treasure cần tìm.....	56
Hình vẽ 3.44: Sơ đồ tại đền thờ.....	56
Hình vẽ 3.45: Sử dụng 5 Treasures tìm được để chiến thắng game	56
Hình vẽ 3.46: Sử dụng @hardreset để chơi lại	57
Hình vẽ 3.47:Flow của pyZork	58
Hình vẽ 3.48: Cách tạo ra danh sách scene trong Game.....	61
Hình vẽ 3.49: Lưu thông tin default scene.....	62
Hình vẽ 3.50: Tạo danh sách các Item trong Game	62
Hình vẽ 3.51: Lấy ra thông tin default scene trong Game.	62
Hình vẽ 3.52: Tạo thông tin người chơi với định dạng json.....	63
Hình vẽ 3.53: Xử lý thông tin người chơi lúc chuyển cảnh.....	63
Hình vẽ 3.54: Tạo một ChatterBot trong source code	64
Hình vẽ 3.55: Cách nhận lại response cho người chơi.....	64
Hình vẽ 4.1: Tạo ứng dụng facebook.....	65
Hình vẽ 4.2: Chọn add products trong facebook app.....	66
Hình vẽ 4.3: Box Token Generation	66

Hình vẽ 4.4: Chọn Fanpage và copy token	66
Hình vẽ 4.5: Add to submission facebook messenger	67
Hình vẽ 4.6: Dán token đã copy vào giá trị biến ACCESS_TOKEN.....	67
Hình vẽ 4.7: ngrok http 5000	67
Hình vẽ 4.8: Config webhook	68

DANH MỤC BẢNG

Bảng 3.1: Một số command trong pyZork.....	28
Bảng 3.2: Các thuộc tính bí ẩn cần tìm trong pyZork.....	42
Bảng 3.3: Thiết kế các khung cảnh lớn trong pyZork.....	43

TÓM TẮT KHÓA LUẬN

Công nghệ phát triển hàng ngày, mỗi ngày có hàng ngàn ứng dụng được đưa lên các chợ ứng dụng: Google Play, App Store, Tizen Store.... Dường như khi nhắc đến giải pháp cho một vấn đề và người ta nghĩ ngay đến ứng dụng, điều đó làm cho mọi chuyện trở nên rắc rối và bão hòa. Công nghệ Conversational Chatbot hình thành để cứu rỗi điều đó. Minh chứng cho thấy Conversational Chatbot là một công nghệ sẽ kế thừa kỷ nguyên ứng dụng (kỷ nguyên hậu ứng dụng) khi giám đốc của Facebook Messenger đã nói rằng: “Hôm nay có thể là sự khởi đầu của một kỷ nguyên mới!” trong buổi ra mắt Facebook Bot Messenger.

Conversational Chatbot ra đời như một cứu cánh cho sự bão hòa về ứng dụng, đây là một mảnh đất đang được các ông lớn trong lĩnh vực công nghệ: Facebook, Google, Apple... ném hàng tỷ đô vào để phát triển với những ưu tiên hàng đầu.

Từ những kết quả khảo sát sơ bộ trên, em đã quyết định chọn đề tài “Tìm hiểu Conversational Chatbot ứng dụng trong xây dựng text-based game” làm đề tài khóa luận.

Quá trình tìm hiểu, nghiên cứu về Conversational Chatbot bao gồm các vấn đề cơ bản: Định nghĩa lại Conversational Chatbot là gì, so sánh các platform hỗ trợ Conversational Chatbot đang phổ biến trên thế giới, chọn nền tảng để phát triển text-based game. Tiếp theo cần tìm hiểu các vấn đề chuyên sâu: tìm hiểu ngôn ngữ python 3.5, Flask Framework, Webhook, Facebook bot Messenger. Cuối cùng là việc áp dụng các công nghệ, kiến thức đã tìm hiểu được xây dựng text-based game.

Kết quả đề tài là một text-based game hoàn chỉnh với thể loại phiêu lưu truy tìm báu vật. Người chơi với nhiệm vụ chính của mình trong thế giới game là truy tìm 5 báu vật và mang chúng đến điện thờ để chiến thắng.

MỞ ĐẦU

Về bản chất, Chatbot đơn giản chỉ là một chương trình xử lý text bình thường. Chatbot đã xuất hiện từ rất lâu và vẫn được dùng hàng ngày trong một số ứng dụng xung quanh ta. Ngay từ thời lập trình vừa bắt đầu, chương trình máy tính đầu tiên được chạy ta có thể gọi đó là một chatbot được rồi. Tuy nhiên, Chatbot thời đó chưa thể thông minh nhiều, chưa thể đối đáp và giải quyết được nhiều vấn đề linh hoạt như ngày nay.

Trong nhiều năm trở lại đây, cuộc chạy đua trợ lý ảo của các hãng công nghệ lớn làm cho sự nóng của Chatbot trở lại, có thể kể tiêu biểu các trợ lý ảo lớn: Google Assistant, Apple Siri, Microsoft Cortana. Chatbot giờ đã có thể trò chuyện với con người bằng cả âm thanh, giúp con người đặt vé máy bay, đặt taxi hoặc đơn giản chỉ là hát cho chủ nhân nghe. Sự bùng nổ này của Chatbot là một điều tất yếu khi mà sự tăng trưởng của mảng ứng dụng di động toàn cầu đang dần trỗi lại, thì Chatbot là một lối thoát cho sự hiếu kỳ, luôn hứng thú của loài người.

Trong đề tài này ta sẽ cùng đi qua về các định nghĩa: Chatbot, các Chatbot lớn đang có trên thị trường, tìm hiểu về công nghệ xây dựng Chatbot trên Facebook Messenger và chạy Game pyZork hoàn thiện trên nền tảng Facebook Bot Messenger sử dụng Python 3.5 trên framework Flask.

Chương 1. TỔNG QUAN

1.1. Lịch sử Conversational Chatbot

1.1.1. Chatbot là gì?



Hình vẽ 1.1: Chatbot là gì?

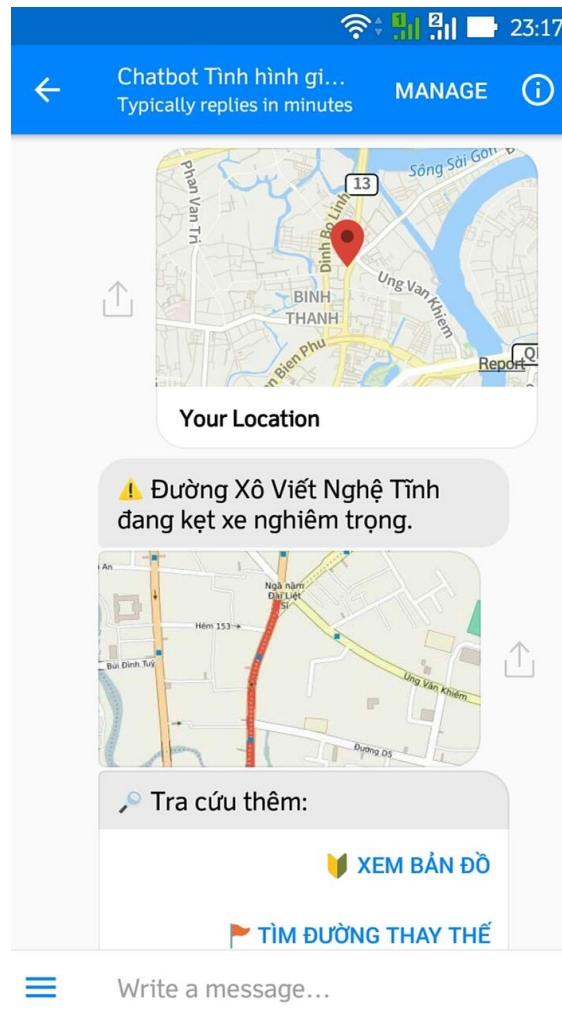
Một cách đơn giản để hiểu thì **Chatbot** là một phần của phần mềm. Ở đó bạn giao tiếp với hệ thống bằng các dòng chat hoặc câu nói để thực hiện 1 tác vụ nào đó hay đơn giản là giải trí.

Chatbot như một sự thay thế cho các ứng dụng mà bạn tải xuống từ các store (Google Play, Apple Store). Thay vì mở ứng dụng lên để xem nhiệt độ, bạn có thể hỏi **Chatbot** này và sẽ nhận được kết quả thời tiết của địa điểm bạn mong

muốn bằng cách hỏi chúng: “Thời tiết hôm nay ở Trường Đại Học Công Nghệ Thông Tin thế nào?”.

Một **Chatbot** có thể giúp người dùng giải quyết mọi thứ như một người trợ lý ảo. Ví dụ như việc đặt một chiếc Uber đi từ địa điểm ‘nhà C trường Đại Học Công Nghệ Thông Tin’ đến ‘Nhà Điều Hành Đại Học Quốc Gia Thành phố Hồ Chí Minh’, hoặc việc ‘thiết lập một cuộc họp tại Văn phòng khoa Công Nghệ Phần Mềm’. Hay đơn giản là việc tâm sự về các vấn đề trong cuộc sống với trí thông minh nhân tạo có bên trong Chatbot (Nói chuyện với Siri, Cortana, Google Assistant, Simsimi. ..)

Ví dụ một Chatbot với chức năng ‘thông báo trạng thái giao thông nơi công cộng’ có thể cho bạn biết rằng vào giờ này trên đường từ công ty về nhà bạn đang kẹt xe bằng cách gửi notification đến chiếc điện thoại của bạn. Chatbot khuyên bạn nên ở lại công ty thêm thời gian để chờ qua giờ cao điểm, hoặc gợi ý cho bạn một con đường khác để được về nhà mà không gặp phải tình trạng kẹt xe.



Hình vẽ 1.2: Mô tả ví dụ Chatbot hỗ trợ kiểm tra tình hình giao thông

1.1.2. Sự phát triển của Chatbot

Chatbot có một sự phát triển khá lâu dài và xuyên suốt quá trình phát triển của ngành Công Nghệ Thông Tin của loài người. Từ những năm 1960 người ta đã tạo ra các Chatbot với mục đích giải trí.

1966 ELIZA: Một Chatbot bắt chước các đoạn hội thoại của con người thông qua các chỉ định có sẵn. Chatbot này đã vượt qua một vài bài test trí tuệ nhân tạo đơn giản thời ấy.

1972 PARRY: PARRY giống như một phiên bản với nhiều độ phức tạp và thông minh hơn ELIZA. PARRY đã biết sử dụng các biểu cảm tốt hơn so với ELIZA

1988 JABBERWACKY: Một trong những bước đi đầu của việc cho AI học tập các hành vi của con người trong giao tiếp. Chủ yếu phát triển hệ thống hỗ trợ không còn dùng text-based mà chuyển sang xử lý thông qua giọng nói của người dùng.

1992 DR. SBAITSO: Một trí thông minh nhân tạo được tổng hợp để hỗ trợ MS DOS-based PCs. Được thiết kế để điều khiển bằng giọng nói.

1995 A.L.I.C.E: là viết tắt của tổ hợp từ “Artificial Linguistic Internet Computer Entity”. A.L.I.C.E là một bộ xử lý ngôn ngữ tự nhiên, ALICE có thể áp dụng các đoạn hội thoại của người dùng và học chúng để xử lý cho các đoạn hội thoại sau đó.

2001 SMARTERCHILD: Một AI được phân bố rộng rãi trong mạng tin nhắn SMS. Chức năng chính của SMARTERCHILD là truy cập dữ liệu nhanh và giao tiếp vui vẻ với con người. AI này được xem xét để ứng dụng cho Apple Siri và Samsung S Voice.

2006 IBM’S WATSON: Watson là một AI đặc dụng, sử dụng hệ thống xử lý ngôn ngữ tự nhiên và máy học để thao tác với các dữ liệu lớn. Từ đó thuận tiện trong việc giao tiếp con người

2010 SIRI: Một AI với nhiệm vụ làm trợ lý ảo, một phần của iOS với tính năng chính là trả lời các câu hỏi, làm một số thao tác với điện thoại và cung cấp các trang web được tìm kiếm theo từ khóa.

2012 GOOGLE NOW: Được phát triển bởi Google áp dụng trong việc hỗ trợ người dùng tìm kiếm trên di động, Google Now ra đời như một sự cạnh tranh với Siri của Apple về sự thông minh, tính hài hước và khả năng gắn gũi với giao tiếp con người

2015 ALEXA: Một AI với khả năng tương tác tiếng nói với con người. Công nghệ trong ALEXA là xử lý ngôn ngữ tự nhiên, phát hiện và xử lý âm thanh, máy học, phát âm từ text

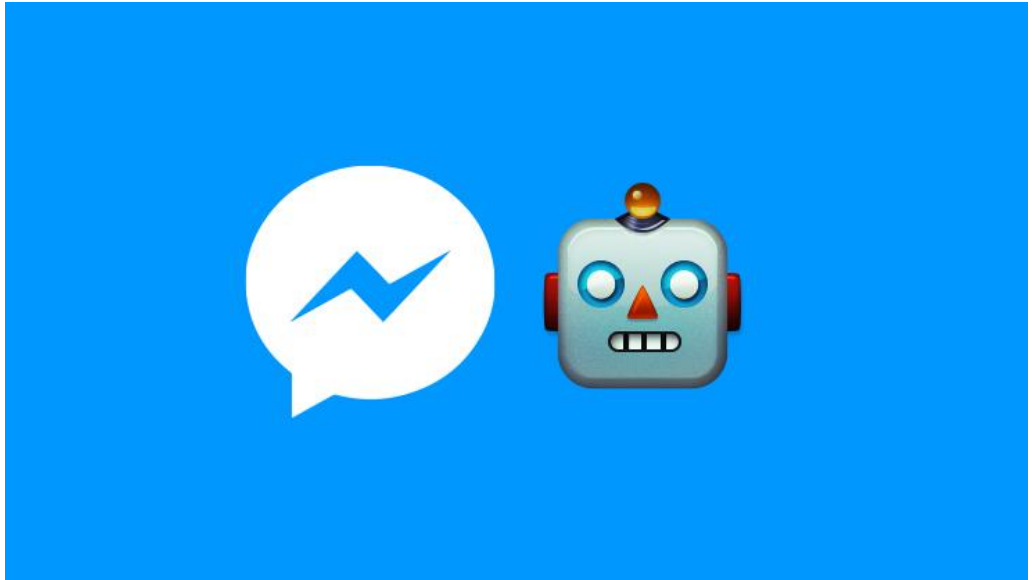
2015 CORTANA: Một AI làm trợ lý ảo cá nhân cho người dùng Windows, Cortana hỗ trợ chúng ta đặt lịch, thông báo, giao tiếp về một số câu hỏi liên quan đến tri thức nhân loại, trả lời câu hỏi thông qua tìm kiếm thông tin trên Bing.

2016 BOTS FOR MESSENGER: Tháng 4 năm 2016, Facebook ra đời nền tảng Facebook Bot Messenger cho phép các nhà phát triển trên toàn thế giới tạo ra Chatbot với những chức năng đa dạng theo mỗi nhà phát triển. Ngay sau đó vào tháng 7 cùng năm, đã có 11 nghìn Facebook Bot Messengers được đưa vào sử dụng trên Facebook.

Tóm lại, Chatbot vẫn và đang được phát triển, hoàn thiện hằng ngày. Trong tương lai, Chatbot sẽ bắt đầu được đưa vào sử dụng ở các hệ thống bán hàng, hỏi đáp, chăm sóc khách hàng rộng rãi trên nền tảng Facebook Messenger.

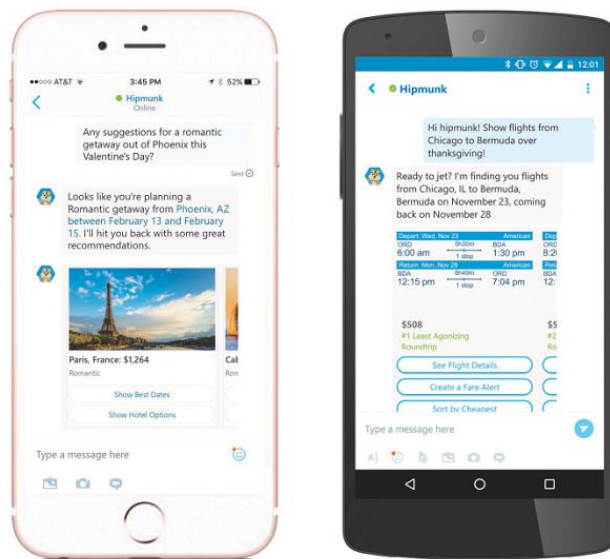
1.1.3. Một số nền tảng Chatbot đang có trên thị trường

Facebook Messenger: Facebook đã mở nền tảng Messenger Platform trong khoảng tháng 4 năm 2016 để kích hoạt Bot giao tiếp với người dùng thông qua ứng dụng Facebook Apps và Facebook Pages. Hiện tại Facebook Messenger là một nền tảng có đông người dùng active nhất thế giới. Động thái này của Facebook cho thấy họ muốn đưa Chatbot của Facebook Messenger lên đứng đầu trong cuộc đua công nghệ Conversational Chatbot này.



Hình vẽ 1.3: Facebook Chatbot

Skype: Sau Facebook Messenger và một số dịch vụ nhắn tin khác, Skype chính là cái tên phổ biến tiếp theo ra mắt chính thức tính năng chatbot dành cho dịch vụ nhắn tin của mình, một thời gian ngắn sau khi triển khai phiên bản thử nghiệm Skype Bots hồi tháng 4. Mục đích và chức năng của Skype Bots cũng giống như Facebook Messenger. Nó cho phép các doanh nghiệp và cửa hàng thực hiện hỗ trợ, giải quyết những thắc mắc một cách nhanh chóng nhất cho các khách hàng của mình. Bên cạnh đó, người dùng cũng có thể tích hợp chatbot vào trong cuộc trò chuyện nhóm và nó sẽ quản lý các công việc diễn ra với mọi người hoặc gia đình, cũng như giúp thực hiện một số nhiệm vụ khác. Ví dụ như bạn có thể dùng Hipmunk để xem thông tin chuyến bay hoặc tìm ra những địa điểm du lịch phù hợp với nhu cầu của bản thân. Skype cũng đã hợp tác với các dịch vụ như StubHub hay Skyscanner giúp mang lại các tính năng bổ sung cho Skype Bots, cũng như IFTTT Bot cho phép bạn tạo ra chatbot của riêng mình với các tính năng cơ bản như nhận thông báo đến từ mạng xã hội, hoặc cảnh báo khi có người đến nhà.



Hình vẽ 1.4: Skype Chatbot

Telegram: Telegram là một mã nguồn mở về Chatbot. Telegram cung cấp các API hỗ trợ người lập trình viên viết các Chatbot của họ ngay trên nền tảng Telegram. Để sử dụng Chatbot trong telegram ta chỉ cần tải app về, đăng ký tài khoản telegram và bắt đầu sử dụng. Trên telegram bạn có thể tìm được rất nhiều Chatbot khác nhau: dạy học, chơi game, nhắc nhở, tìm kiếm.... Rất nhiều tính năng đòi thường được hiện thực hóa trong Chatbot và đưa lên Telegram.



Hình vẽ 1.5: Telegram Chatbot

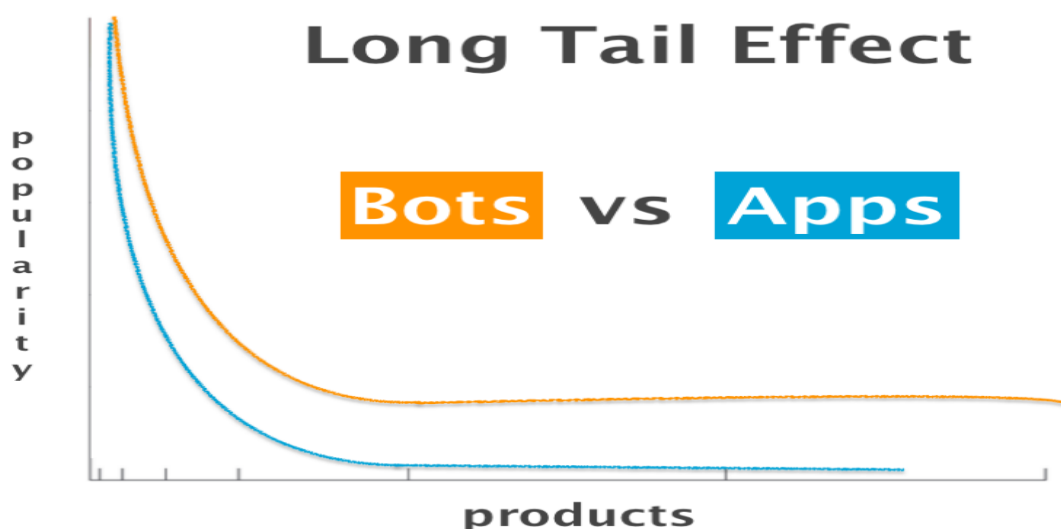
1.2. Bối cảnh nghiên cứu

Sự bùng nổ và phát triển quá mạnh mẽ của các Store bán ứng dụng trên các nền tảng di động, điều đó đồng nghĩa với việc các ứng dụng được đưa lên ngày càng nhiều và thật sự rắc rối, khó chọn lọc được ứng dụng tốt để người dùng có thể trải nghiệm ngay.

Sự rắc rối và phức tạp trong giao diện của từng ứng dụng khiến người dùng chần chừ gặp rắc rối khi thao tác và làm quen sử dụng với một ứng dụng mới.

