ASHESI UNIVERSITY

ASHESIBOT: A CHATBOT FOR HANDLING ASHESI FACEBOOK MESSENGER PLATFORM

APPLIED PROJECT

B.Sc. Computer Science

Primerose Ruvimbo Katena

2019

ASHESI UNIVERSITY

ASHESIBOT: AN AUTOMATED CHATBOT FOR HANDLING ASHESI FACEBOOK MESSENGER PLATFORM

APPLIED PROJECT

Applied Project submitted to the Department of Computer Science, Ashesi University College in partial fulfilment of the requirements for the award of Bachelor of Science degree in Computer Science

Primerose Ruvimbo Katena

April 2019

DECLARATION

I hereby declare that this applied project is the result of my own original work and that no part of it has been presented for another degree in this university or elsewhere.

Candidate's Signature:

………………………………………………………………………………………

Candidate's Name:

………………………………………………………………………………………

Date:
………………………………………………………………………………………

I hereby declare that preparation and presentation of this applied project were supervised in accordance with the guidelines on supervision of applied project laid down by Ashesi University College.

Supervisor's Signature:

………………………………………………………………………………………

Supervisor's Name:

………………………………………………………………………………………

Date:
………………………………………………………………………………………

ACKNOWLEDGEMENT

I would like to thank God for how far he has brought me. I would also want to thank my supervisor for his unwavering support towards this project. Many thanks to my beloved, family and Ashesi colleagues for their keen encouragement in making this project successful.

ABSTRACT

As each day passes, new ways of improving our lives through technology emerge. Companies keep on exploring ways of reaching out to their customers and other stakeholders in a way that is more comfortable and efficient. Regardless of their exploration, the problem of presenting information in a way that allows quick retrieval of information by users remains a challenge. To solve the problem of information retrieval, companies have implemented search bars on their website and other functionalities. Unfortunately, search bars have proven to be inefficient because they bring everything that matches with the input of the user. Additionally, others have implemented live chat systems to help mitigate the problem of information retrieval. However, live chats are time-consuming especially when there are few workers behind the live chats systems providing services to numerous customers who demand immediate attention. These problems necessitate a novel approach for retrieving information that meets the needs of businesses and individuals; using chatbots.

The idea of chatbots is not new, but it is becoming more refined and useful with the application of machine learning to solve the problem of information retrieval. The advent of chatbots on websites and social media platforms is fast spreading, helping with a 24/7 retrieval of information. The use of automated chatbots to handle a conversation with a business's customers is now a trend because it allows for quick retrieval of information with little or no support from the staff members. Use of chatbots gives room for staff members of an organization to focus on more significant issues. Customers equally benefit from using automated chatbot system as they no longer need to be in a queue. To sum up, this paper seeks to solve the problem of information retrieval for Ashesi Facebook messenger subscribers using an automated chatbot.

Table of Contents

LIST OF FIGURES

Chapter 1: Introduction

Chapter Overview

This chapter introduces the AshesiBot project. It focuses on the project's background, how Artificial Intelligence (AI) is improving the lives of people around the world, the strengths and weaknesses of traditional methods that exists for information delivery and customer engagement. Based on the weaknesses of the traditional methods and the current implementation used by Ashesi in retrieving information, a problem statement is stated. Also, this chapter highlights the need for Ashesi to use an automated chatbots to address the problem statement. Many companies in different sectors are drifting away from the traditional methods of engaging with online customers such as emails, live chats to automated chatbots. More generally, the aim of this project is to design and implement an easier information retrieval system that consume less time and reduces the hustles stakeholders of Ashesi University goes through in an online environment with the help of an automated chatbot.

The related work is based on how the Artificial Intelligence is solving the problem of information retrieval through automated chatbots. Many companies and organizations are embracing automated chatbots in different sectors such as chatbot to handle financial queries of a school and a chatbot for university frequently asked questions are referenced in the related work. Part of related work includes novel chatbot that act as personalized travelling agents to improve the experience of a travelers. The related work also touches on the merits chatbots are bringing to improve the online customer experience.