Một số ứng dụng yêu cầu đăng nhập, và làm quá nhiều thủ tục thừa chỉ để được sử dụng chức năng chính. Điều này hoàn toàn là không thực sự cần thiết cho người dùng.

Người ta thường thay đổi App khá nhanh, nhưng rất ít khi và hầu như không bao giờ gỡ bỏ app Facebook Messenger. Điều này hỗ trợ cho việc Chatbot luôn có đất sống trên ứng dụng Facebook Messenger.



Hình vẽ 1.6: Tầm ảnh hưởng của ChatBot so với ứng dụng khác

Facebook thấu hiểu các khó khăn trên đối với người dùng, đã tạo ra nền tảng Facebook Bot Messenger. Cung cấp các API giúp lập trình viên có thể thoải mái tạo các Chatbot phục vụ các tính năng yêu cầu đơn giản cho riêng của mình.

1.3. Động lực nghiên cứu

Conversational Chatbot không phải là một xu hướng mới, nhưng đây là một mảnh đất được phát hiện và đầu tư mạnh trong 2 năm trở lại của nhiều nền tảng lớn trên thế giới: Facebook, Telegram, Skype, Slack, Google, Apple ... Điều này chứng tỏ tiềm lực không nhỏ từ Chatbot sẽ giúp con người giải quyết nhiều vấn đề trong cuộc sống.

Converstational Chatbot không chỉ đơn giản là text mà còn có thể là hình ảnh, âm thanh, là địa điểm. Ta có thể mua bán, quảng cáo và chơi game ngay trên ứng dụng Messenger của mình.

Chatbot có thể được sử dụng online mọi lúc, mọi nơi thông qua smartphone có cài đặt ứng dụng Chat. Ta dễ nhận thấy sự tiện dụng của Chatbot, khi mà phải cài hàng chục app để làm các công việc: Mua vé xem phim, đặt vé máy bay, tìm hiểu kiến thức, mua quần áo hay đơn giản chỉ là trò chuyện, giải trí. Chỉ cần có ứng dụng Chat ta có thể sử dụng mọi dịch vụ này mà không cần cài thêm một thành phần nào nữa.

Người dùng có thể cài một app và gỡ chúng ngay vài ngày sau đó. Tuy nhiên với những ứng dụng Messenger thì khác, người dùng có tính trung thành và đa dạng cao. Tỷ lệ gỡ bỏ và ngưng sử dụng Messenger ở người dùng di động là thấp hơn nhiều so với các ứng dụng, game khác. Điều này làm tăng số lượng người được tiếp cận với các Chatbot, khiến Chatbot có một lượng user tiềm năng cao.

Làm game trên **Converstational Chatbot** là một thử thách đáng để tham gia, học hỏi và tích lũy cho bản thân nhiều kinh nghiệm làm Game cho nền tảng mới. Giúp đa dạng vốn kiến thức, sẵn sàng nghiên cứu và phát triển ứng dụng đón đầu xu thế.

Từ những điều trên thúc đẩy bản thân em chọn đề tài Converstational Chatbot và thực hiện.

1.4. Giới hạn đề tài

- Nghiên cứu, tiếp cận công nghệ Converstational Chatbot.
- Tìm hiểu Python 3.5, Framework Flask
- Tìm hiểu, sử dụng Natural Language ToolKit, Chatterbot trong game

Chương 2. CƠ SỞ CÔNG NGHỆ

2.1. Python

2.1.1. Python là gì?

Python được phát triển bởi Guido Van Rossum và cuối những năm 80 và đầu những năm 90 tại viện Toán-Tin ở Hà Lan. Python có kế thừa nhiều ngôn ngữ như: ABC, Module-3, C, C++, Unix Shell, ..

Ngôn ngữ Python được cập nhật khá thường xuyên để thêm các tính năng hỗ trợ mới. Phiên bản mới nhất hiện nay của Python là Python 3.5.2 được công bố vào ngày 27 tháng 6 năm 2016.

Python là ngôn ngữ lập trình bậc cao, thông dịch, hướng đối tượng và là một ngôn ngữ lập trình động.

Cú pháp trong Python rất linh hoạt. Python hỗ trợ mẫu đa lập trình, bao gồm lập trình hướng đối tượng, lập trình hàm và mệnh lệnh hoặc là các phong cách lập trình theo thủ tục.

Python không chỉ làm việc trên các lĩnh vực riêng như web, mà rộng ra tất cả các lĩnh vực: client, server và cả IoT

Trong Python người lập trình viên có thể khai báo $a = 1$ để biểu thị biến số nguyên giá trị 1 và ngay sau đó định nghĩa lại $a = 'a'$ mà không cần một phép ép kiểu tường minh nào cả. Với Python, việc phát triển ứng dụng và debug trở nên nhanh hơn bởi vì không cần đến quá trình build, biên dịch vì chu trình edit-test-debug của Python diễn ra rất nhanh.

2.1.2. Tại sao chọn Python?

Cú pháp Python dễ đọc: Python có điểm chặt chẽ rất giống với ngôn ngữ tiếng Anh, sử dụng những từ 'not', 'in' nên khi viết một chương trình hoặc đoạn script Python người lập trình viên sẽ có cảm giác như là đang nói. Mã nguồn của Python là tương đối dễ để bảo trì và duy trì và có khả năng mở rộng.

<pre> 1 #!/usr/bin/python 2 3 print "Hello, World!"; 4 </pre> <p>"Hello, World!" program in Python</p>	<pre> 1 #include <stdio.h> 2 3 int main() 4 { 5 printf("Hello, World! \n"); 6 return 0; 7 } 8 </pre> <p>"Hello, World!" program in C</p>
--	--

Hình vẽ 2.1: Code của Python đơn giản, gần với ngôn ngữ nói.

Các thư viện trong Python phong phú: Python đã tồn tại được 20 năm, vì vậy có rất nhiều code viết bằng Python được xây dựng từ nhiều thập kỷ. Đây là một ngôn ngữ mã nguồn mở, nên được cộng đồng đóng góp và xây dựng rất nhiều. Các thư viện có thể được tìm thấy tại '<https://pypi.python.org>'.

Python dễ dàng tích hợp: Python cho phép người dùng tích hợp vào các module để có thể sử dụng trong các chương trình khác. Có thể dễ dàng tích hợp với C, C++, COM, CORBA, ActiveX, Java.

Cộng đồng người dùng lớn: Python có cộng đồng người dùng ở khắp mọi nơi, và hỗ trợ nhiệt tình trong quá trình xây dựng và phát triển ứng dụng.

2.1.3. Một số tính chất của Python

Python is Interpreted: Nhờ chức năng thông dịch mà trình thông dịch (Interpreter) của Python có thể xử lý lệnh tại thời điểm chạy chương trình (runtime). Nhờ đó mà ta không cần biên dịch chương trình trước khi thực hiện nó (tương tự như Perl và PHP).

Python is Interactive: Tính năng tương tác của Python giúp ta có thể tương tác trực tiếp với trình thông dịch của nó ngay tại dấu nhắc lệnh. Cụ thể: Ta có thể thực hiện lệnh một cách trực tiếp tại dấu nhắc của Python.

Python is Object-Oriented: Python hỗ trợ mạnh cho phong cách lập trình hướng đối tượng và kỹ thuật lập trình gói mã trong đối tượng.

Python is a Beginner's Language: Mặc dầu Python được xem là ngôn ngữ lập trình dành cho những ai mới làm quen với việc lập trình trên máy tính, nhưng nó hỗ trợ mạnh cho việc phát triển nhiều loại ứng dụng khác nhau, từ các chương trình xử lý văn bản đơn giản đến các ứng dụng web, đến các chương trình game,...

2.2. Flask

Flask là một web framework mã nguồn mở viết trên nền Python. Được cấp chứng chỉ BSD. Hiện tại Flask có phiên bản mới nhất là 0.12

Flask cung cấp cho lập trình viên các công cụ, thư viện và công nghệ cho phép người ta xây dựng các ứng dụng web một cách tiện lợi và nhanh chóng nhất.

Hello World with Flask

```
from flask import Flask
app = Flask(__name__)

@app.route('/')
def hello_world():
    return 'Hello, World!'
```

Hình vẽ 2.2: Hello World with Flask

Routing in Flask

```
@app.route('/')
def index():
    return 'Index Page'

@app.route('/hello')
def hello():
    return 'Hello, World'
```

Hình vẽ 2.3: Routing in Flask

Variable Rules


```

@app.route('/user/<username>')
def show_user_profile(username):
    # show the user profile for that user
    return 'User %s' % username

@app.route('/post/<int:post_id>')
def show_post(post_id):
    # show the post with the given id, the id is an integer
    return 'Post %d' % post_id

```

Hình vẽ 2.4: Variable Rules in Flask

Unique URLs / Redirection Behavior

```

@app.route('/projects/')
def projects():
    return 'The project page'

@app.route('/about')
def about():
    return 'The about page'

```

Hình vẽ 2.5: Unique URLs / Redirection Behavior in Flask

URL Building

```

>>> from flask import Flask, url_for
>>> app = Flask(__name__)
>>> @app.route('/')
... def index(): pass
...
>>> @app.route('/login')
... def login(): pass
...
>>> @app.route('/user/<username>')
... def profile(username): pass
...
>>> with app.test_request_context():
...     print url_for('index')
...     print url_for('login')
...     print url_for('login', next='/')
...     print url_for('profile', username='John Doe')
...
/
/login
/login?next=/
/user/John%20Doe

```

Hình vẽ 2.6: URL Building in Flask

HTTP Methods

```

from flask import request

@app.route('/login', methods=['GET', 'POST'])
def login():
    if request.method == 'POST':
        do_the_login()
    else:
        show_the_login_form()

```

Hình vẽ 2.7: HTTP Methods in Flask

The Request Object

```

@app.route('/login', methods=['POST', 'GET'])
def login():
    error = None
    if request.method == 'POST':
        if valid_login(request.form['username'],
                        request.form['password']):
            return log_the_user_in(request.form['username'])
        else:
            error = 'Invalid username/password'
    # the code below is executed if the request method
    # was GET or the credentials were invalid
    return render_template('login.html', error=error)

```

Hình vẽ 2.8: Request Object in Flask

File Uploads

```

from flask import request

@app.route('/upload', methods=['GET', 'POST'])
def upload_file():
    if request.method == 'POST':
        f = request.files['the_file']
        f.save('/var/www/uploads/uploaded_file.txt')
    ...

```

Hình vẽ 2.9: File Upload in Flask

```

from flask import request
from werkzeug.utils import secure_filename

@app.route('/upload', methods=['GET', 'POST'])
def upload_file():
    if request.method == 'POST':
        f = request.files['the_file']
        f.save('/var/www/uploads/' + secure_filename(f.filename))
    ...

```

Hình vẽ 2.10: File Upload bảo mật trong Flask

Cookies

```
from flask import request

@app.route('/')
def index():
    username = request.cookies.get('username')
    # use cookies.get(key) instead of cookies[key] to not get a
    # KeyError if the cookie is missing.
```

Hình vẽ 2.11: Reading Cookie in Flask

```
from flask import make_response

@app.route('/')
def index():
    resp = make_response(render_template(...))
    resp.set_cookie('username', 'the username')
    return resp
```

Hình vẽ 2.12: Strong cookie in Flask

Redirects and Errors

```
from flask import abort, redirect, url_for

@app.route('/')
def index():
    return redirect(url_for('login'))

@app.route('/login')
def login():
    abort(401)
    this_is_never_executed()
```

Hình vẽ 2.13: Redirect in Flask

```
from flask import render_template

@app.errorhandler(404)
def page_not_found(error):
    return render_template('page_not_found.html'), 404
```

Hình vẽ 2.14: Error Handle in Flask

Sessions

```
from flask import Flask, session, redirect, url_for, escape, request

app = Flask(__name__)

@app.route('/')
def index():
    if 'username' in session:
        return 'Logged in as %s' % escape(session['username'])
    return 'You are not logged in'

@app.route('/login', methods=['GET', 'POST'])
def login():
    if request.method == 'POST':
        session['username'] = request.form['username']
        return redirect(url_for('index'))
    return '''
        <form method="post">
            <p><input type="text" name="username">
            <p><input type="submit" value="Login">
        </form>
    '''

@app.route('/logout')
def logout():
    # remove the username from the session if it's there
    session.pop('username', None)
    return redirect(url_for('index'))

# set the secret key. keep this really secret:
app.secret_key = 'A0Zr98j/3yX R~XHH!jmN]LWX/,?RT'
```

Hình vẽ 2.15: Session in Flask

Logging

```
app.logger.debug('A value for debugging')
app.logger.warning('A warning occurred (%d apples)', 42)
app.logger.error('An error occurred')
```

Hình vẽ 2.16: Logging in Flask

2.3. Webhook

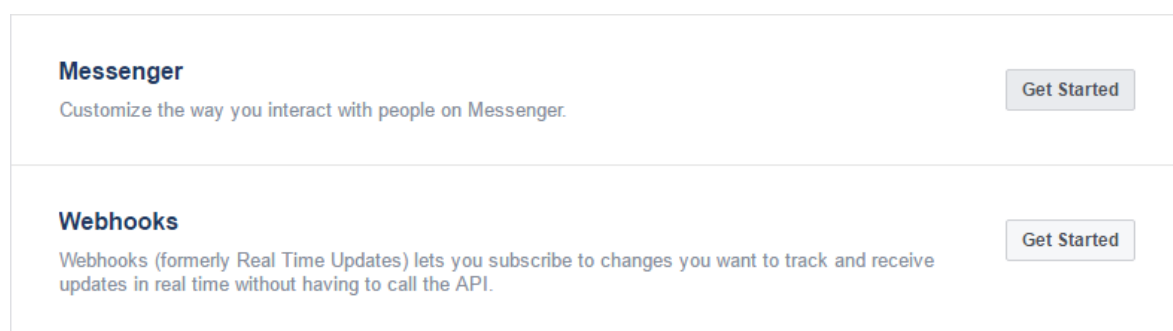
Webhook trong lập trình web là một phương pháp tùy chỉnh hành vi của một trang web với callback tương ứng. Những callback này sẽ được một bên thứ ba

mà không nhất thiết phải có được liên kết với trang web hoặc ứng dụng quản lý. Thuật ngữ này xuất hiện vào năm 2007 và đặt tên bởi Jeff Lindsay.

Trong Facebook Messenger, Webhook ứng dụng là cầu nối trung chuyển xử lý tin nhắn giữa người dùng và Chatbot. Phía người phát triển Chatbot có thể tùy ý xây dựng các logic xử lý trên Bot miễn thỏa mãn yêu cầu của Webhook phía facebook Messenger là cung cấp link Webhook và private token để truy cập webhook.

2.4. Facebook Messenger Platform

Facebook cung cấp Messenger Platform giúp lập trình viên xây dựng các Chatbot với mục đích của họ. Những Chatbot này được tạo ra tùy biến 100% theo nhu cầu cụ thể của nhà phát triển. Có thể là game, app hoặc đơn giản là một bot để trò chuyện phiếm giết thời gian.



Hình vẽ 2.17: Facebook Messenger cho Facebook developer

2.5. Natural Language ToolKit

2.5.1. Natural Language ToolKit là gì?

Natural Language ToolKit là một nền tảng hàng đầu trong việc xây dựng và phát triển các chương trình Python để làm việc với ngôn ngữ loài người. Natural Language ToolKit cung cấp hơn 50 corpora và các nguồn resource từ WordNet, các bộ xử lý văn bản để phân loại, tokenization, gán thẻ, phân tích, lý luận ngữ nghĩa...

Nhờ có nhiều hướng dẫn, tài liệu về API đầy đủ. Natural Language ToolKit phù hợp cho người sử dụng là các nhà lập trình ngôn ngữ học, sinh viên, kỹ sư trong việc giải quyết vấn đề xử lý ngôn ngữ tự nhiên.

Natural Language ToolKit là một công cụ tuyệt hảo cho việc nghiên cứu, giảng dạy trong việc xử lý ngôn ngữ với Python. Đây là một thư viện tuyệt vời để làm việc với ngôn ngữ tự nhiên.

Hơn hết, Natural Language ToolKit là một mã nguồn mở, cộng đồng rộng lớn và hỗ trợ tốt.

Một số hỗ trợ cơ bản của Natural Language ToolKit:

- Tokenize và tách các tag từ text input.

```
>>> import nltk
>>> sentence = """At eight o'clock on Thursday morning
... Arthur didn't feel very good."""
>>> tokens = nltk.word_tokenize(sentence)
>>> tokens
['At', 'eight', "o'clock", 'on', 'Thursday', 'morning',
'Arthur', 'did', "n't", 'feel', 'very', 'good', '.']
>>> tagged = nltk.pos_tag(tokens)
>>> tagged[0:6]
[('At', 'IN'), ('eight', 'CD'), ("o'clock", 'JJ'), ('on', 'IN'),
('Thursday', 'NNP'), ('morning', 'NN')]
```

Hình vẽ 2.18: NLTK hỗ trợ Tokenize và tách các tag từ text input

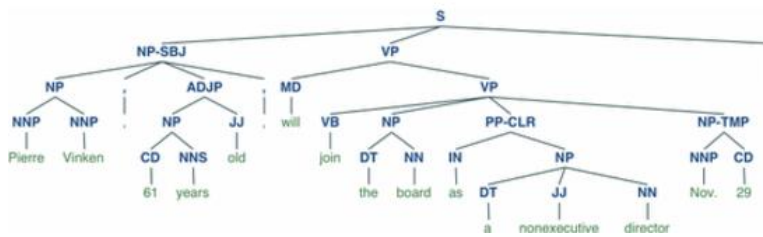
- Xác định các entities.

```
>>> entities = nltk.chunk.ne_chunk(tagged)
>>> entities
Tree('S', [(('At', 'IN'), ('eight', 'CD'), ("o'clock", 'JJ'),
('on', 'IN'), ('Thursday', 'NNP'), ('morning', 'NN'),
Tree('PERSON', [(('Arthur', 'NNP')]),
('did', 'VBD'), ("n't", 'RB'), ('feel', 'VB'),
('very', 'RB'), ('good', 'JJ'), ('.', '.'))])])
```

Hình vẽ 2.19: Entities trong NLTK

- Hiển thị parse tree.

```
>>> from nltk.corpus import treebank
>>> t = treebank.parsed_sents('wsj_0001.mrg')[0]
>>> t.draw()
```



Hình vẽ 2.20: Parse tree trong NLTK

2.5.2. Why Natural Language Toolkit?

Một số nguyên nhân Natural Language Toolkit được chọn làm thư viện xử lý ngôn ngữ tự nhiên của project pyZork:

- Natural Language Toolkit là một thư viện có nhiều hướng dẫn và một cộng đồng lớn hỗ trợ lẫn nhau trong quá trình làm việc.
- Natural Language Toolkit là mã nguồn mở, nên miễn phí. Điều này thích hợp cho sinh viên nghiên cứu và phát triển ứng dụng trên đó.
- Natural Language Toolkit được tích hợp trực tiếp vào project mà không cần dùng qua một Service nào bên ngoài để gọi. Điều này giúp project chạy tốt và ổn định hơn trong quá trình xử lý.
- Natural Language Toolkit được đánh giá cao trong cộng đồng nghiên cứu xử lý ngôn ngữ tự nhiên trên toàn cầu.

2.5.3. ChatterBot

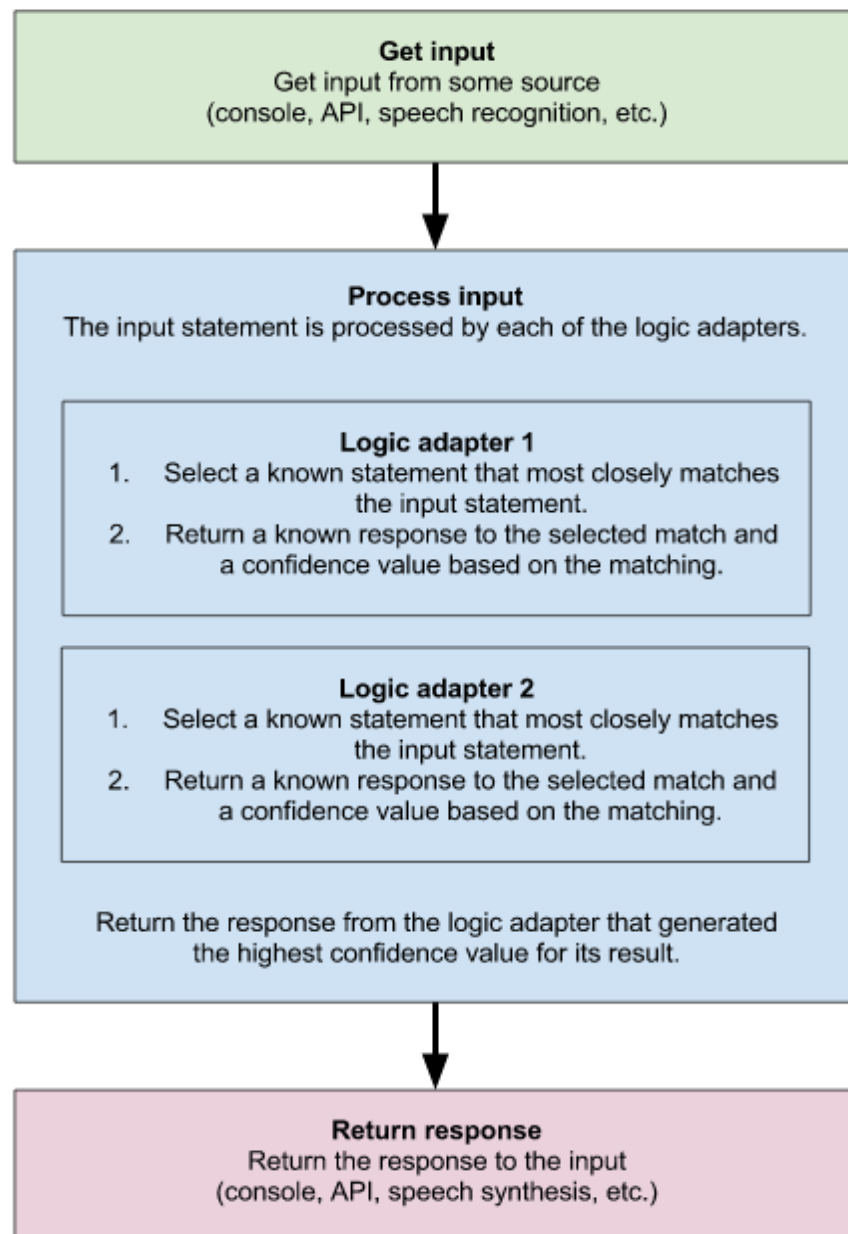
ChatterBot là một thư viện mã nguồn mở phát triển trên Python, core của ChatterBot viết lên từ Natural Language Toolkit. ChatterBot giúp nhà phát triển dễ dàng tạo ra những câu trả lời cho đoạn chat mà người dùng nhập vào. ChatterBot sử dụng các thuật toán của máy học để tự tạo ra nhiều kết quả hơn và linh hoạt hơn trong giao tiếp với người dùng.