1.1 Background

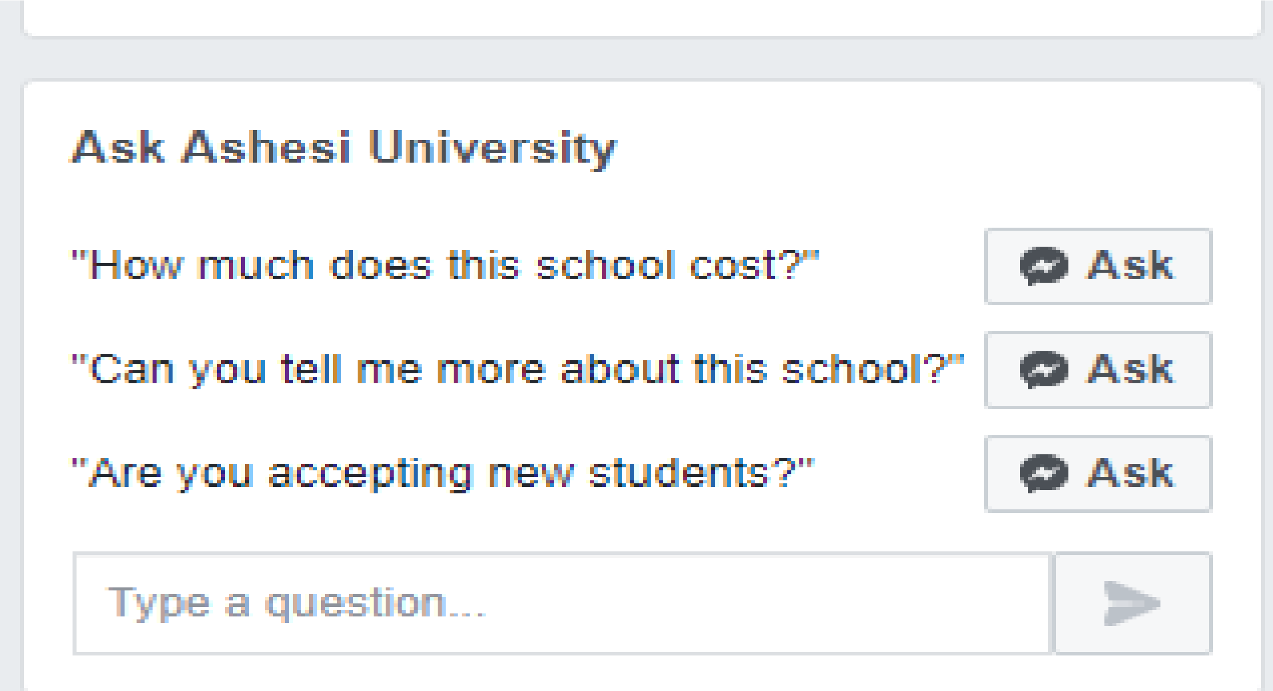
The technology exists to make human lives easier. Thanks to research in science and technology, many fields have proven beneficial to our society. Notable amongst them is the area of Artificial Intelligence (AI). Indeed, the application of AI has not only helped companies to deliver quality customer services but also, created new opportunities for businesses to create more wealth. Despite the benefits that comes with the introduction of AI, companies still struggle with content (information) delivery and customer engagement. As a result of competition and continuous improvement, different companies are always searching for better ways of presenting information to their customers or targeted group in a way that fosters friendly interaction and allows for immediate feedback.

The traditional approach to achieving the problem mentioned above is to implement websites that allow customers to reach out in case of any queries. Thankfully, with the advent of technology in the field of AI, companies with websites are evolving from the traditional email feedback and phone calls to live chats that allow for human-to-human interaction. It is worth mentioning that the emerging interest in live chats or virtual assistant is due to the new trend in communication— texting. It only makes logical sense in an internet driven world, to implement a communication system (with the help of automated chatbots) that can leverage on the existing technology such as social media and online texting.

The live chat support is a new value-added system (VAS) used by the customer care department of a service industry to respond and engage with client's queries. Customers using the live chat support system end up queuing as they wait for the response from the customer service agents [2]. The drawback of implementing live chatbots is that not all customers have enough time to wait for other customers to be served before their turn [1]. Having understood this, it is worth exploring better ways of presenting information in a way that is interactive, efficient and allow quick response for effective communication.

1.2 Problem Statement

Ashesi University is an educational institution established with the mission of instilling ethics and entrepreneurial skills into its members. This institution however is crippled with poor information retrieval system, especially on its website. Retrieving information from the website can be very time consuming and not easy for users. To make easy retrieval of information about Ashesi online, Ashesi has a Facebook page with over thirty-seven thousand (37, 0000) followers. For instance, when a prospective student wants a quick information about Ashesi, users can either reach out through email or ask questions at Ashesi Facebook page messenger or ask someone from Ashesi community with the correct information or visit the Ashesi website. To allow quick retrieval of information on Ashesi Facebook page, Ashesi has created three buttons with only three questions out of the many frequently asked questions. These buttons allow the user to easily get information they need, otherwise, they will have to type in their question. Below is a snapshot of the three questions that are currently running on Ashesi Facebook page:



Ask Ashesi University

"How much does this school cost?"

"Can you tell me more about this school?"

"Are you accepting new students?"

Figure 1. 1 Ashesi Facebook messenger

The Ashesi public relations (PR) takes, on average, about a week to respond because they are committed to other work besides responding to messages on their social media platforms. Therefore, users are not receiving the information they want on timely. Sometimes responses from the PR comes when they no longer need it.

The option of visiting the Ashesi's website to retrieve information can be expensive sometimes. This approach is time-consuming as one must navigate across several web pages before one can access the information they want. Added to this problem is the issue of inadequate information on the website. For instance, one might want a quick response to the location of Dennis Owusu, a faculty member's office. This information, however, is not found on the website.

Another instance is when an external user tries searching for when Ashesi was commenced using the search bar on Ashesi's website. In general, people will key in the question in different formats, but bearing the same semantic meaning. As a result of this, the website will return to the user, all pages that bear the phrase typed into the search bar. Therefore, one must scroll through all the output returned to get exactly what they searched for on the website. This retrieved information is poorly presented to a user, as you may notice. It is quite sad to note that if the wording of a user's query doesn't match with the text on the website, they will more likely not get a response even though what they searched for is available on their website but in a different word structure or morphology.

Having been painted with the above problem, this project seeks to provide readily available information to online users in a way that mimics the human conversation and is less time consuming, thus, a chatbot. The goal of the Ashesi support bot (AshesiBot) is to automatically and accurately provide response to queries posed to it, within Ashesi context. This solution allows to shorten response time which this paper believes will be favorable for the Ashesi's stakeholders. The proposed implementation is a chatbot system that responds to questions posed to Ashesi on their Facebook messenger platform.

1.3 Related Work

Several companies are implementing chatbot to meet their specific needs. The University of Sydney implemented a chatbot that handles financial queries as a way of reducing the burden of the work in the finance department [3]. A chatbot for university-related FAQs was implemented in 2017 using Artificial Intelligence and Latent Semantic

Analysis to handle parents and prospective students' queries concerning college admission and academics [2]. It is clear that chatbot are found useful in the education domain.

In 2017, researchers proposed a system architecture that handles the social media grievances customers send concerning a company's product or service they offer and initiate a conversation through a chatbot whenever the complaints need and action [1]. This application reduces the number of employees required for the customer service department and gives customers a satisfying treatment.

Research shows that chatbots can be used as a travelling agent for travelers. Using chatbots as a travelling agent can improve the experience of travelers because they are always available, can be personalized to keep track of the traveler's history, preference to make wise recommendations [4].

In the field of medicine, chatbots can be of greater value to the psychiatric patients. Introduction of psychiatric counsellor chatbot that can extract emotions and respond with sympathetic and helpful advice where necessary during a conversation with a patient can enhance the lives of the psychiatric patients [5]. However, this paper proposes to make a chatbot for Ashesi University designed to handle the queries the stakeholders might have, as a way of providing quick response to Facebook messenger users who patronize Ashesi Facebook messenger platform.

1.4 Aim

This project aims to present both the available and unavailable information in an engaging interactive form. It also seeks to provide information that cannot be found on the current platforms such as Ashesi website and Facebook page. More importantly, the project aims to reduce the time and hustles one goes through to access information about Ashesi. The domain for this project is Ashesi mission, vision, Honor code system, majors offered at Ashesi, school fees related issues, scholarships, history of the school and admission.