Nhờ ChatterBot mà các nhà phát triển ứng dụng có liên quan đến xử lý ngôn ngữ tự nhiên sẽ dễ dàng hơn trong việc tạo ra các đoạn hội thoại tự động với người dùng.



Hình vẽ 2.21: Thư viện ChatterBot

Ngôn ngữ thiết kế độc lập của ChatterBot cho phép nó được huấn luyện để nói nhiều loại ngôn ngữ khác nhau. Bản chất máy học của ChatterBot cho phép Bot tự nâng cao trình độ bằng cách học hỏi các đoạn hội thoại đã cũ để có nhiều câu trả lời linh hoạt.



Hình vẽ 2.22: Quy trình hoạt động của ChatterBot

2.6. JSON

JSON (JavaScript Object Noation) là một định dạng hoán vị dữ liệu nhanh. Chúng dễ dàng cho chúng ta đọc và viết. Dễ dàng cho thiết bị phân tích và phát sinh. Chúng là cơ sở dựa trên tập hợp của Ngôn Ngữ Lập Trình JavaScript, tiêu chuẩn ECMA-262 phiên bản 3. JSON là một định dạng kiểu text mà hoàn toàn độc lập với các ngôn ngữ họ hàng C, gồm có C, C++, C#, Java, JavaScript, Perl,

Python và nhiều ngôn ngữ khác. Những đặc tính trên cho thấy JSON là một ngôn ngữ hoán vị dữ liệu lý tưởng.

JSON được xây dựng trên 2 cấu trúc:

- Là tập hợp của các cặp tên và giá trị name – value. Trong những ngôn ngữ khác nhau, đây được coi là một đối tượng (object), sự ghi (record), cấu trúc (struct), từ điển (dictionary), bảng băm (hash table), danh sách khóa (keyed list), hay mảng liên hợp.
- Là một tập hợp các giá trị đã được sắp xếp. Trong hầu hết các ngôn ngữ, được nhập thấy là một mảng, vector, tập hợp hay là 1 dãy sequence.

Đây là một cấu trúc dữ liệu phổ dụng. Hầu như tất cả các ngôn ngữ lập trình hiện đại đều hỗ trợ chúng trong một hình thức nào đó. Chúng tạo nên ý nghĩa của một định dạng hoán vị dữ liệu với các ngôn ngữ lập trình cũng đã được cơ sở hóa trên cấu trúc này.

Trong khóa luận này, Json đảm nhận vai trò lưu thông tin của người chơi. Điều này hết sức tiện lợi khi kết hợp với ngôn ngữ Python.

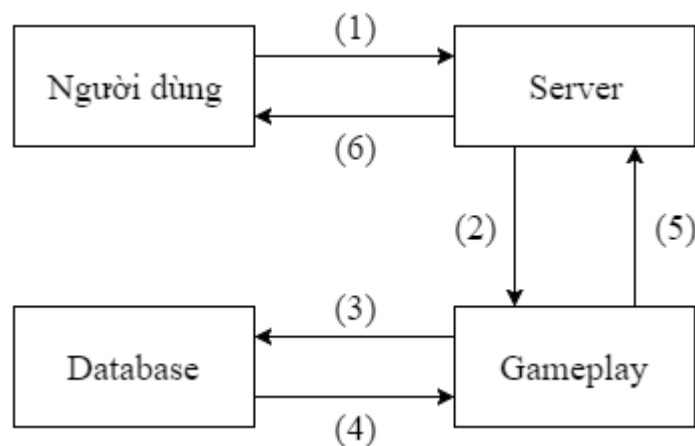
Chương 3. PHÂN TÍCH THIẾT KẾ

3.1. Tổng quan chức năng

Text-based Game pyZork gồm các tính năng chính:

- Đăng ký / Reset / Lưu người chơi
- Module nhận và xử lý thông tin text từ người chơi
 - o Các command trong Game
 - o Xử lý ingame: keyword trong mỗi màn chơi, vật phẩm trong mỗi màn chơi
- Tích hợp xử lý ngôn ngữ tự nhiên đối với trường hợp người chơi nhập text sai keyword của màn chơi.

3.2. Kiến trúc tổng quát



Hình vẽ 3.1: Tổng quan kiến trúc game pyZork

Giải thích các hình khối:

- Người dùng: Là người chơi game, trực tiếp nói chuyện với Chatbot để vượt qua các thử thách trong Game.
- Server: Đây là nơi giao tiếp, dùng để gửi và nhận các tin nhắn phía người dùng.
- Gameplay: Đây là nơi cốt lõi xử lý game. Các logic về màn chơi, xử lý ngôn ngữ tự nhiên sẽ được hiện thực trong Gameplay.

- Database: Lưu thông tin người chơi.

Giải thích Flow:

- (1): Đây là bước gửi in nhắn (request) từ người dùng đến server. Server đóng vai trò là người nhận tin và gửi tin. Không đảm nhiệm vai trò xử lý logic game.
- (2): Sau khi nhận được tin nhắn server sẽ gọi module Gameplay để xử lý. Gameplay sẽ đảm nhận vai trò xử lý logic game. Sẽ gọi sang cơ sở dữ liệu để lấy các thông tin người chơi.
- (3): Gọi sang DB để lấy thông tin hiện tại của người chơi.
- (4): Dữ liệu được trả về từ Database
- (5): Gameplay xử lý xong sẽ gửi thông tin cho Server
- (6): Server nhận thông tin này và gửi đến người dùng.

3.3. Game pyZork

3.3.1. Yêu cầu

- Tên game: pyZork
- Thể loại: text-based game
- Đối tượng hướng đến: 12+
- Nền tảng: Facebook Messenger
- Chủ đề: Game phiêu lưu, mạo hiểm
- Môi trường phát triển: Python 3.5
- Thời gian dự kiến: 3 tháng

3.3.2. Cốt truyện

Năm 2016 trước công nguyên, cả thế giới lúc ấy chưa thật sự rộng lớn như hiện giờ. Tất cả nhân loại sống chung với nhau trong một vương quốc mang tên Zork thật thanh bình.

Sau một vụ mùa bội thu, gặt hái nhiều mùa màng. Nhà vua của Zork mở hội ăn mừng, vừa là để đón chào một năm mới thật sung túc cũng như chúc mừng đến thành công của vụ mùa năm cũ.

Lúc này một thế lực hắc ám luôn muốn thôn tằm trái đất chọn lúc vương quốc Zork sơ hở đã cho quân xâm chiếm kinh thành Zork. Đánh tan hàng phòng thủ của vương quốc và chiếm quyền trở thành kẻ thống trị loài người.

Kể từ ngày đó, Zork được đặt lại là pyZork. Nơi mà thế lực hắc ám tồn tại, chúng lan rộng chết chóc và sự đau đớn không chừa một ai.

Một nhóm pháp sư lương thiện và kỵ sĩ với mong muốn phục quốc đã thoát khỏi sự kìm hãm của thế lực pyZork. Họ tập hợp lại và cùng nhau tìm ra 5 Nguyên Tố Tự Nhiên để mang đến đền Zork và phong ấn thế lực pyZork thêm 2016 năm nữa. Mang về thịnh vượng cho vương quốc Zork.

Và cuộc hành trình khó nhọc này bắt đầu.

3.3.3. GamePlay

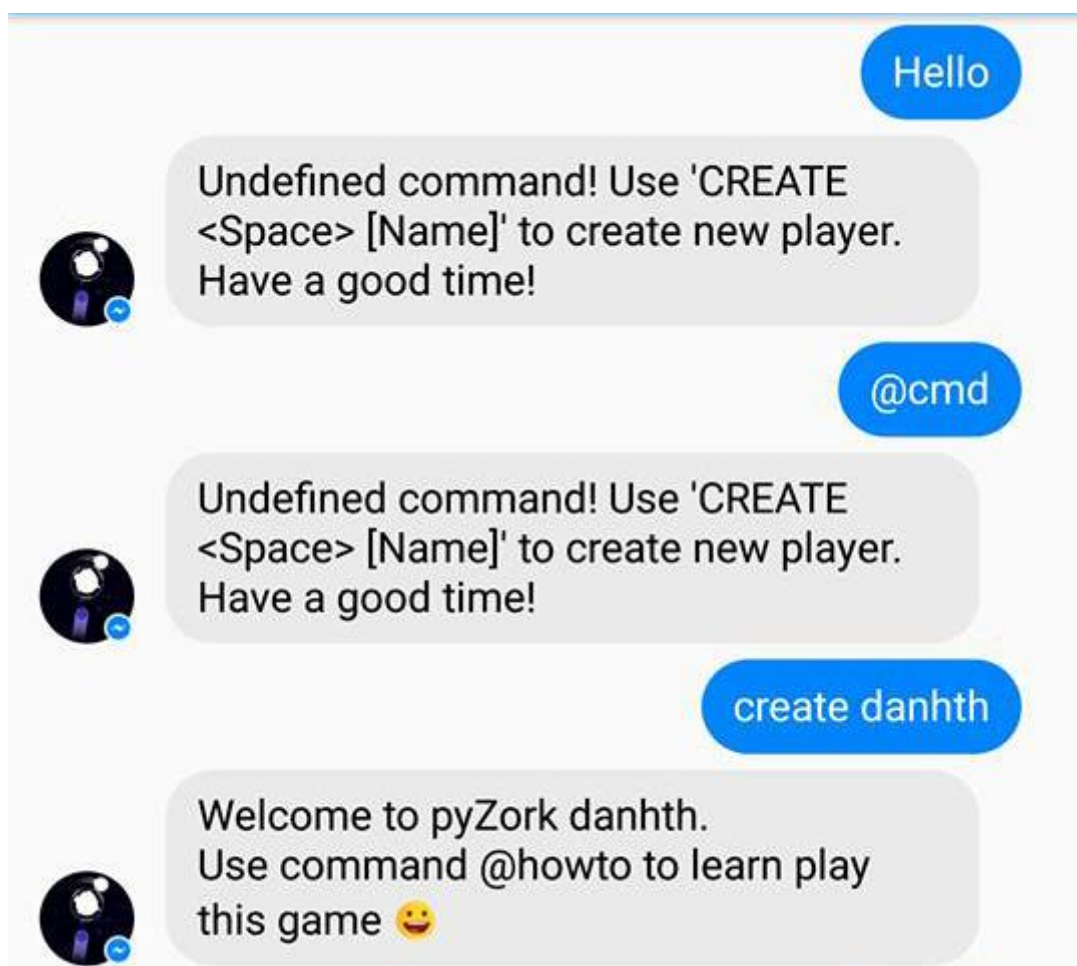
Để chơi pyZork người chơi cần đăng ký dữ liệu người chơi bằng tài khoản Facebook. Sau khi đăng ký người chơi có thể hòa mình vào thế giới của pyZork.

Một số command cơ bản trong Game:

Command	Tính năng
CREATE	Tạo tài khoản mới
@cmd	Lấy về thông tin của các command
@status	Thông tin trạng thái hiện của người chơi
@inventory	Thông tin vật phẩm đã thu thập
@howto	Hướng dẫn cách chơi game
@hardreset	Reset quá trình và chơi lại từ đầu.

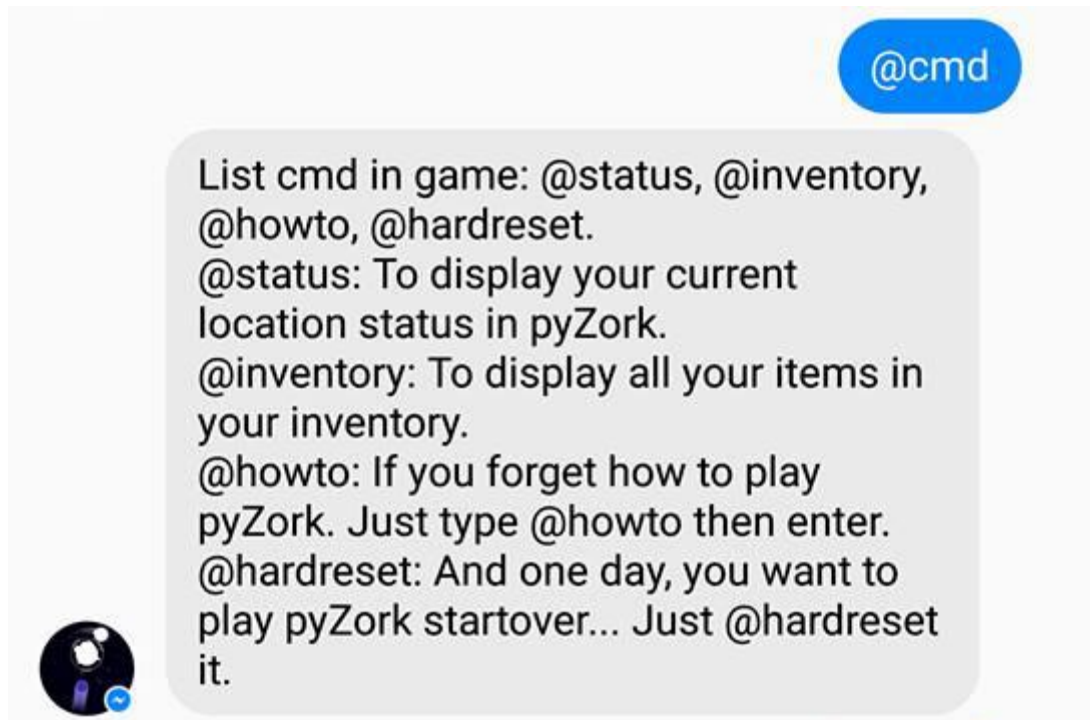
Bảng 3.1: Một số command trong pyZork

- **CREATE:** Khi người dùng lần đầu tiên nhắn tin đến Chatbot. Hệ thống sẽ tự động phát hiện và gửi yêu cầu người chơi tạo nhân vật mới để được chơi game. Nếu không sử dụng command CREATE để tạo nhân vật mới thì người chơi sẽ không thể tham gia vào thế giới game pyZork.



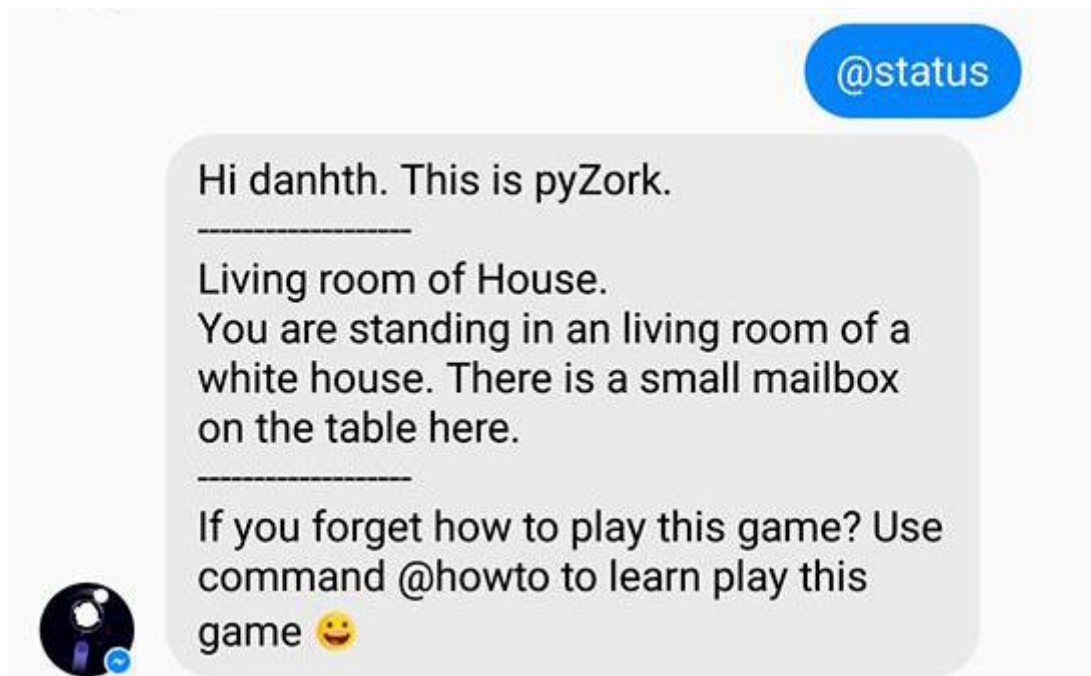
Hình vẽ 3.2: CREATE command trong pyZork

- **@cmd:** Trong game có nhiều command mà người chơi không thể nhớ hết. Chỉ cần gõ **@cmd** thì sẽ được Chatbot hỗ trợ danh sách các command của hệ thống sử dụng trong lúc chơi game.



Hình vẽ 3.3: Danh sách các command hệ thống trong pyZork

- **@status**: Command này hỗ trợ người chơi nhìn lại thông tin của mình ở cảnh hiện tại.

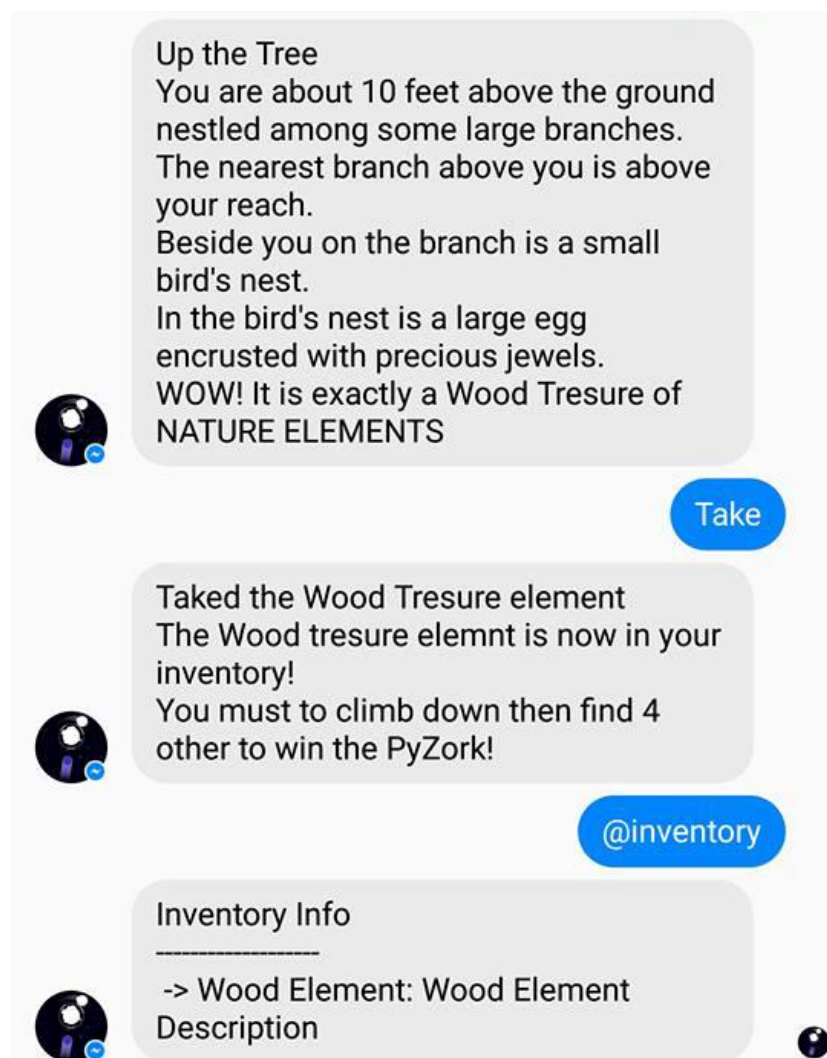


Hình vẽ 3.4: Command lấy thông tin hiện tại của người chơi

- **@inventory**: Trong game có nhiều item cần thu thập. Người chơi có thể kiểm tra danh sách các items của mình bất kỳ lúc nào bằng cách sử dụng command **@inventory**.

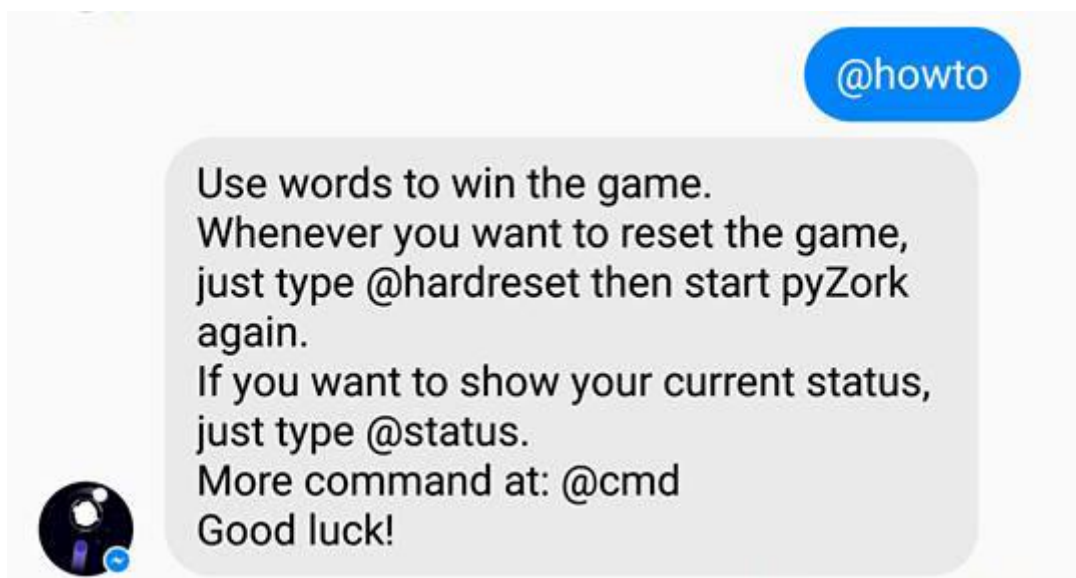


Hình vẽ 3.5: Inventory rỗng



Hình vẽ 3.6: Lấy thông tin về kho đồ của người dùng.

- **@howto:** Để đọc lại luật chơi game và hướng dẫn cách chơi. Người chơi gõ **@howto** để được hỗ trợ hướng dẫn cách làm việc với pyZork.



Hình vẽ 3.7: Hướng dẫn chơi game pyZork

- **@hardreset:** Nếu người chơi muốn được chơi lại từ đầu với pyZork. Dùng command **@hardreset** để được reset tất cả thông tin và chơi lại từ đầu.

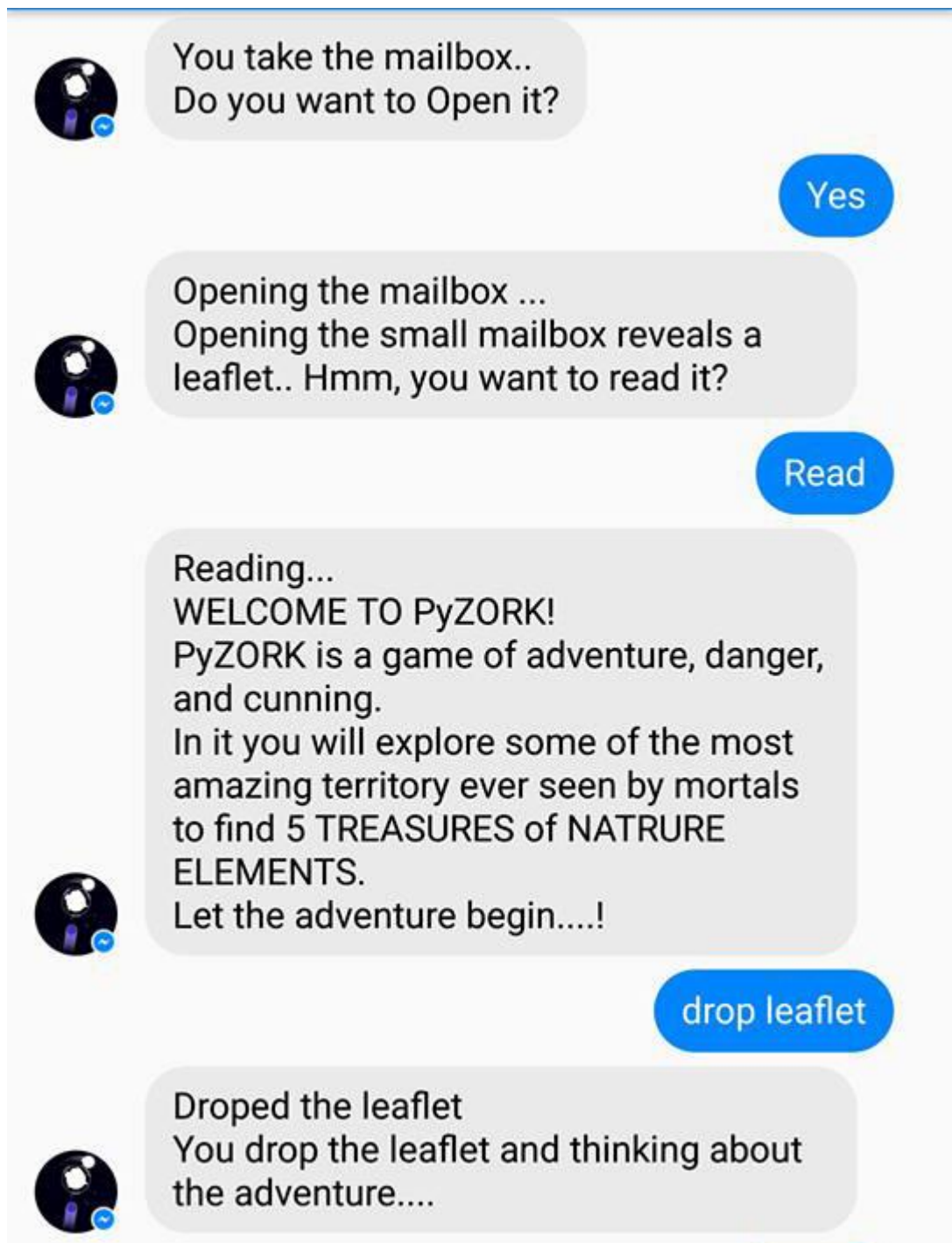


Hình vẽ 3.8: Reset pyZork data

pyZork là một text-based game, chính vì vậy thử thách lớn nhất của người chơi là đưa ra quyết định ở mỗi chặng trong game sao cho phù hợp để được đi tiếp.

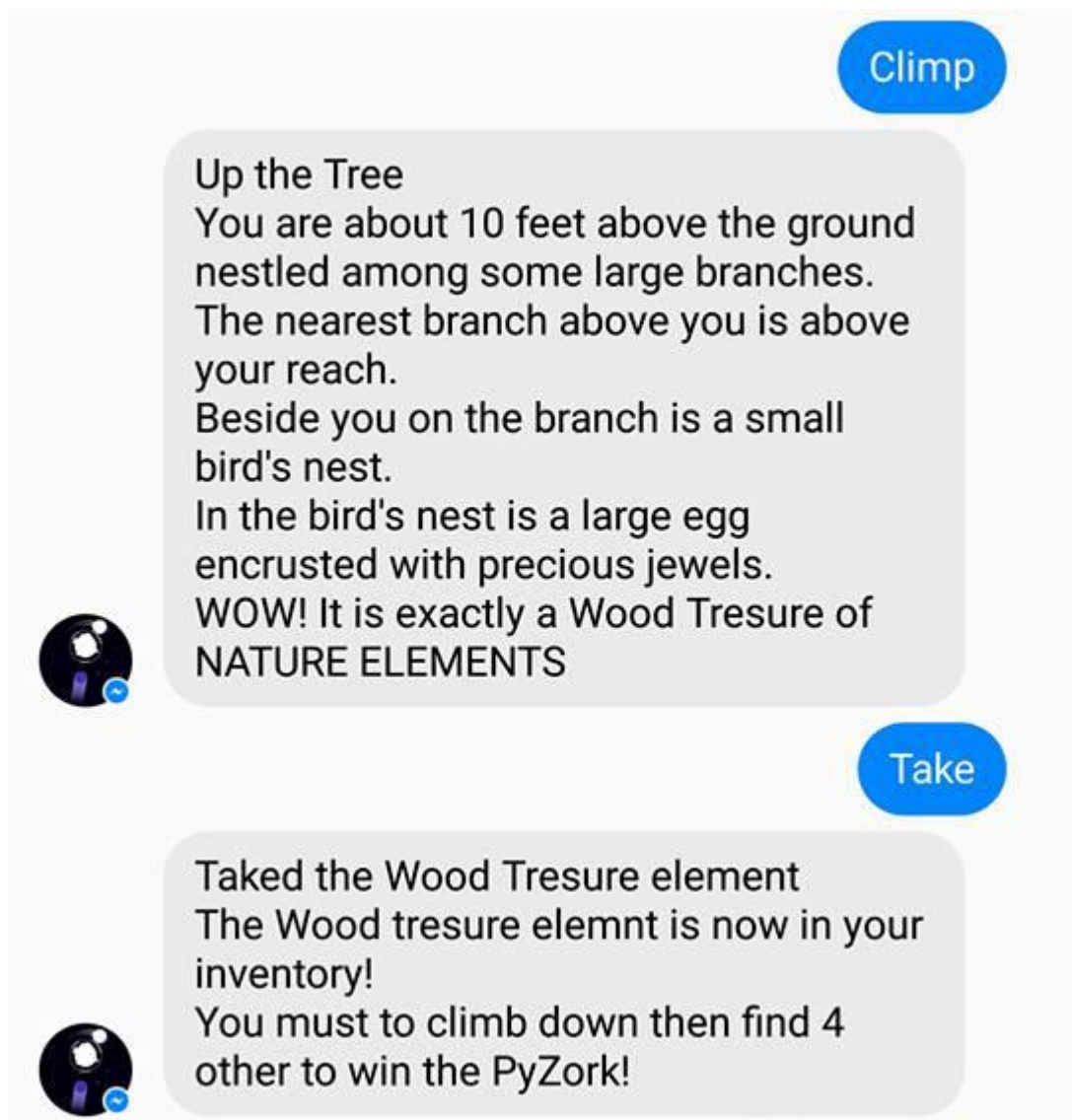
Với mỗi chặng sẽ là một cảnh quan được miêu tả thông qua một description, dựa theo description đó và những dữ liệu từng thu thập, người chơi phải đưa ra quyết định của mình thông qua text.

Các text nhập vào có thể đơn giản như: ‘go east’, ‘go west’, ‘take gem’, cũng có thể khó khăn như ‘kill the monster’, ‘take the treasure’.



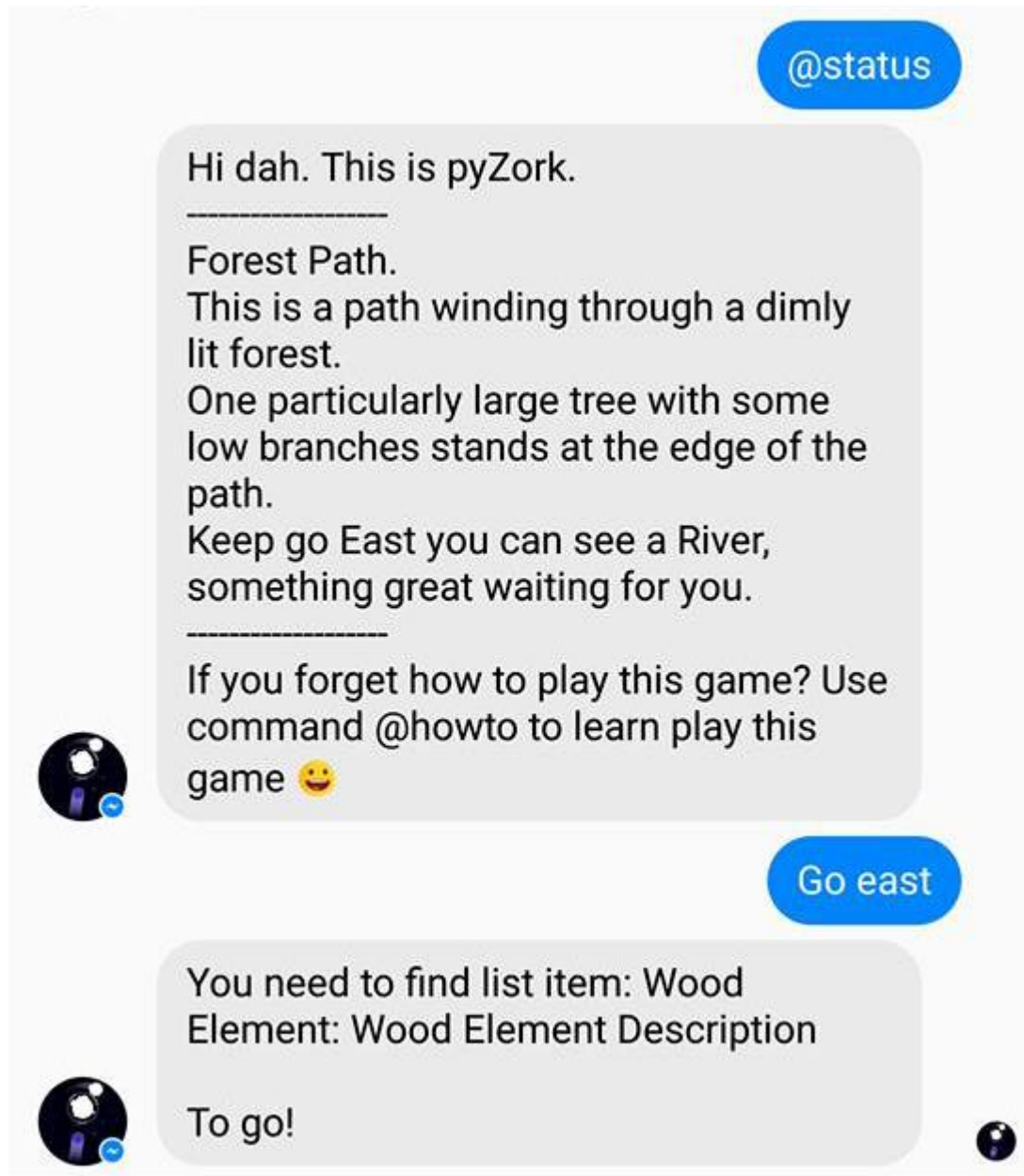
Hình vẽ 3.9: Người dùng nhập text để di chuyển trong pyZork

Một số động từ thông dụng trong pyZork: go, take, grab, back, east, north, west, south, kill, cross....

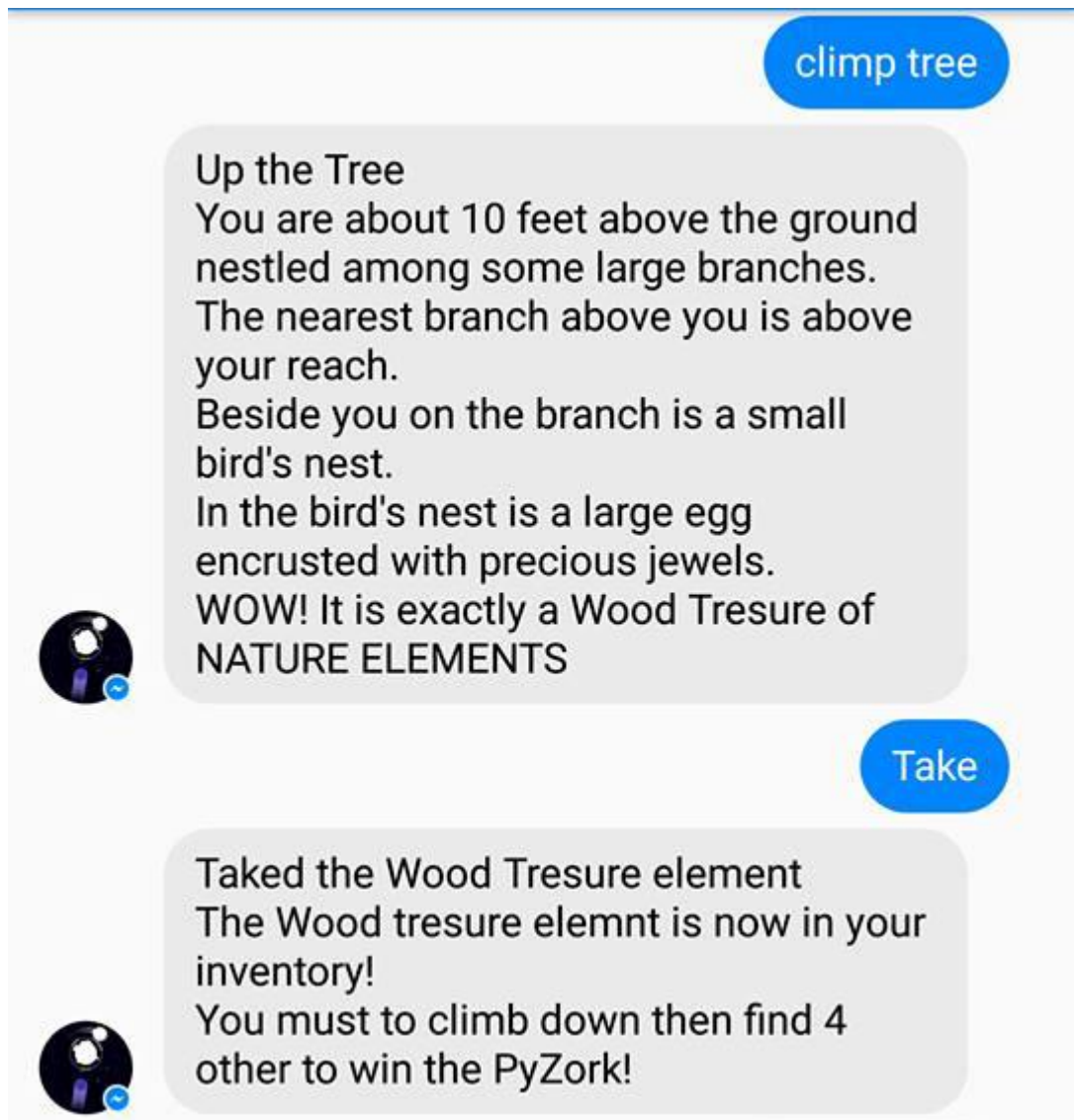


Hình vẽ 3.10: Dùng các động từ thường để dễ dàng chơi game hơn

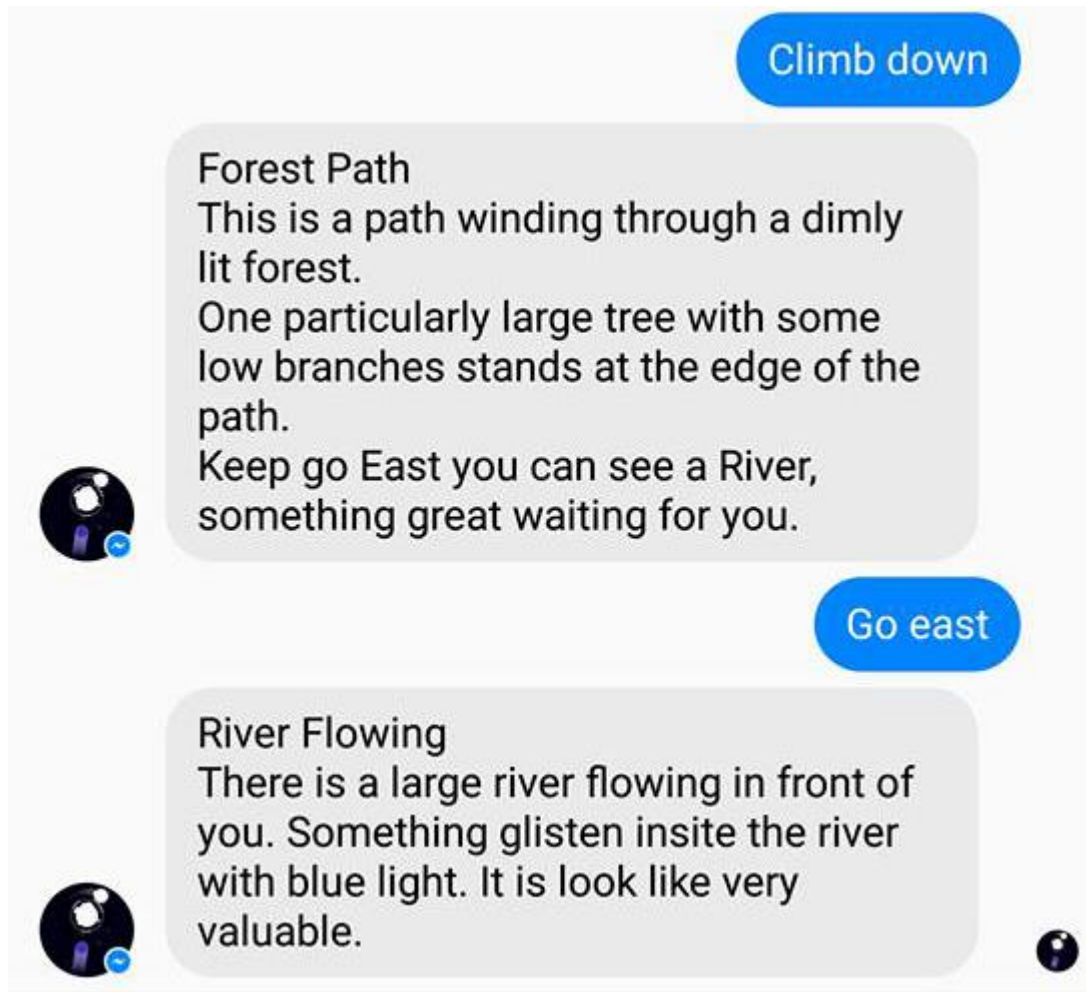
Với mỗi màn chơi sẽ có các yêu cầu về item đặc biệt để mở cửa. Ví dụ khi người chơi cần đi đến một màn chơi, mà ở đó chứa nguyên tố Lửa. Thì yêu cầu họ cần phải mang theo trong người yếu tố Nước để được vào. Nếu vẫn cố vào thì sẽ bị trả về lại điểm xuất phát vì mất lượt chơi.