Chapter 2: System Requirements

2.1 Chapter Overview

This chapter serves to lay down the functional and non-functional requirements that act as a benchmark against the development of the proposed chatbot system. The functional requirement has been further divided into two; user and system requirements, with each of them explained in detail. To give a better view of the problem based on the requirement gathered, three scenarios implemented reveal the stakeholders of Ashesi who are greatly affected by the problem of information retrieval. Moreover, use cases and actor diagrams as implemented below depicts the actions the user does to interact with the system and actions the system interacts with the user. The design constraints are layered out as well. The domain of the questions a user can ask the chatbot is limited to history of the school, honor code, mission and vision of the school, school fees related questions, master's program, student admissions related questions and courses offered as well as location of the lecturer's office.

2.2 Scope

The chatbot to be built is a Facebook Messenger application that will provide the community of Ashesi with real-time, accurate and credible information about the university within the domain specified above. Users will submit their queries or questions to the chatting bot, and a response will be rendered to them (users) in a user-friendly interface.

2.3 Scenarios

Scenario 1

Partson is a freshman at Ashesi University. It is his first week at Ashesi, and he does not know the location of his Introduction to Computing lecturer’s office. Being an inquisitive and curious student, Partson decides to visit his lecturer to inquire more about programming during office hours. The good thing is, he knows that his lecturer's office is in the Engineering block, but he doesn’t know where exactly (the room number). He visited Ashesi Facebook messenger platform and found AshesiBot. He asked the AshesiBot for the location of Mr. Sampah’ s office, and the AshesiBot quickly directed him to Engineering block office number 205-C. Partson then moved towards office number 205-C and found his lecturer and they began the session.

Scenario 2

John is a high school graduate who is looking for a good university to attend in Ghana. He come across Ashesi university website while surfing the internet. John saw many pages with different information about Ashesi. He got tired of going through different webpages on the Ashesi website before getting the information he wants. He decides to visit Ashesi Facebook messenger to have quick responses to his questions concerning the application process at Ashesi University. He asked the AshesiBot; “Steps to apply to Ashesi University?”. The bot responded with vital information about the application process and was even directed with a URL to read further on this topic. He continued his conversation with the AshesiBot asking many questions under topics such as Honor Code System, Scholarship, history of the school. He found that using the AshesiBot was more interactive, timesaving and he felt he was talking to a human being. He wished all universities in Ghana had this software. His interaction with the AshesiBot inspired him to consider Computer Science as a major to take if his application at Ashesi is successful.

Scenario 3

Mercy works in the Public Relations department at Ashesi University. She has many other responsibilities to take care of besides responding to inquiries posted on Ashesi Facebook messenger in the comment section or via Facebook messenger or Instagram. However, she wants the Ashesi Facebook messenger to have the best experience and faster response rate whenever people interact with Ashesi on Facebook messenger. To make that possible, she created three buttons on Ashesi Facebook page with the most frequently asked questions. With that being done, the user clicks on these buttons with the question that is similar to their question instead of typing it. However, this is not enough, and she rarely finds time to respond to people especially during the admission period. Currently, she responds to Ashesi Facebook messenger messages at least once a week because of the workload she has. She wishes to respond to everyone's inquiries on time. Anna believes an automated chatbot can solve her current problem.

2.4 Use Case and Actor Diagram

In this section, diagrams and charts will be used to give a high-level overview of how the various actors of the project interact together.