Hình vẽ 3.11: Yêu cầu item để được đi tiếp.



Hình vẽ 3.12: Nhật vật phẩm yêu cầu để được qua màn



Hình vẽ 3.13: Người chơi được đi tiếp sau khi nhặt được vật phẩm yêu cầu

Chính vì yêu cầu khó như trên, người chơi cần tập trung đọc kỹ các thông tin có trên màn chơi. Dựa theo các lời gợi ý đó và suy luận ra hướng đi đúng đắn cho mình.

3.3.4. Phân tích, thiết kế Game

Text-based game là thể loại game kinh điển từ những ngày hệ điều hành DOS còn sống. Game chơi bằng cách nhập text từ người chơi, AI trong game sẽ xử lý chuỗi text đó thành các hành động của người chơi và hành động trong game sau đó trả về một đoạn text để báo hiệu kết quả của hành động vừa rồi của người dùng.

Game chia làm 3 giai đoạn chính:

- **Bước 1:** Nhận input từ người dùng

Khi người dùng gửi một tin nhắn đến Facebook Bot Messenger, ta cần hiện thực một hàm lắng nghe tin nhắn trên webhook. Mỗi lúc người dùng gửi tin nhắn cho Chatbot, Facebook sẽ mang tin nhắn đó gửi sang webhook của lập trình viên. Tại đây lập trình viên có thể tùy ý sử dụng nội dung tin nhắn và đáp trả lại nội dung tin nhắn đã được gửi.

- **Bước 2:** Xử lý input từ người dùng

Khi tin nhắn người dùng được đón nhận, sẽ là bước tiếp theo - xử lý.

Trong gói tin gửi đến webhook sẽ bao gồm thông tin của người gửi, ta dùng thông tin đó để kiểm tra xem thông tin người gửi trong cơ sở dữ liệu của game. Sau đó tiến hành xử lý chuỗi tin nhắn của người dùng cho phù hợp với trạng thái người dùng hiện tại: chưa đăng ký game, đã đăng ký game, đang ở phân cảnh nào trong game...

Sau khi xử lý xong sẽ trả về một đoạn tin nhắn dùng để hồi đáp lại phía người dùng.

- **Bước 3:** Trả kết quả về cho người dùng

Khi đã có đoạn tin nhắn hồi đáp cho người dùng, điều tiếp theo là cần sử dụng Facebook Messenger Postback để gửi trả lại kết quả của hành động vừa rồi từ người dùng.

Tiếp tục và tiếp tục cho đến khi hoàn thành game.

Phân tích 5 thuộc tính thiên nhiên:



Hình vẽ 3.14: Mô tả ngũ hành trong pyZork

Thiết kế về 5 thuộc tính thiên nhiên:

Tên	Yêu cầu	Ý nghĩa
Thuộc tính Kim	<p>Xuất hiện ở một nơi hoang vắng. Được cất giữ bởi một con Rồng khổng lồ.</p> <p>Để đạt được thuộc tính Kim ta cần mang theo thanh gươm tiêu diệt con Rồng.</p>	<p>Kim là một yếu tố thể hiện sự mạnh mẽ và kiên trì trong ngũ hành. Kim đại diện cho những gì rắn chắc, khả năng chịu đựng và khỏe mạnh. Mặt khác, Kim là vật dẫn cho điện, nhiệt nên rất linh hoạt. Ở mặt tích cực, Kim truyền</p>

		<p>đạt thông tin, sắc sảo và sự kiên trì, khỏe mạnh. Khi mặt tiêu cực, Kim có thể là sự hủy hoại, hiểm họa diệt vong và phiền muộn. Kim như một con dao cực sắc nhọn với 2 lưỡi.</p> <p>Vật đại diện: Kim loại, đao kiếm</p>
Thuộc tính Gỗ	<p>Xuất hiện trong khu rừng già.</p> <p>Nằm trên một ngọn cây cao nhất trong khu rừng.</p> <p>Để có được thuộc tính Mộc, người chơi cần leo lên đỉnh cây và thu thập thuộc tính Mộc này.</p>	<p>Mộc chỉ mùa xuân, sự tăng trưởng và đời sống cây cỏ. Cây tre Việt Nam được ca ngợi về khả năng mềm dẻo trước gió, nhưng lại được dùng làm giáo. Được nhìn với hình tượng cây, năng lượng của Mộc có tính sinh sôi nảy nở, nuôi dưỡng và mềm dẻo.</p> <p>Vật đại diện: Cây cối</p>
Thuộc tính Đất	<p>Trên đỉnh một ngọn đồi trong thung lũng sâu.</p> <p>Người chơi leo lên đỉnh đồi và chọn thu thập nguyên tố Thổ.</p> <p>Cần mang theo thuộc tính</p>	<p>Thổ nói về môi trường ươm trồng, nuôi dưỡng và phát triển. Nơi sinh ra và chết đi của vạn vật. Thổ nuôi dưỡng, hỗ trợ và tương sinh đến các hành khác trong ngũ hành. Ở mặt tích cực, Thổ biểu thị sự</p>

	Kim và Thủy để có thể thu thập được Thổ.	công bằng, tinh mịch, trí khôn ngoại và bản năng thiên tài. Khi tiêu cực, Thổ tạo ra ngột ngạt hoặc biểu thị tính hay lo về những khó khăn không tồn tại. Vật đại diện: Đất, đá
Thuộc tính Nước	Nằm sâu dưới đáy một con sông. Khi đi qua dòng sông cần chú ý để dừng lại thu thập thuộc tính Thủy.	Thủy chỉ về sự dịu mát của nước, hoặc có thể là sự dữ dội của nước biển, bão bùng... Thủy là hiện thân của nghệ thuật và vẻ đẹp tinh khiết. Thủy liên quan đến mọi vật. Trong mặt tích cực, Thủy biểu hiện tính nuôi dưỡng, hỗ trợ.. Khi tiêu cực, Thủy biểu hiện ở sự hao mòn và kiệt quệ. Kết với cảm xúc, Thủy gợi cho thấy nỗi sợ hãi, lo lắng... Vật đại diện: Sông, biển
Thuộc tính Lửa	Nằm trong miệng núi lửa đang tuôn trào. Để nhận được thuộc tính Lửa, người chơi cần mang theo thuộc tính Thủy để dập tắt miệng núi lửa và đi vào nhật	Lửa chỉ sức nóng, ám chỉ đến khói lửa. Lửa đem lại ánh sáng, hơi ấm, hạnh phúc hoặc cũng có thể là sự tuôn trào, bùng nổ và bạo tàn. Ở khía cạnh tích cực, Lửa tiêu biểu

	lấy thuộc tính Lửa.	cho danh dự và sự công bằng. Ở mặt hạn chế, Lửa tượng trưng cho sự thích gây hấn, chiến tranh. Vật đại diện: Mặt trời, Núi lửa
--	---------------------	--

Bảng 3.2: Các thuộc tính bí ẩn cần tìm trong pyZork

Thiết kế các khung cảnh lớn trong pyZork:

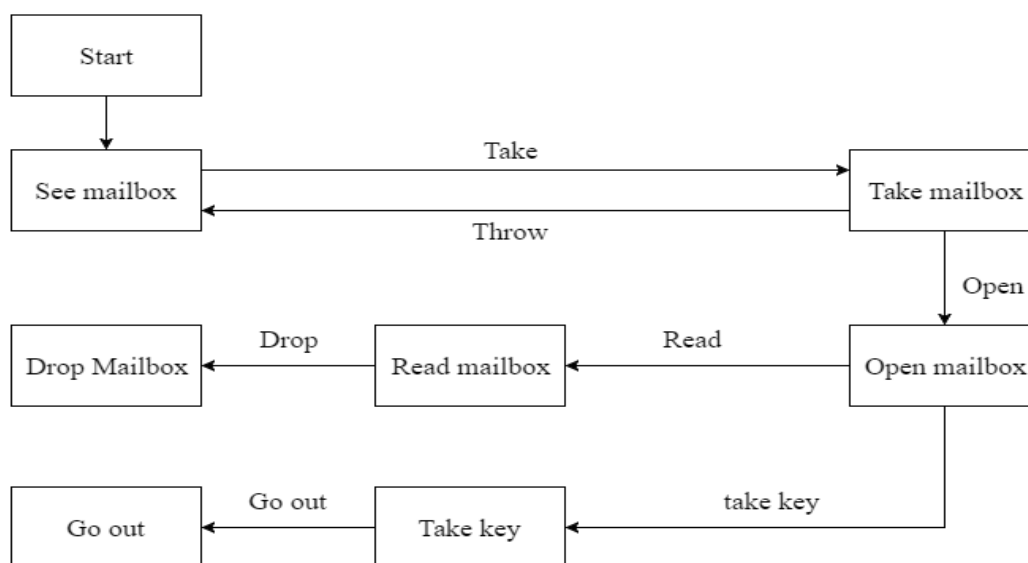
Cảnh	Nội dung	Item yêu cầu	Item nhận được
Bắt đầu, trong một căn phòng	Người chơi đang trong căn phòng, trên bàn là một chiếc thư có ghi nội dung mật dẫn đến chỗ cất giấu các báu vật ngũ hành, giải cứu Zork.	-	Chìa khóa
Ra ngoài	Khung cảnh xung quanh tối mịt mù, phía Đông là một khu rừng rậm ẩn chứa nhiều bí mật mà chưa ai tìm thấy.	Chìa khóa	-
Trong rừng phía Đông	Trong khu rừng rậm, trên ngọn cây cao nhất khu rừng có chứa bí mật về thuộc tính Gõ	-	Thuộc tính Gõ
Dòng sông bạc ở phía Đông khu rừng	Dòng sông bạc ẩn chứa một tinh thể lạ lấp lánh màu xanh. Rất có thể đó là một thuộc tính thiên nhiên	Thuộc tính Gõ	Thuộc tính Nước

Đi về hướng Tây Bắc	Đi qua một ngọn núi lửa phía Tây Bắc	Thuộc tính Nước	Thuộc Tính Lửa
Về ngôi làng, gặp thợ rèn	Gặp thợ rèn và sử dụng thuộc tính Lửa	Thuộc tính Lửa	Thanh kiếm
Đi về hướng Nam	Gặp hang rồng, đánh với con rồng bằng thanh kiếm để nhận thuộc tính Kim	Thanh kiếm	Thuộc Tính Kim
Đi về hướng Tây Nam	Gặp một thung lũng.	Thuộc Tính Nước, Kim	Thuộc tính Đất
Đi thẳng về điện thờ phía Nam	Đến điện thờ phía Nam và giao nộp 5 thuộc tính ngũ hành	Kim, Gỗ, Đất, Nước, Lửa	Thắng

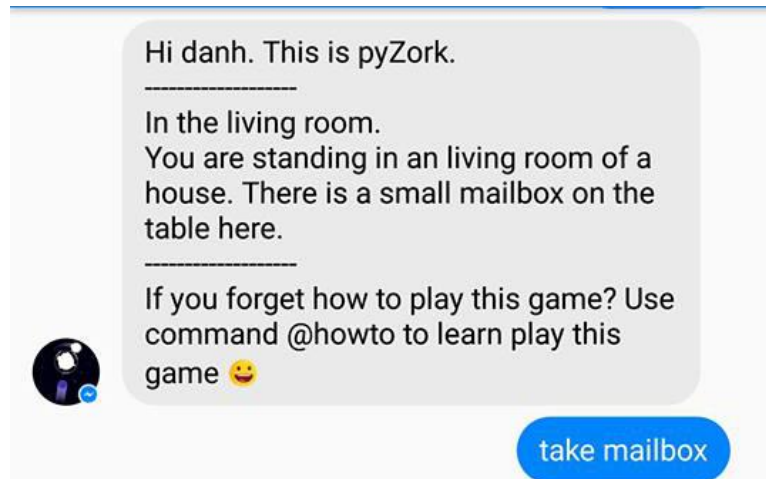
Bảng 3.3: Thiết kế các khung cảnh lớn trong pyZork

Mỗi khung cảnh sẽ có những vật phẩm và sự kiện khác nhau xảy ra. Mỗi vật phẩm, sự kiện trong game đều có tác dụng riêng, tất cả đều hướng đến việc thử thách và đưa người chơi đến chiến thắng.

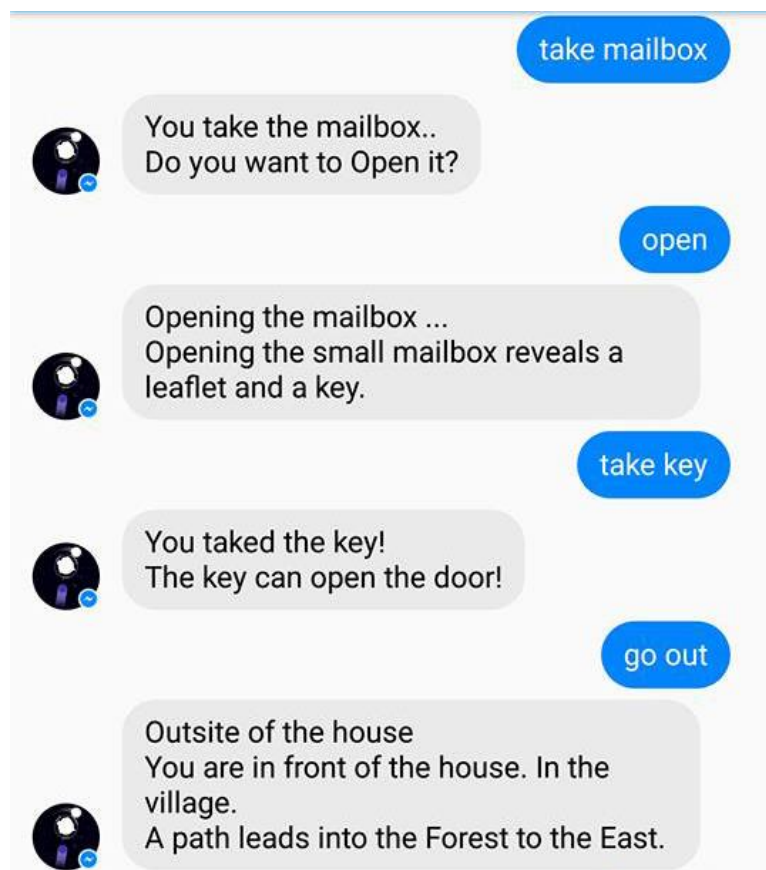
- **Bắt đầu trong một căn phòng:**



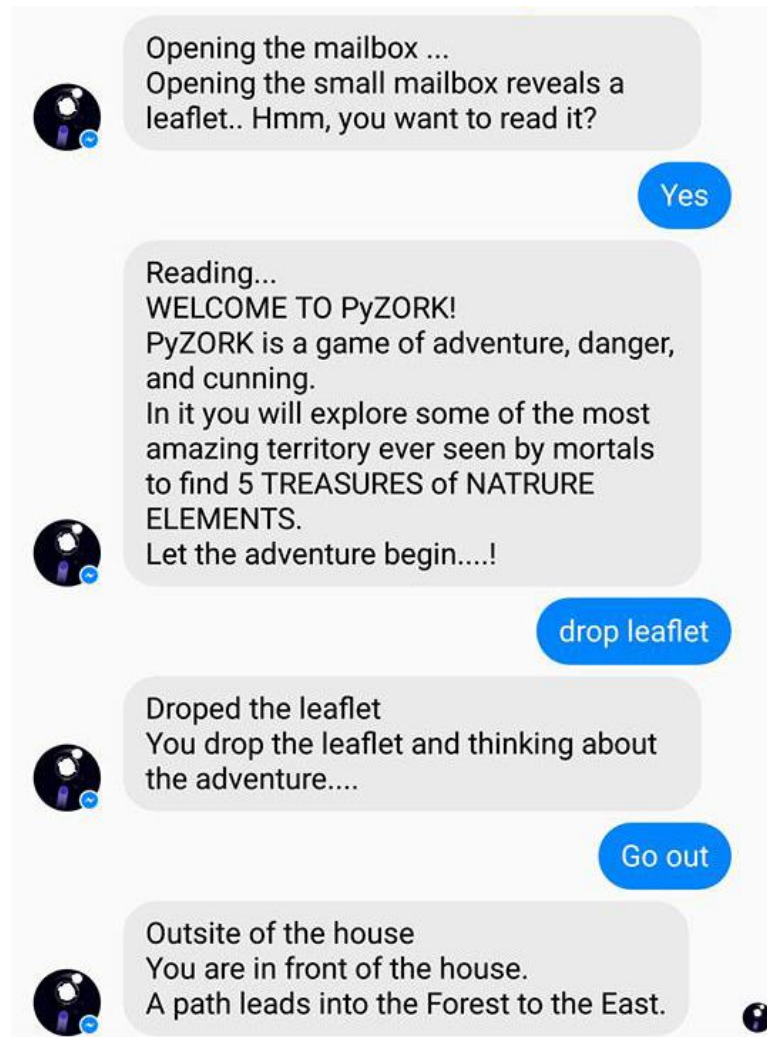
Hình vẽ 3.15: Sơ đồ màn đầu trong pyZork



Hình vẽ 3.16: Mở bức thư để lấy thông tin

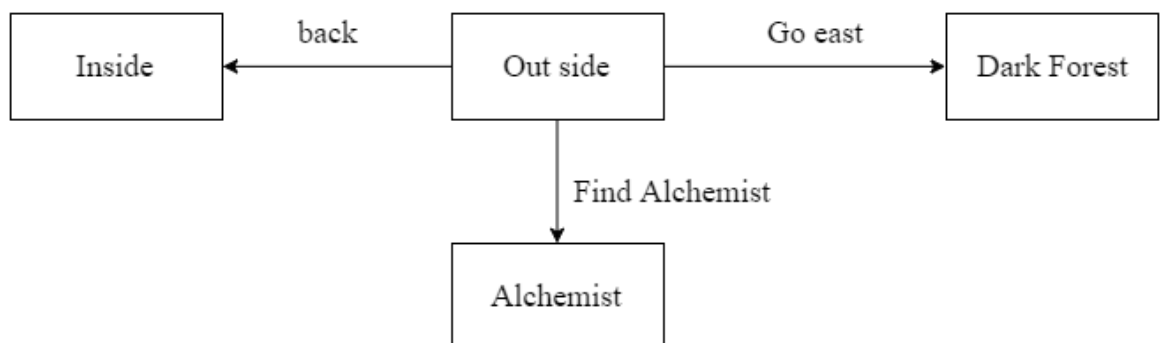


Hình vẽ 3.17: Cần tìm ra chiếc chìa khóa để mở cửa ra ngoài.