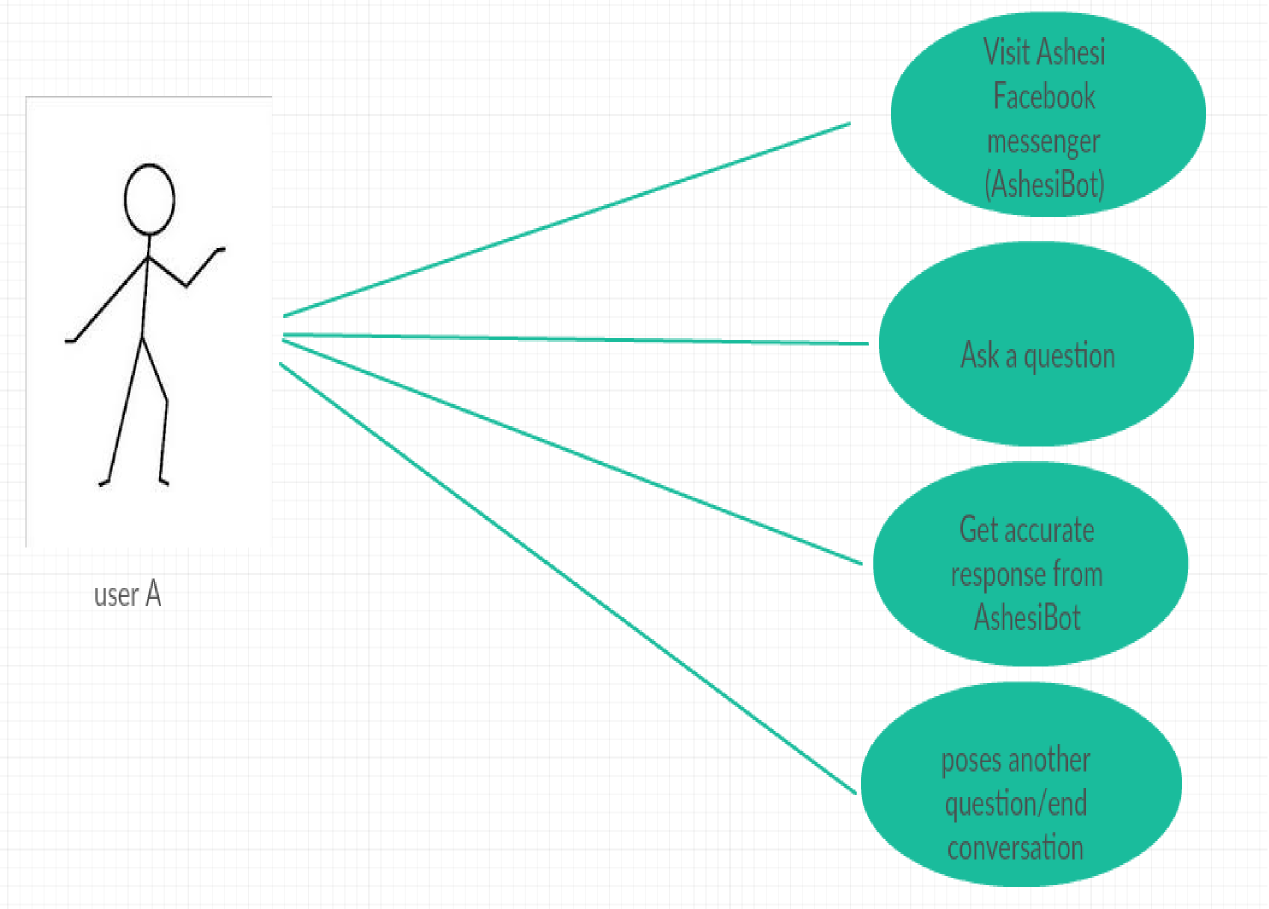

user A
Visit Ashesi Facebook messenger (AshesiBot)
Ask a question
Get accurate response from AshesiBot
poses another question/end conversation

Figure 2. 1: User case (user actions)