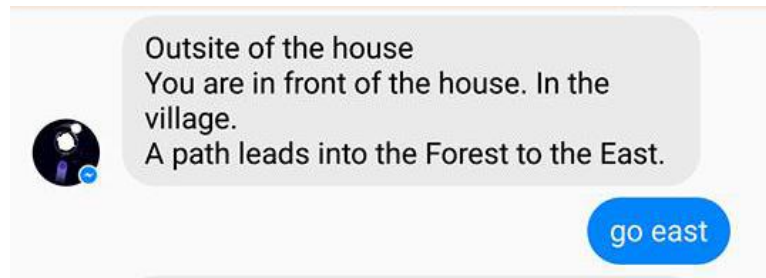


Hình vẽ 3.18: Đọc nội dung trong bức thư giới thiệu game

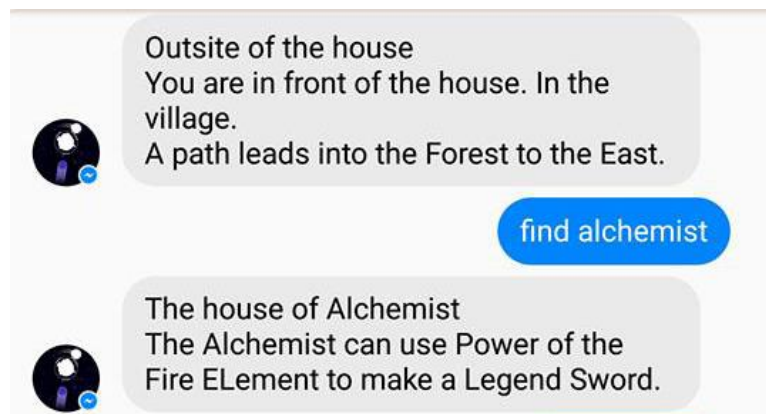
- Ra ngoài khu vực ngôi nhà



Hình vẽ 3.19: Ra bên ngoài ngôi nhà, bắt đầu hành trình

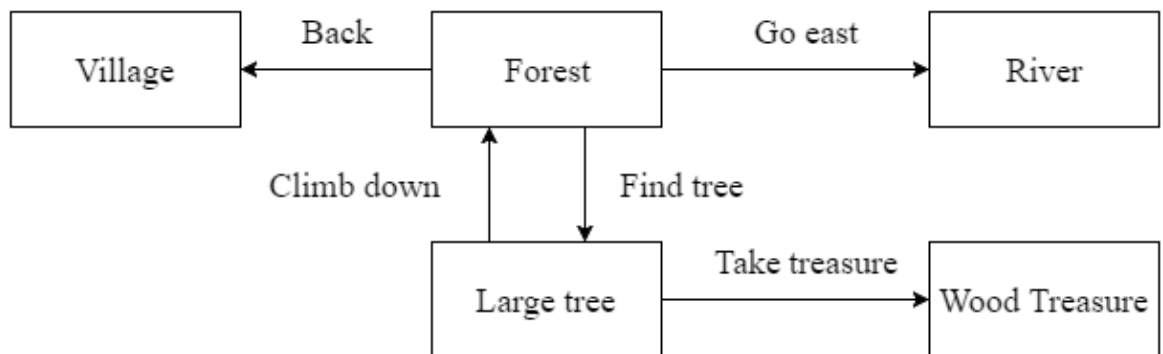


Hình vẽ 3.20: đi về khu rừng hướng đông

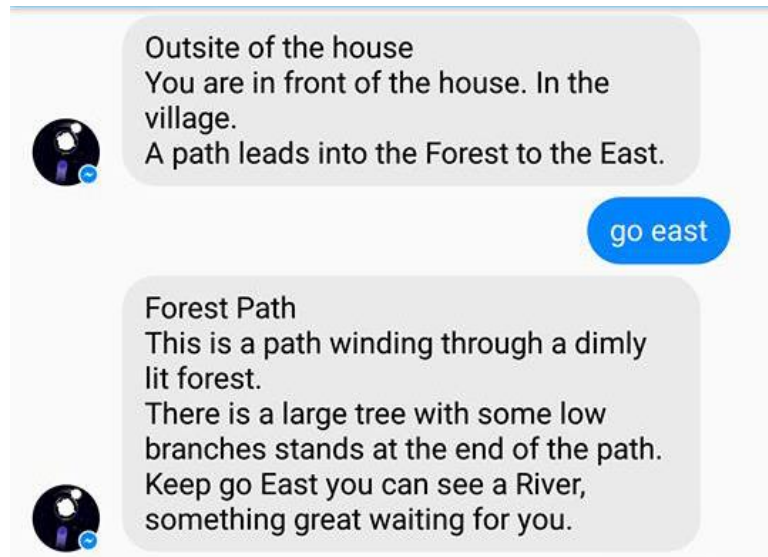


Hình vẽ 3.21: Tìm thợ rèn trong ngôi làng

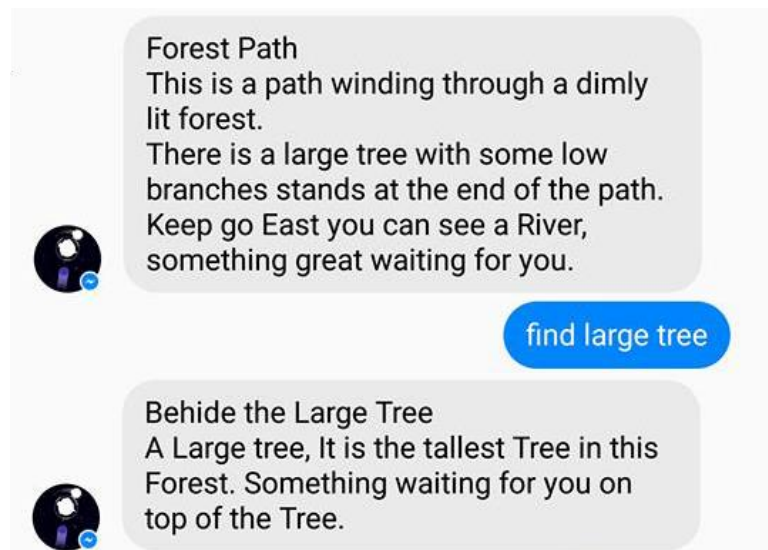
- Vào rừng



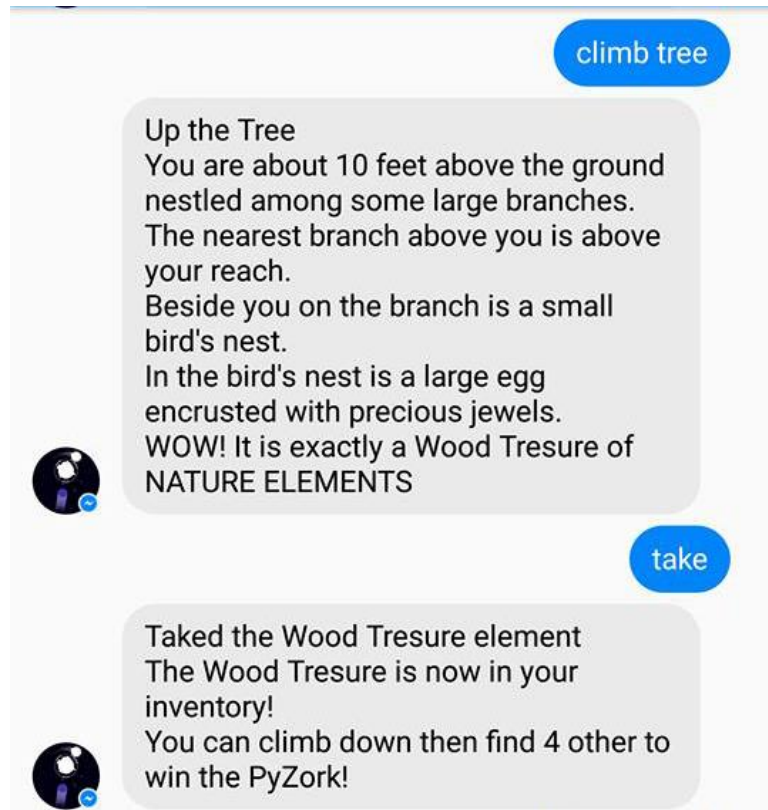
Hình vẽ 3.22: Sơ đồ bên trong khu rừng



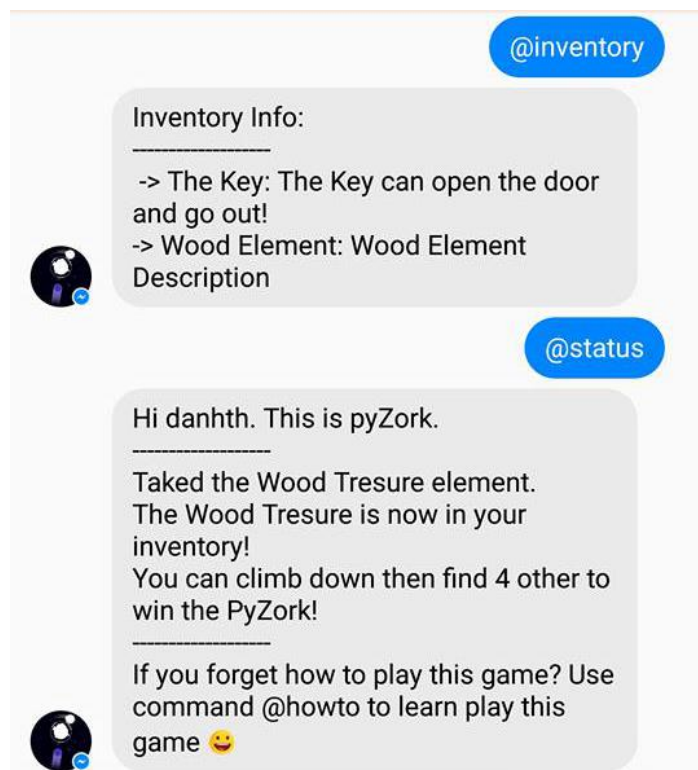
Hình vẽ 3.23: Đi từ ngôi làng vào rừng



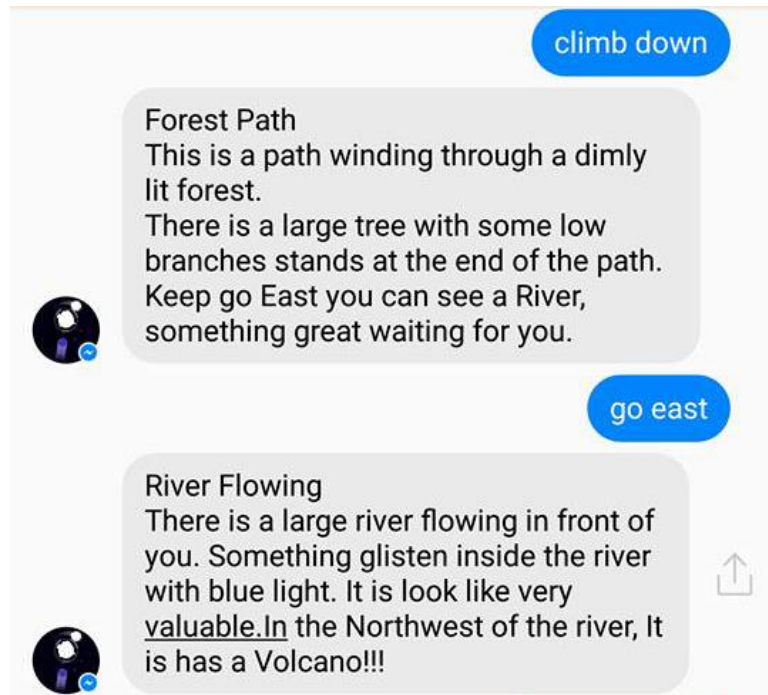
Hình vẽ 3.24: Tìm đến cây đại thụ đang có Wood Treasure