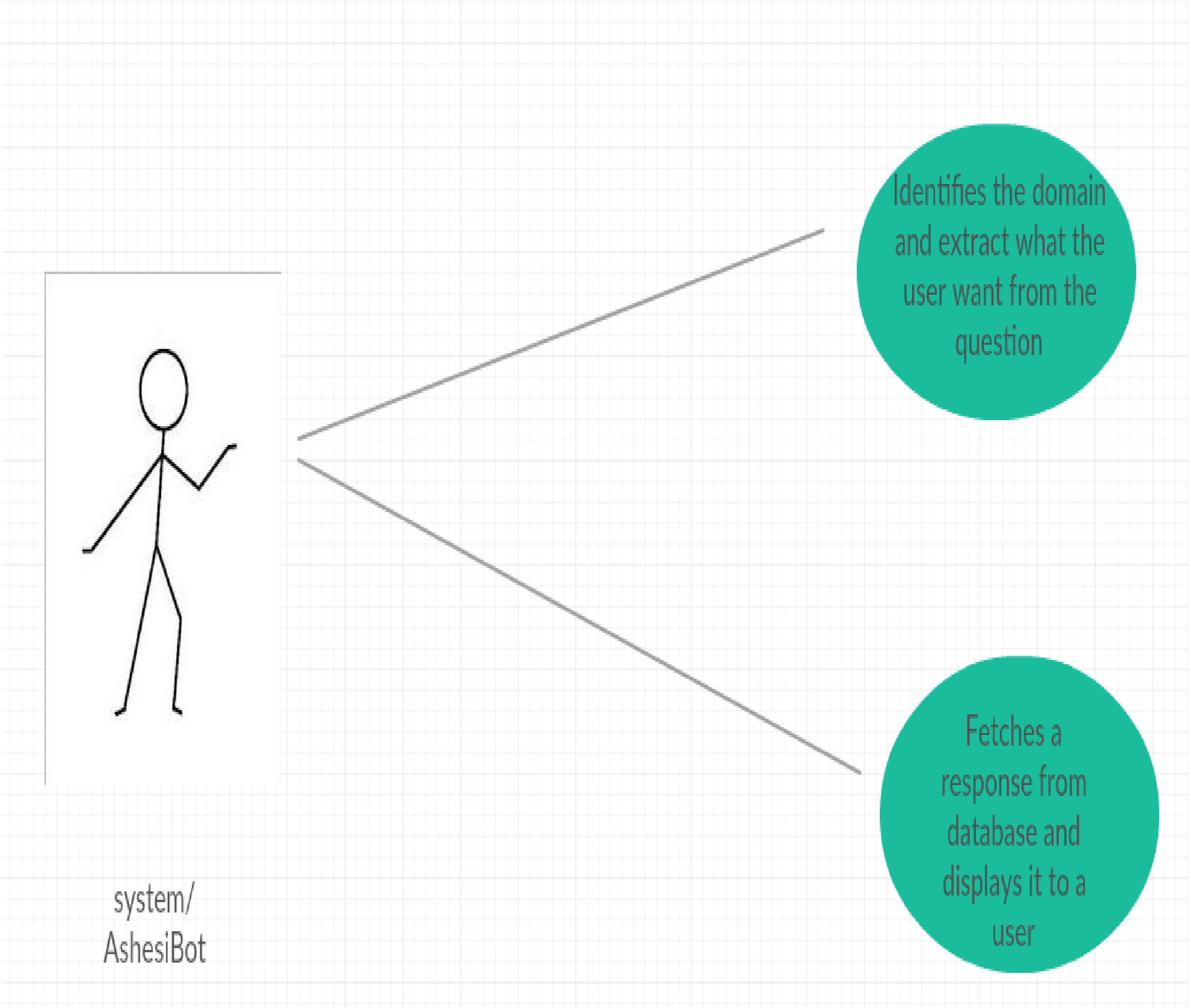

Identifies the domain and extract what the user want from the question
Fetches a response from database and displays it to a user
system/ AshesiBot

Figure 2. 2 Use Case diagram (System Actions)

2.5 Design Constraints and Assumption

In designing the software, the following assumptions were made:

I. The application will only be accessed on Facebook Messenger.

II. It is assumed here that the user is literate; can read and write.

III. The user has an adequate internet connection, which will be used to make a request to the chatbot.

IV. It is assumed that the user can only use English to communicate with the chatbot.

V. The user will not use shorthand to communicate with the bot. For instance, idk meaning I don't know.

2.6 Functional Requirements

This section describes the functional requirements of the AshesiBot. It is divided into two sub categories; user requirements and system requirements.

2.6.1 User Requirements

- The user should be able to key in or send their request to the system.
- The user should be able to read their response from the chatbot.
- The user should be able to hold a conversation with the chatbot and have their queries responded.

2.6.2 System Requirements

- The system should be able to provide accurate response to every question asked.
- The system should be accessible on Ashesi Facebook page using Messenger platform.

2.7 Non-Functional Requirements

Performance: The system should have low latency when communicating with a user.

Reliability: The system should always respond in Natural Language (English) that can be understood by humans.

Naturalness: - The chatbot should be able to mimic a natural human conversation.

Chapter 3: System Design and Architecture

3.0 Chapter Overview

This chapter focuses on the system architecture of the AshesiBot. Moreover, it defines the environment in which the AshesiBot operates and AshesiBot components. The architecture is a three-tier architecture which consists of the presentation, application and database layer. The presentation layer consists of the Facebook messenger, where the system and user interact, the application layer is the layer that handles the dialogue. It consists of the routing engine (Python Flask) that routes data among presentation, application and database layers as designed. The Rasa NLU consists of trained data made up of domains and intents. The Rasa NLU identifies the domain and intents of a user message based on the trained. data. The database layer consists of tables. Each table is a domain and attributes consist of the intents name and corresponding responses.