Hình vẽ 3.25: Leo lên cây và lấy Wood Treasure

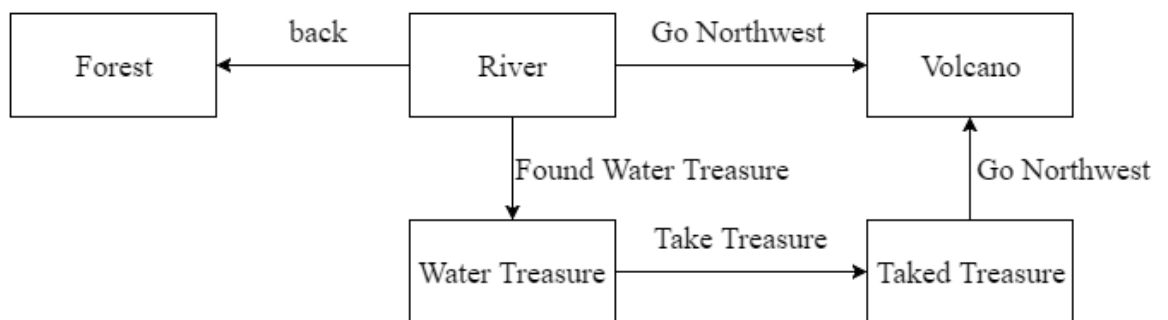


Hình vẽ 3.26: Kiểm tra trạng thái hiện tại

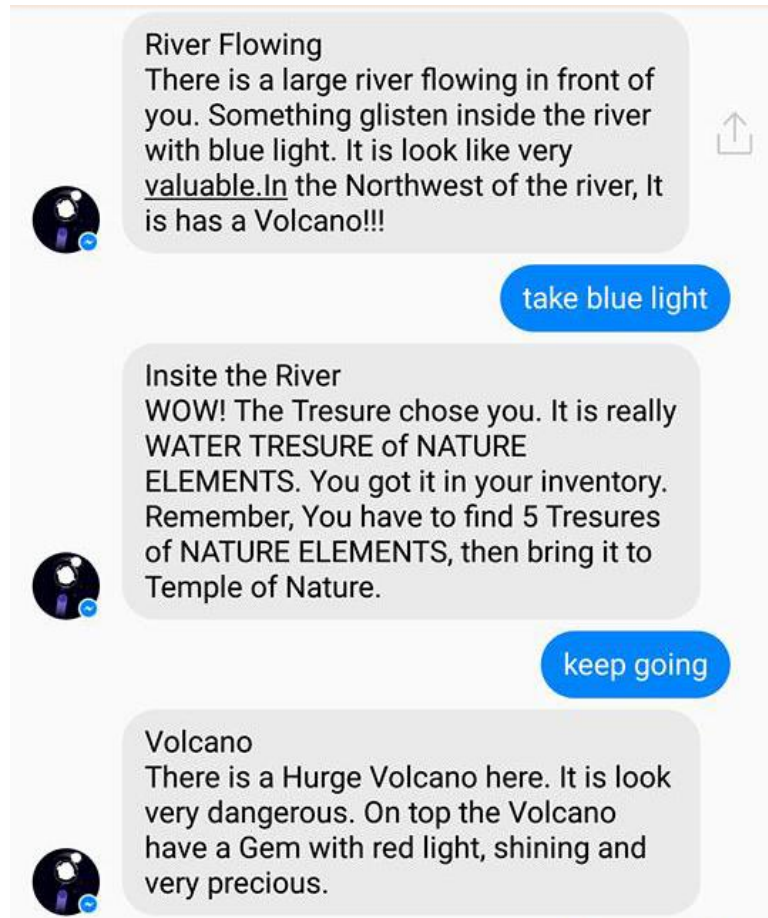


Hình vẽ 3.27: Đến bờ sông hướng đông

- Đến bờ sông

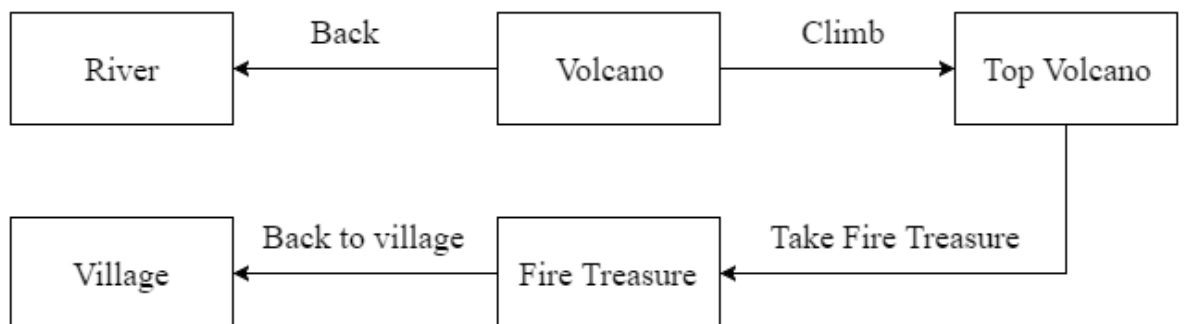


Hình vẽ 3.28: Sơ đồ tại bờ sông

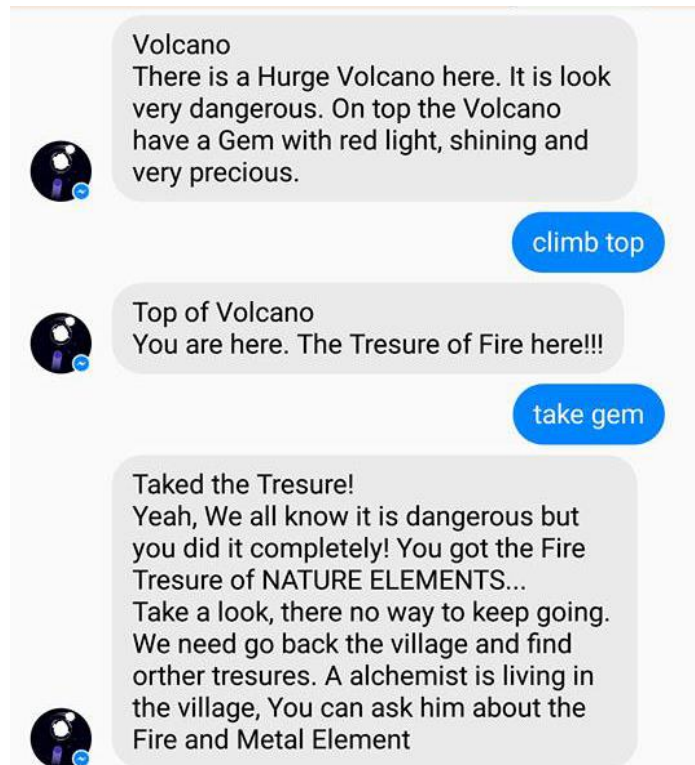


Hình vẽ 3.29: Cách tìm thấy được Water Treasure và đi đến núi lửa

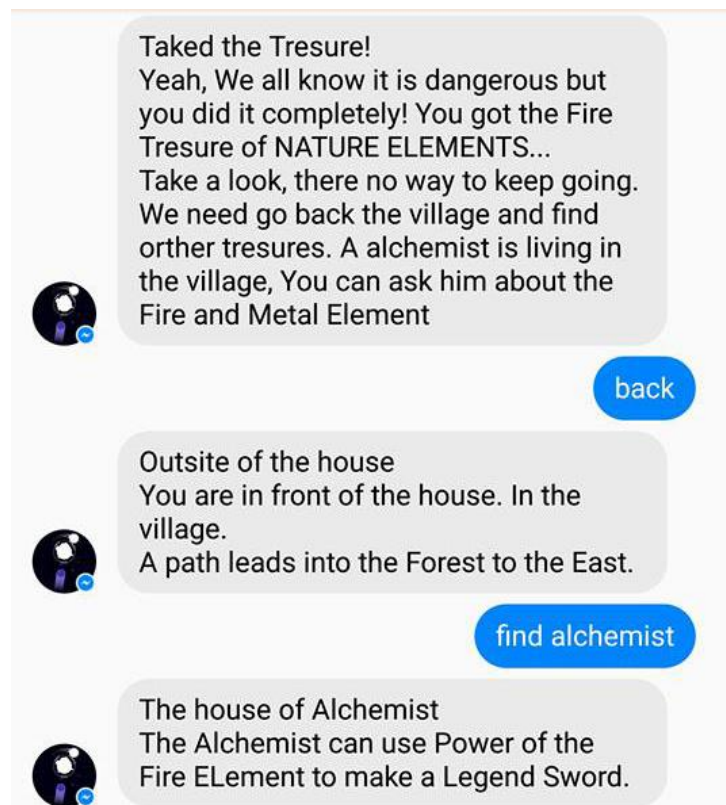
- Đến núi lửa



Hình vẽ 3.30: Sơ đồ tại núi lửa

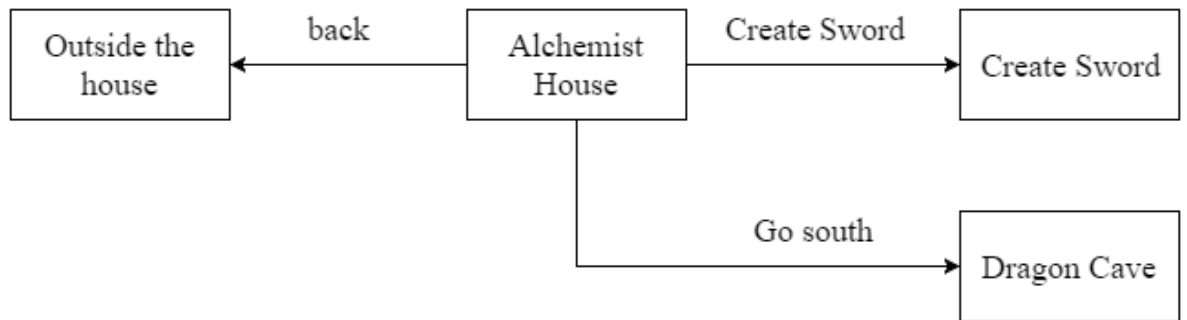


Hình vẽ 3.31: Cách lấy gem trên núi lửa

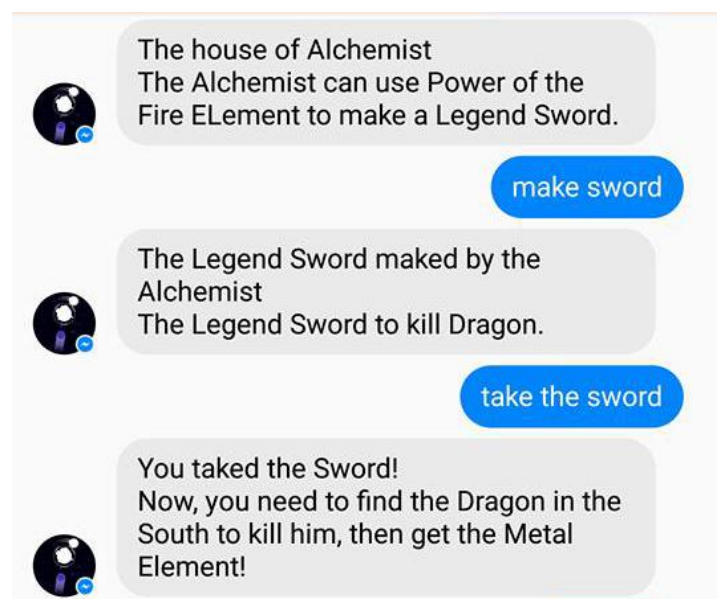


Hình vẽ 3.32: Đi tìm thợ rèn sau khi có Fire Treasure

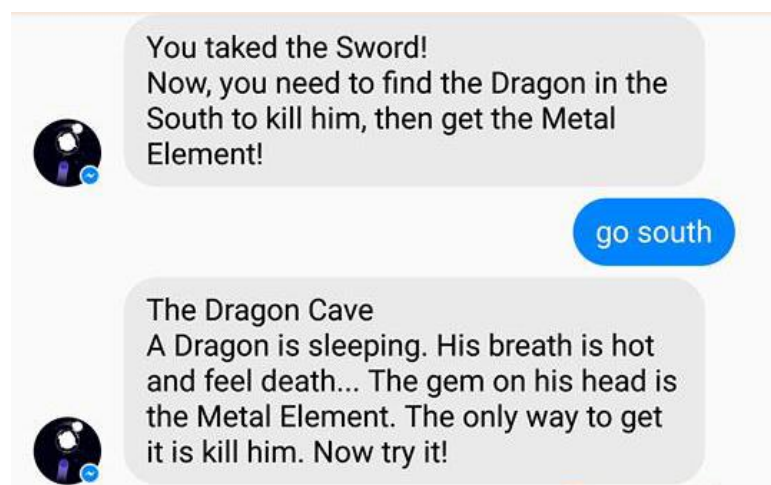
- Tìm thợ rèn



Hình vẽ 3.33: Sơ đồ tìm thợ rèn

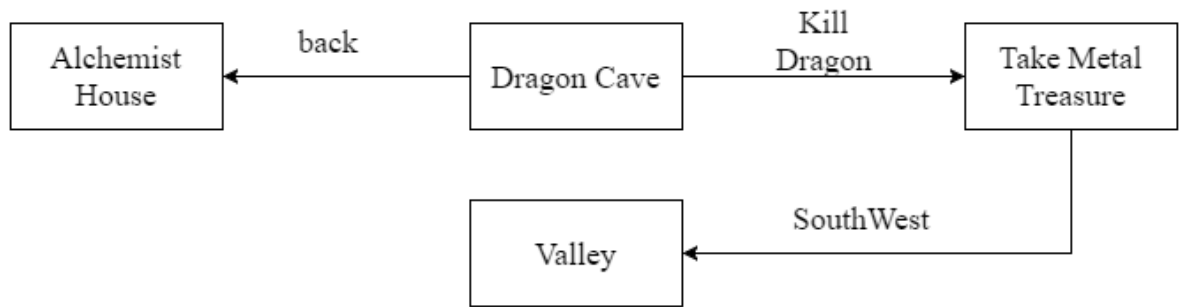


Hình vẽ 3.34: Rèn thanh kiếm giết rồng bằng Fire Treasure

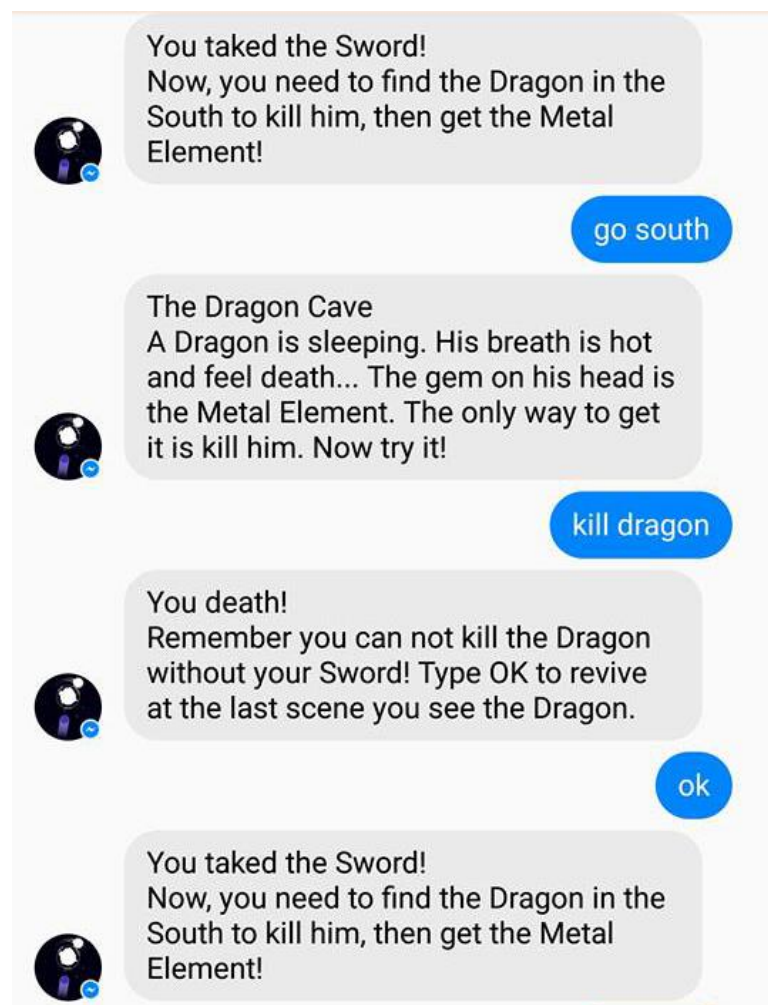


Hình vẽ 3.35: Về hướng nam để tìm hang rồng

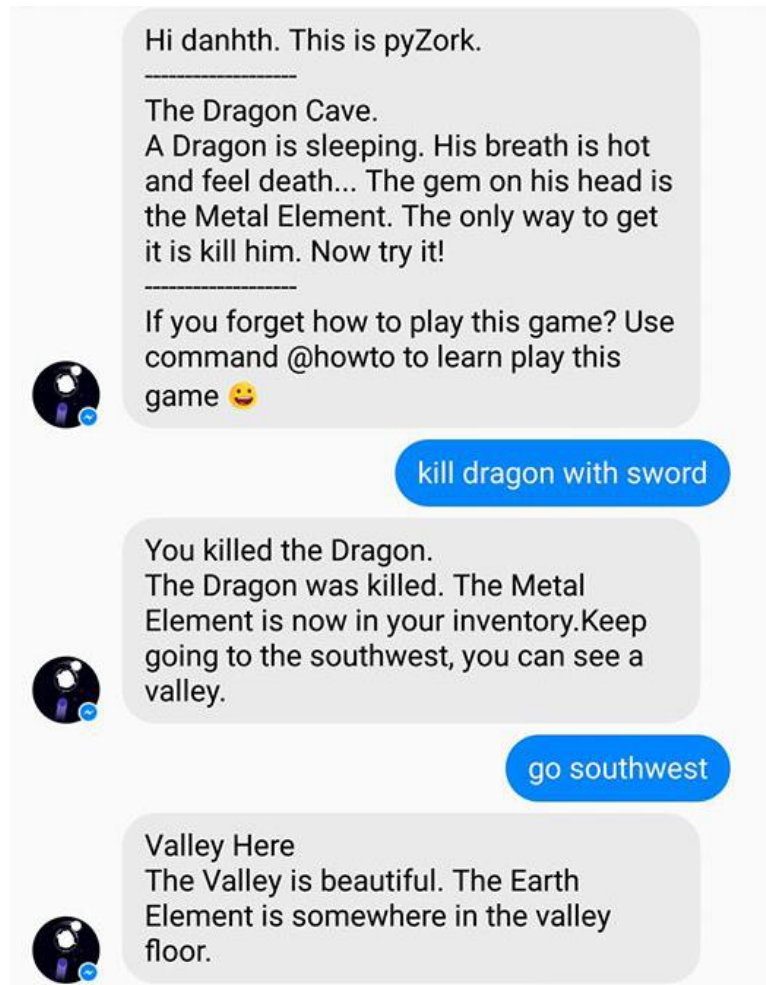
- Diệt rồng



Hình vẽ 3.36: Sơ đồ tại hang rồng

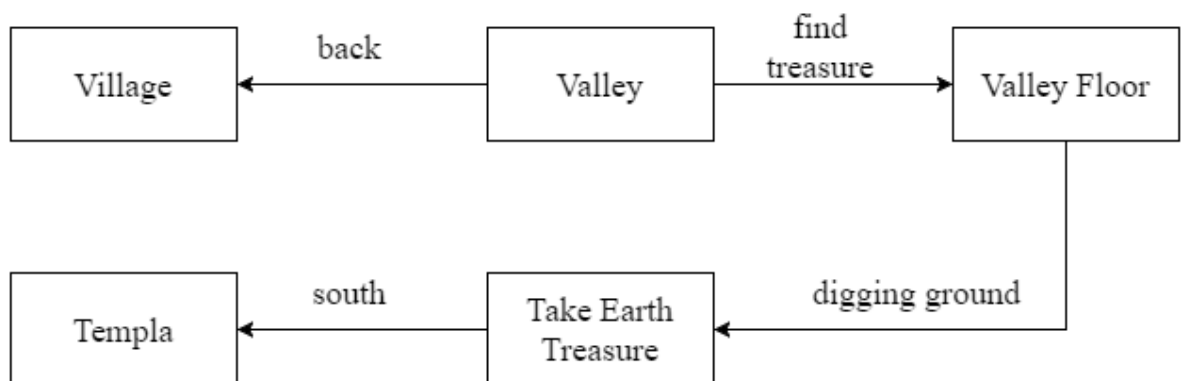


Hình vẽ 3.37: Cách tiêu diệt rồng

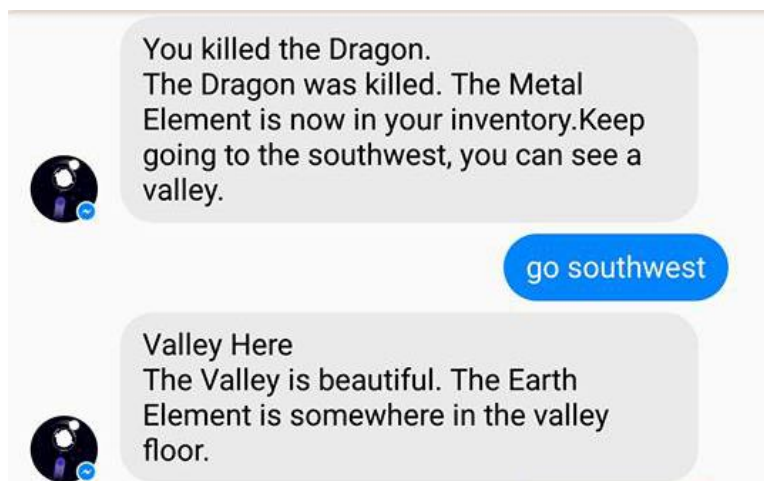


Hình vẽ 3.38: Tiêu diệt rồng để lấy Metal Treasure

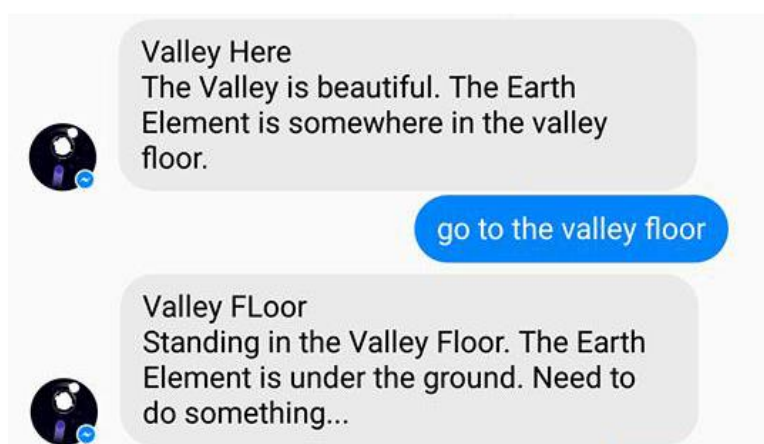
- Đến thung lũng



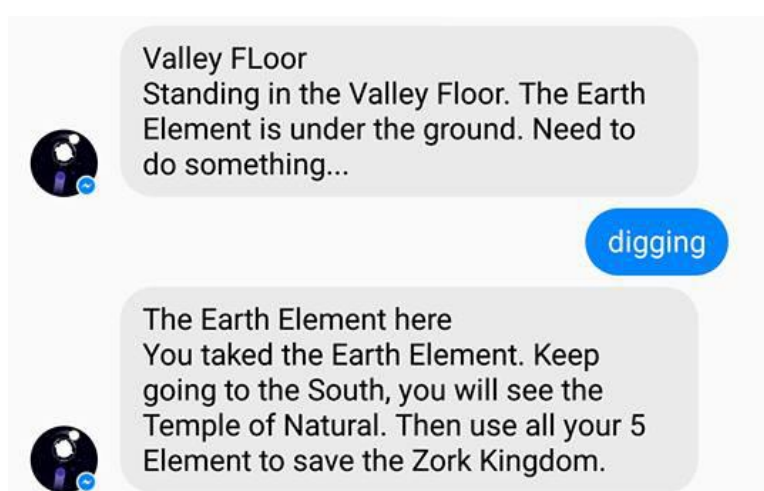
Hình vẽ 3.39: Sơ đồ tại thung lũng



Hình vẽ 3.40: Theo hướng Tây Nam để tìm thấy thung lũng



Hình vẽ 3.41: Xuống đáy thung lũng để tìm Earth Treasure

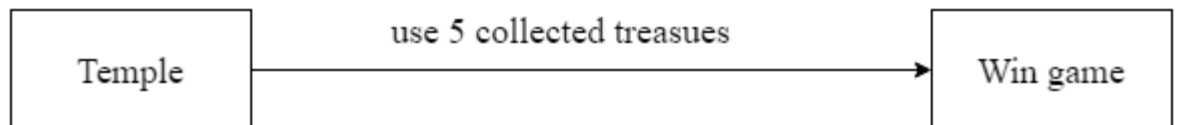


Hình vẽ 3.42: Đào xuống lòng đất để tìm Earth Treasure

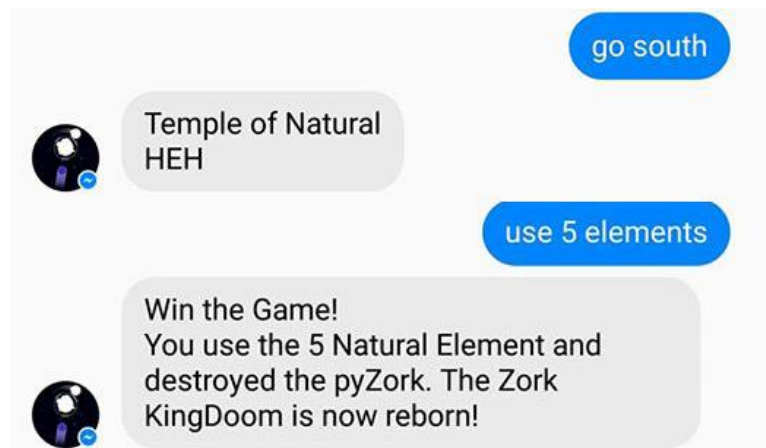


Hình vẽ 3.43: Kiểm tra đủ số lượng 5 Treasure cần tìm

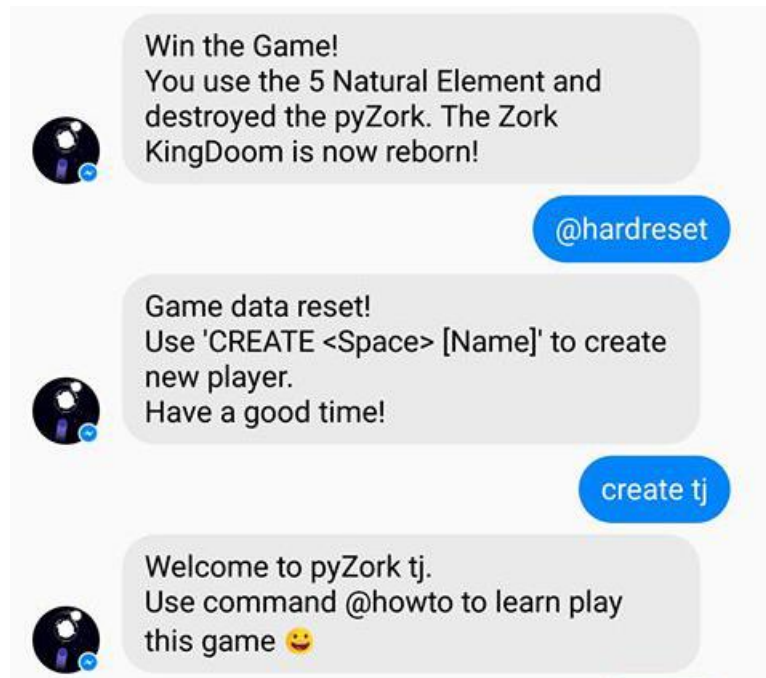
- Đến đền thờ



Hình vẽ 3.44: Sơ đồ tại đền thờ



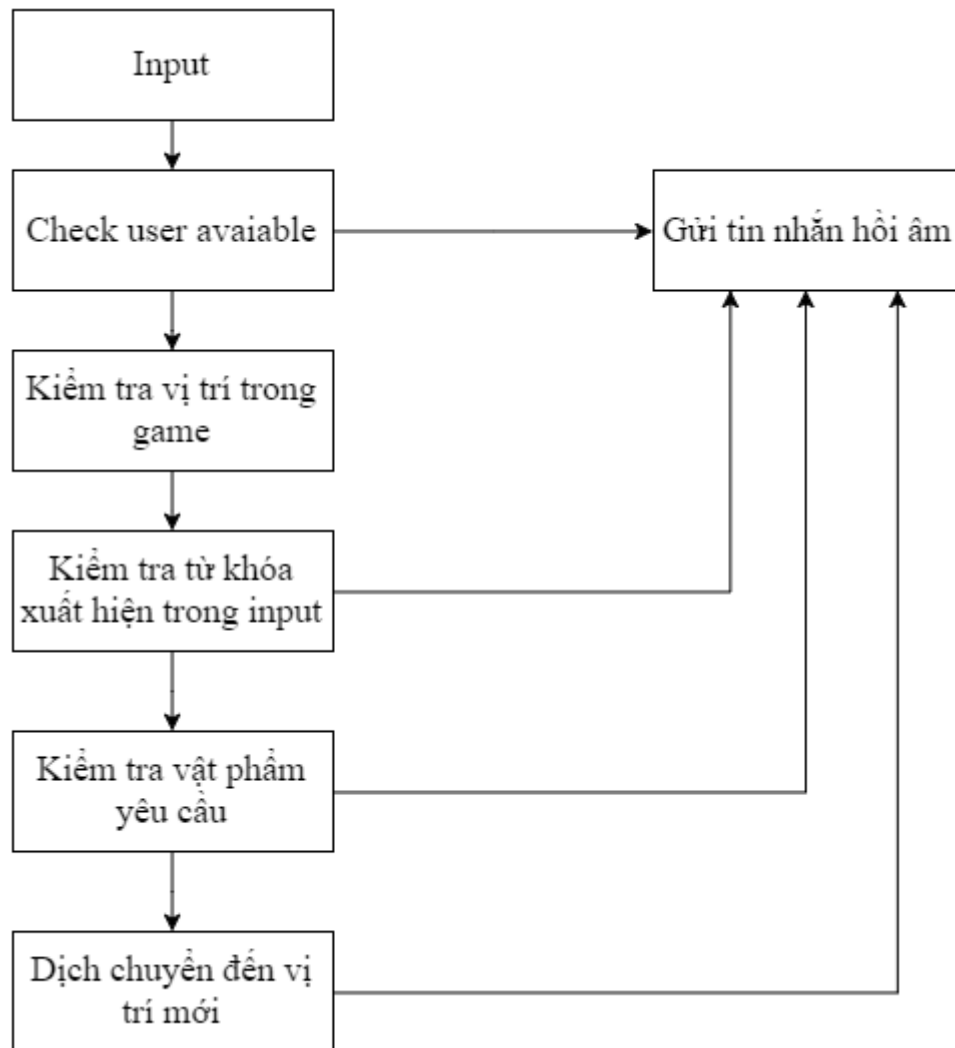
Hình vẽ 3.45: Sử dụng 5 Treasures tìm được để chiến thắng game



Hình vẽ 3.46: Sử dụng @hardreset để chơi lại

3.3.5. Sơ đồ Flow Game

Thể loại text-based Game không hề có giao diện. Nên việc thiết kế flow cho game cực kỳ khó khăn và mất rất nhiều thời gian để có được kết quả như mong muốn, với tính tái sử dụng source code cao để có thể thiết kế nhiều game trên một source code duy nhất.



Hình vẽ 3.47:Flow của pyZork

Giải thích sơ đồ:

- **Input:** Ở bước này, pyZork sẽ nhận các tin nhắn từ người chơi về và tiến hành phân tích cú pháp có trong tin nhắn. Sau đó chốt lọc thông tin lấy ra tin nhắn và id của người chơi rồi gửi sang bộ phận xử lý người chơi.
- **Check user available:** Trong bước này ta kiểm tra người chơi đã tạo và chơi game trước đó chưa? Nếu chưa sẽ gửi tin nhắn thông báo cú pháp tạo tài khoản mới để chơi game. Hoặc nếu rồi sẽ tiến hành bước tiếp theo là kiểm tra vị trí hiện tại của người chơi trong game.

- **Kiểm tra vị trí trong game:** Kiểm tra vị trí người chơi trong Game là bước bắt đầu để xác định xem người chơi có nhập đúng hành động để đi tiếp, hoặc sai và yêu cầu nhập lại. Đây là một bước lấy ra thông tin người chơi hiện tại: Tên, vị trí hiện tại, item đang có, trạng thái hiện tại... và chuẩn bị cho việc hồi đáp các trường hợp đặc biệt để sang cảnh mới.
- **Kiểm tra từ khóa xuất hiện trong input:** Mỗi hành động của người chơi (mỗi tin nhắn) sẽ mang một ý nghĩa riêng. Mỗi liên kết scene trong Game cũng sẽ yêu cầu một số từ khóa nhất định để được truyền từ Scene này sang Scene khác. Ở bước này ta tiến hành kiểm tra tất cả các Scene có khả năng liên kết với vị trí hiện tại của người chơi, trong mỗi Scene liên kết đó ta kiểm tra và đối chiếu các từ khóa để vào Scene có tương đồng với từ khóa mà người chơi đã nhập vào. Nếu có Scene liên kết nào chứa từ khóa trùng với từ khóa mà người chơi đã nhập, ta tiến hành phân tích yêu cầu vật phẩm của Scene đó. Nếu duyệt tìm tất cả các Scene liên kết nhưng không thấy kết quả trùng. Điều đó có nghĩa là hành động của người chơi ở Scene này không được chấp nhận → Ta gửi tin nhắn thông báo người chơi đã nhập sai và thử với một số hành động khác.
- **Kiểm tra vật phẩm yêu cầu:** Mỗi Scene trong game có thể sẽ cần một hoặc nhiều vật phẩm để unlock và đi đến Scene đó. Các vật phẩm này có thể sẽ được thu thập qua việc di chuyển, tìm hiểu xung quanh các vị trí khác và thu thập. Điều này làm game trở nên khó hơn, thử thách nhiều hơn. Đồng thời khiến người chơi thật sự phải suy nghĩ logic và tinh ý để có thể chiến thắng trò chơi.
- **Dịch chuyển đến vị trí mới:** Sau khi kiểm tra tất cả các yếu tố trên và thỏa mãn điều kiện. Người chơi sẽ được đưa đến cảnh Game mới. Nơi này có thể sẽ có vật phẩm. Có thông tin để tìm kiếm đường đến các góc ngách game khác.
- **Gửi tin hồi âm:** Mỗi trường hợp đều có một kết quả trả về cho người chơi tương ứng để hỗ trợ người chơi đi qua các màn chơi.

3.3.6. Các cấu trúc dữ liệu trong Game

- **PlayerData:** Là đối tượng thao tác chính của toàn bộ Game. Những thông số của Player được server lưu và xử lý trong một phiên gửi tin nhắn từ client lên server và nhận lại tin nhắn từ server gửi cho client.
 - Id: ID của người chơi, ID này trùng với ID Facebook của người chơi.
 - Name: Tên người chơi đặt khi bắt đầu Game
 - CurrentSceneId: Địa chỉ hiện tại mà người chơi đang đứng trong Game.
 - ListItem: Danh sách các vật phẩm hiện có trong người của player. Một số vật phẩm quan trọng sẽ dùng để mở cửa các Scene khó, một số vật phẩm làm chìa khóa chiến thắng Game.
 - ListScene: Danh sách các Scene trong trò chơi hiện tại của người chơi.
- **ItemData:** Là đối tượng lưu thông tin của các vật phẩm trong game.
 - Id: Mỗi vật phẩm chứa một Id riêng biệt để phân biệt với các vật phẩm khác.
 - Name: Tên vật phẩm để hiển thị trong Game.
 - Description: Thông tin bên cạnh của vật phẩm, nói về truyền thuyết, tác dụng...
- **SceneData:** Mỗi phân cảnh trong Game được gọi là một Scene. Mỗi scene đều có những thông tin sau
 - Id: Id của Scene dùng để đánh dấu, phân biệt.
 - SceneName: Tên của Scene hiện tại, dùng để hiển thị.
 - Description: Thông tin của Scene, gợi ý về các hướng lựa chọn trong Game.
 - ItemInScene: Id của vật phẩm có trong Scene. Người dùng có thể nhặt được.
 - ListRequireItems: Danh sách các vật phẩm mà Scene yêu cầu để được mở.

- ListConnection: Danh sách các liên kết Scene với các Scene xung quanh (Các Scene này có khả năng đi qua Scene khác)
- **SceneConnectionData:** Đây là một thông tin cực kỳ quan trọng, thông tin này quyết định người chơi có đi sang một cảnh trong Game được hay không. Lưu tất cả thông tin về liên kết giữa các màn chơi.
 - SceneId: Thông tin Id của scene đang được liên kết
 - ListKeyword: Danh sách các từ khóa, từ liên quan mà người chơi có thể nhập vào để được chuyển qua cảnh Game mới.

3.3.7. Xử lý dữ liệu người chơi

Trong pyZork, thông tin người chơi được lưu ở dạng JSON. Sử dụng linh hoạt thư viện *jsonpickle* hỗ trợ rất nhiều trong việc encode và decode dữ liệu thành JSON và ngược lại.

Thông tin về các phân cảnh trong Game được lưu trong tập tin **DefaultScene**. Ta tạo một file Json chứa thông tin một danh sách các Scene trong Game liên kết với nhau để tạo ra default scene data.

```
listDefaultScene = list()
default_scene_data_path = 'db/defaultscene/default_scene.json'

listDefaultScene.append(
    SceneData(0,
        'Living room of House',
        'You are standing in an living room of a white house.' +
        ' There is a small mailbox on the table here.',
        -1,
        [],
        [SceneConnectionData(1, ['take mailbox'])]
    )
)
```

Hình vẽ 3.48: Cách tạo ra danh sách scene trong Game

```
list_scene_json = jsonpickle.encode(listDefaultScene)
file = open(default_scene_data_path, 'w')
file.write(list_scene_json)
file.close()
print('Saved Default!')
```

Hình vẽ 3.49: Lưu thông tin default scene

Làm tương tự với thông tin các vật phẩm trong Game ta cũng tạo ra được file chứa thông tin các vật phẩm có trong Game.

```
default_item_factory_path = 'db/itemfactory/item_factory.json'
listItemFactory = list()
listItemFactory.append(ItemData(0, 'Metal Element', 'Metal Element Description'))
listItemFactory.append(ItemData(1, 'Wood Element', 'Wood Element Description'))
listItemFactory.append(ItemData(2, 'Earth Element', 'Earth Element Description'))
listItemFactory.append(ItemData(3, 'Water Element', 'Water Element Description'))
listItemFactory.append(ItemData(4, 'Fire Element', 'Fire Element Description'))
list_item_json = jsonpickle.encode(listItemFactory)
file = open(default_item_factory_path, 'w')
file.write(list_item_json)
file.close()
print('Saved Default!')
```

Hình vẽ 3.50: Tạo danh sách các Item trong Game

Trong game, khi người chơi tạo một tên nhân vật mới sẽ gọi hàm tạo dữ liệu player mới.

```
def get_default_scene_data(self):
    scene_data = open(default_scene_data_path, 'r')
    result = jsonpickle.decode(scene_data.read())
    scene_data.close()
    return result
pass
```

Hình vẽ 3.51: Lấy ra thông tin default scene trong Game.

```

def create_user_data_file(self, userId, user_name):

    player_data = PlayerData(userId, user_name, 0, [], self.get_default_scene_data())
    player_json = jsonpickle.encode(player_data)

    file_name = userId + '.json'
    file_path = rootPath + file_name

    file = open(file_path, 'w')
    file.write(player_json)
    file.close()
    return 'Welcome to pyZork ' + player_data.Name + '.\n' + \
        'Use command @howto to learn play this game :D'
    pass

```

Hình vẽ 3.52: Tạo thông tin người chơi với định dạng json

```

save_file = open(path, 'w')
player_object.CurrentSceneId = connection_scene.SceneId
curr_sceneid = player_object.CurrentSceneId
scene_item_id = scene_data_connected.ItemInScene

if scene_item_id is not -1:
    player_object.ListItem.append(scene_item_id)
    scene_data_connected.ItemInScene = -1
    player_object.ListScene[connection_scene.SceneId] = scene_data_connected

player_json = jsonpickle.encode(player_object)
save_file.write(player_json)
save_file.close()

```

Hình vẽ 3.53: Xử lý thông tin người chơi lúc chuyển cảnh.

3.3.8. Xử lý ngôn ngữ tự nhiên

ChatterBot là một thư viện cung cấp khá đầy đủ các hàm hỗ trợ tạo đoạn hội thoại với trí thông minh nhân tạo từ Natural Language Toolkit. Ngoài ra, trong quá trình làm việc với người dùng, ChatterBot còn tự động học các đoạn hội thoại giúp cơ sở dữ liệu cho việc trả lời câu hỏi đa dạng, phong phú nhiều hơn.

Tạo ChatterBot: Để có thể thao tác, lấy tạo hội thoại tự động. Ta cần tạo một đối tượng ChatterBot trong source code.


```
from chatterbot.trainers import ListTrainer

self.chatbot = ChatBot(
    'pyZork'
)
self.chatbot.set_trainer(ListTrainer)
# Train based on the english corpus
self.chatbot.train("chatterbot.corpus.english")
```

Hình vẽ 3.54: Tạo một ChatterBot trong source code

```
# user wrong input
return self.chatbot.get_response(user_input).text
```

Hình vẽ 3.55: Cách nhận lại response cho người chơi

Chương 4. CÀI ĐẶT MINH HỌA

4.1. Yêu cầu

Server:

- Môi trường: python 3.5
- Framework sử dụng: Flask 0.12

Client:

- Nền tảng: Facebook Messenger. Có thể sử dụng qua duyệt web hoặc trên thiết bị di động

4.2. Hướng dẫn cài đặt

Server:

- Bước 1: Tạo ứng dụng facebook
 - o Truy cập vào địa chỉ: <https://developers.facebook.com/apps/>
 - o Nhấn chọn Add a New App
 - o Điền các thông tin

Create a New App ID

Get started integrating Facebook into your app or website

Display Name

pyZork

Contact Email

huudanh.k7@gmail.com

Category

Entertainment ▼

By proceeding, you agree to the [Facebook Platform Policies](#)

Cancel

Create App ID

Hình vẽ 4.1: Tạo ứng dụng facebook

- Bước 2: Add Product Messenger vào ứng dụng vừa tạo
 - o Trong mục Add Products, tìm và chọn vào Messenger

Messenger
Customize the way you interact with people on Messenger.

Get Started

Webhooks
Webhooks (formerly Real Time Updates) lets you subscribe to changes you want to track and receive updates in real time without having to call the API.

Get Started

Hình vẽ 4.2: Chọn add products trong facebook app

- Trong mục **Token Generation** chọn một Fanpage và copy token. Kéo xuống bên dưới click chọn **Add to submission**

Token Generation

Page token is required to start using the APIs. This page token will have all messenger permissions even if your app is not approved to use them yet, though in this case you will be able to message only app admins. You can also generate page tokens for the pages you don't own using Facebook Login.

Page

Select a Page ▼

Page Access Token

You must select a Page to generate an access token.
[Create a new page](#)

Hình vẽ 4.3: Box Token Generation

Token Generation

Page token is required to start using the APIs. This page token will have all messenger permissions even if your app is not approved to use them yet, though in this case you will be able to message only app admins. You can also generate page tokens for the pages you don't own using Facebook Login.

Page

Katsu ▼

Page Access Token

EAAQeCfPQPDQBAJrLfSZCCh2ZBqFkZCLaG0ScBijHXLUhtxiPP5w9ZBvtGaWb8M1jPZAG9ltRICDRnC

[Create a new page](#)

Hình vẽ 4.4: Chọn fanpage và copy token

☐ **pages_messaging** Add to Submission
Enables your app to send and receive messages using a Facebook Page.

☐ **pages_messaging_subscriptions** Add to Submission
Enables your app to send messages using Facebook Pages at any time after the first user interaction. This permission can't be used to send advertising or promotional content.

☐ **pages_messaging_phone_number** Add to Submission
Enables your app to match phone numbers you already have to Messenger accounts. In order to use this permission, your app must be already approved for the Send/Receive API and based in the US.

Hình vẽ 4.5: Add to submission facebook messenger

- Trong source code, mở file **server.py** chọn dòng thứ 12 và dán token đã copy cho biến **ACCESS_TOKEN**

```
app = Flask(__name__)
pyBot = GamePlay()

ACCESS_TOKEN = 'EAAEgy21czGwBAFZCZArhZAR8BimmrCr3ZA9x42GG8M1s1ZAGopN3vgMtNE4pf1Be4sC4DKXLZCffKQ2'
VERIFY_TOKEN = 'Danhth'
```

Hình vẽ 4.6: Dán token đã copy vào giá trị biến ACCESS_TOKEN

- Bước 3: Run server
 - Bật **ngrok** bằng cách gõ ngrok http 5000 trong cửa sổ cmd của project và sau đó chạy **server.py**

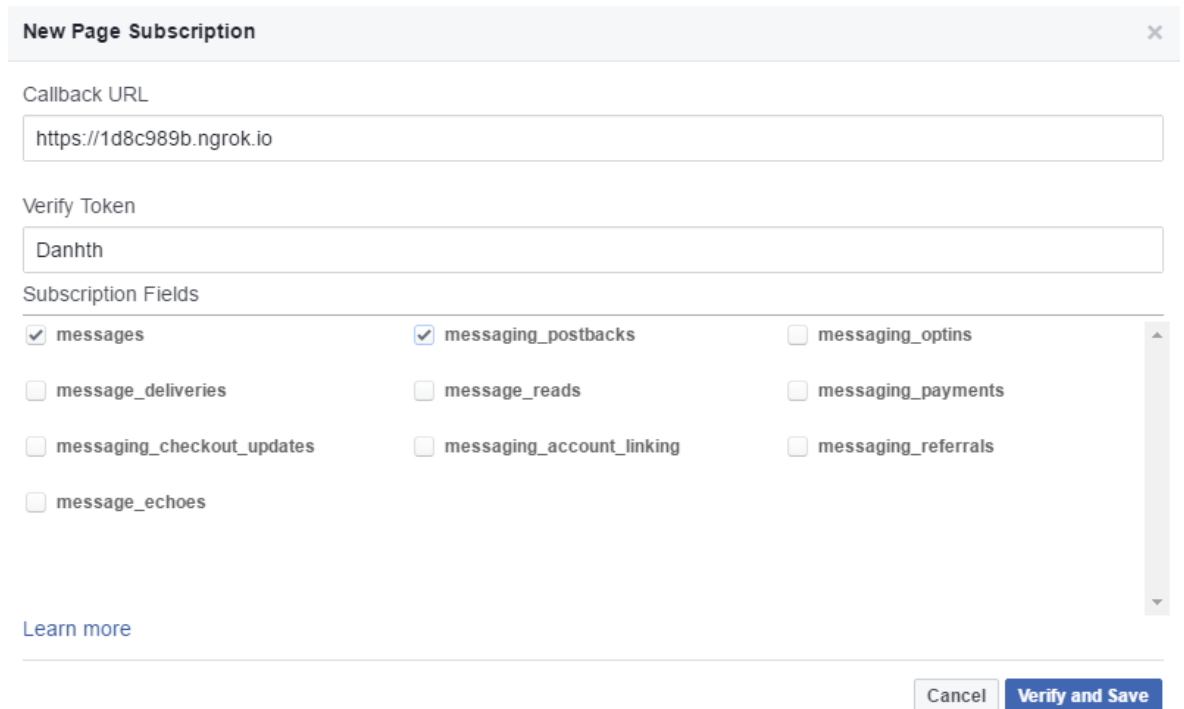
```
Session Status      online
Version             2.1.18
Region              United States (us)
Web Interface        http://127.0.0.1:4040
Forwarding           http://d2394366.ngrok.io -> localhost:5000
Forwarding           https://d2394366.ngrok.io -> localhost:5000

Connections          ttl    opn    rt1    rt5    p50    p90
0                   0      0.00   0.00   0.00   0.00
```

Hình vẽ 4.7: ngrok http 5000

- Chạy file **server.py**
- Bước 4: Config webhook

- Copy link **Forwarding https** trong cửa sổ ngrok sau đó mở giao diện facebook app, click setup webhook.



New Page Subscription

Callback URL
https://1d8c989b.ngrok.io

Verify Token
Danhth

Subscription Fields

<input checked="" type="checkbox"/> messages	<input checked="" type="checkbox"/> messaging_postbacks	<input type="checkbox"/> messaging_optins
<input type="checkbox"/> message_deliveries	<input type="checkbox"/> message_reads	<input type="checkbox"/> messaging_payments
<input type="checkbox"/> messaging_checkout_updates	<input type="checkbox"/> messaging_account_linking	<input type="checkbox"/> messaging_referrals
<input type="checkbox"/> message_echoes		

[Learn more](#)

Cancel Verify and Save

Hình vẽ 4.8: Config webhook

- Nhấn Verifi and Save
- Vào phần setup Roles, chọn thêm tester là tài khoản facebook sử dụng để test. Tài khoản test này cần khác với tài khoản tạo app.

Client:

- Bước 1: Đăng nhập Facebook tài khoản đã được thêm vào test ứng dụng
- Bước 2: Vào fanpage vừa thêm cho ứng dụng ở bước 2
- Bước 3: Bắt đầu chat và chơi game

4.3. Cách chơi game

Thể loại text-based game hết sức dễ chơi. Nhưng rất khó để hoàn thành và chiến thắng.

Cách chơi đơn giản là nhấn tin một đoạn text cho chatbot và đợi nó hỏi âm trả lời. Nếu người chơi đủ tinh ý và suy luận sẽ tìm ra được từ khóa qua màn cho game.

Ví dụ:

- Khung cảnh hiện tại: Bạn đang đứng trong một căn phòng trống. Trên bàn có một bức điện. Bạn sẽ làm gì?
- Để qua được bước này người chơi cần suy luận và tìm cách lấy được bức thư đó để đọc. Và sẽ tìm được lời giải cho các màn tiếp sau

Một số gợi ý khi chơi pyZork:

- Nên dùng động từ, các mẫu từ ngắn gọn. Ví dụ: take gem, go east, back home...
- Nên suy luận và tìm cách nhớ đặc điểm của vị trí mình đi qua.
- Dùng @status để xem lại trạng thái, vị trí hiện tại của người chơi.

Đối với người muốn tự tạo một text-based game riêng biệt. Có thể sử dụng 2 file 'create_default_scene.py' và 'create_item_factory.py'. Custom 2 file theo ý của mình. Và sau đó thử chơi với game mình vừa tạo ra.

Chương 5. KẾT LUẬN, HƯỚNG PHÁT TRIỂN

5.1. Kết luận

Từ những phân tích, thiết kế, mô hình và những phương pháp tiếp cận nêu trên đã giúp em xây dựng đạt yêu cầu tìm hiểu công nghệ Conversational Chatbot và làm ứng dụng game text-based game có tựa đề pyZork.

Facebook Messenger của Facebook với một lượng người dùng active hàng tháng cao đến một tỷ người dùng / tháng. Điều này góp phần không nhỏ vào tiềm năng thành công của Chatbot trong tương lai.

5.1.1. Đánh giá chung

- Làm được
 - Tìm hiểu, tiếp cận Conversational Chatbot
 - Làm việc được với ngôn ngữ Python trên framework Flask
 - Sử dụng linh hoạt Json để làm cơ sở dữ liệu trong game
 - Mở rộng kiến thức về thiết kế game. Là một bước tiến mới để cố gắng hơn trong việc hiện thực ước mơ làm Game Designer.
 - Xây dựng game pyZork. Là một game text-based game thú vị và đầy nguy hiểm.
- Chưa làm được
 - Hệ thống AI trả lời thông minh theo màn chơi.
 - Bảng điểm so sánh giữa các người chơi.
 - Chưa mở rộng được các tính năng thật giống với game giao diện.

5.1.2. Bài học

Sau khi hoàn thành khóa luận, em đã học được nhiều điều hỗ trợ bổ ích cho con đường sự nghiệp sau này.

- Hiểu quy trình phát triển game từ giai đoạn tìm hiểu thị trường, phân tích, ý tưởng, thiết kế và hoàn thiện sản phẩm
- Hiểu sâu hơn về Facebook API, Facebook Messenger

- Hiểu rõ và làm việc thành thạo với ngôn ngữ Python, Framework Flask
- Tận dụng được các nguồn lực có sẵn là thư viện về Xử lý ngôn ngữ tự nhiên, để đưa vào game linh hoạt. Giúp game thân thiện hơn.

5.2. Hướng phát triển

pyZork là một bước khởi đầu cho việc tiếp cận với công nghệ Conversational Chatbot. Có thể dự đoán rằng con tinh thần pyZork sẽ không được phát triển mạnh và nhiều người đón nhận nhưng với kiến thức và kinh nghiệm làm việc với Conversational Chatbot ta hoàn toàn có thể tạo ra những sản phẩm Chatbot tốt hơn, ưu việt hơn để hỗ trợ cuộc sống người dùng.

Một số tính năng cho hướng phát triển pyZork:

- Thêm nhiều map game, cho phép người chơi lựa chọn map và chơi. Lúc này game sẽ rất đa dạng và không còn riêng 1 thế giới pyZork nữa.
- Xây dựng nhiều vật phẩm hơn, nhiều giai thoại, truyền thuyết hơn cho sản phẩm thêm hấp dẫn – kỳ bí.
- Tối ưu thư viện Chatterbot để nâng cấp, cải tiến phần xử lý ngôn ngữ tự nhiên trong game.
- Tích hợp quảng cáo game, bán hàng trong quá trình chơi game.
- Đưa game này ra các nền tảng Chatbot khác, không chỉ riêng Facebook Messenger.

TÀI LIỆU THAM KHẢO

- [1.] Conversational Chatbot: http://www.webopedia.com/TERM/C/chat_bot.html
- [2.] Facebook Messenger: <https://github.com/hult/facebook-chatbot-python>
- [3.] Flask: <http://flask.pocoo.org/>
- [4.] History of Chatbot (1): <http://futurism.com/images/the-history-of-chatbots-infographic/>
- [5.] History of Chatbot (2): <http://venturebeat.com/2016/08/15/a-short-history-of-chatbots-and-artificial-intelligence/>
- [6.] Natural Language ToolKit: <http://www.nltk.org/>
- [7.] Ngrok: <https://ngrok.com/>
- [8.] Programming skill: <http://stackoverflow.com/>
- [9.] Python v3.5: <https://www.python.org/>