3.1 High Level System Architecture

The overall system architecture, which shows the various components of the software is shown below:

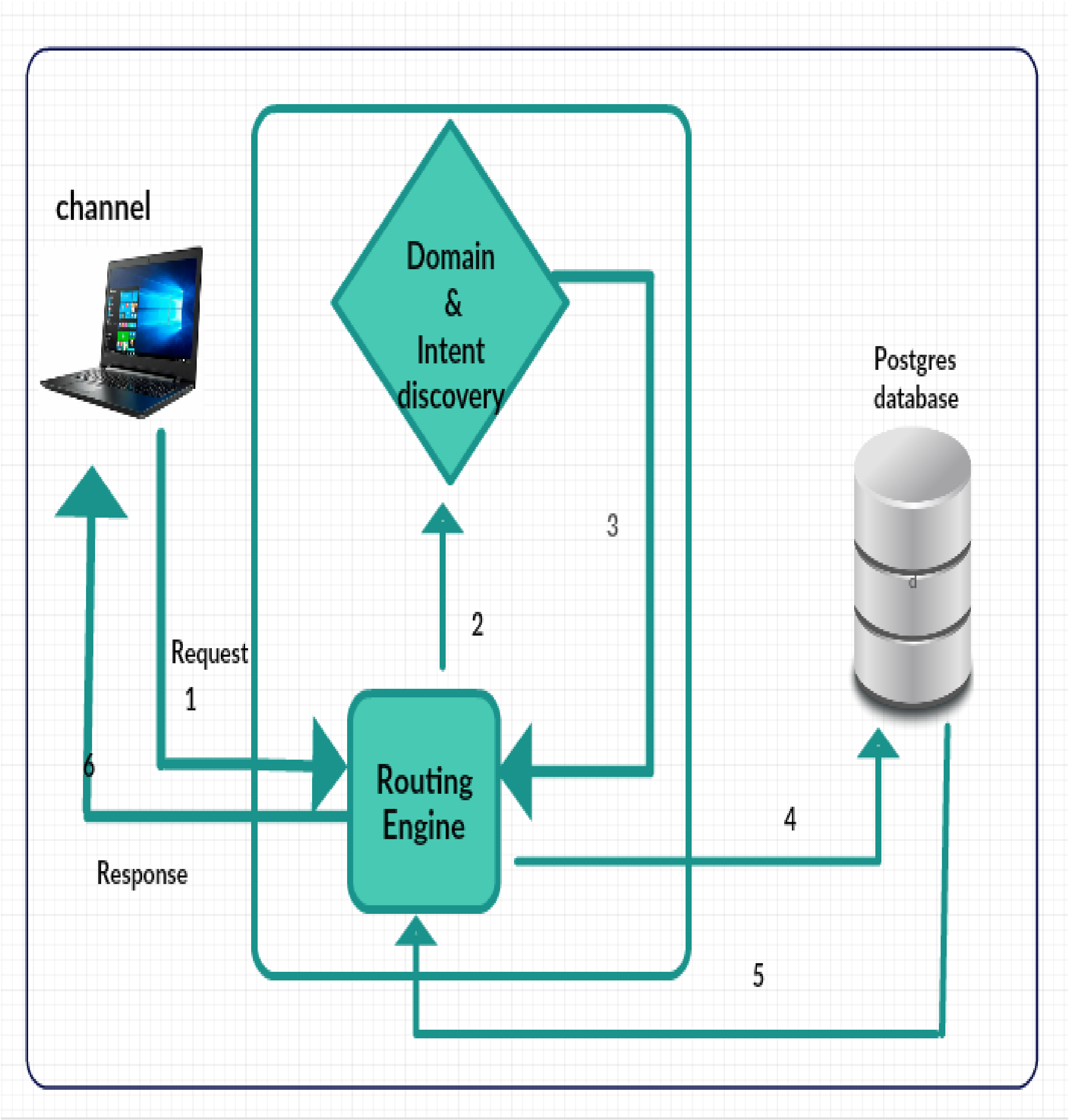


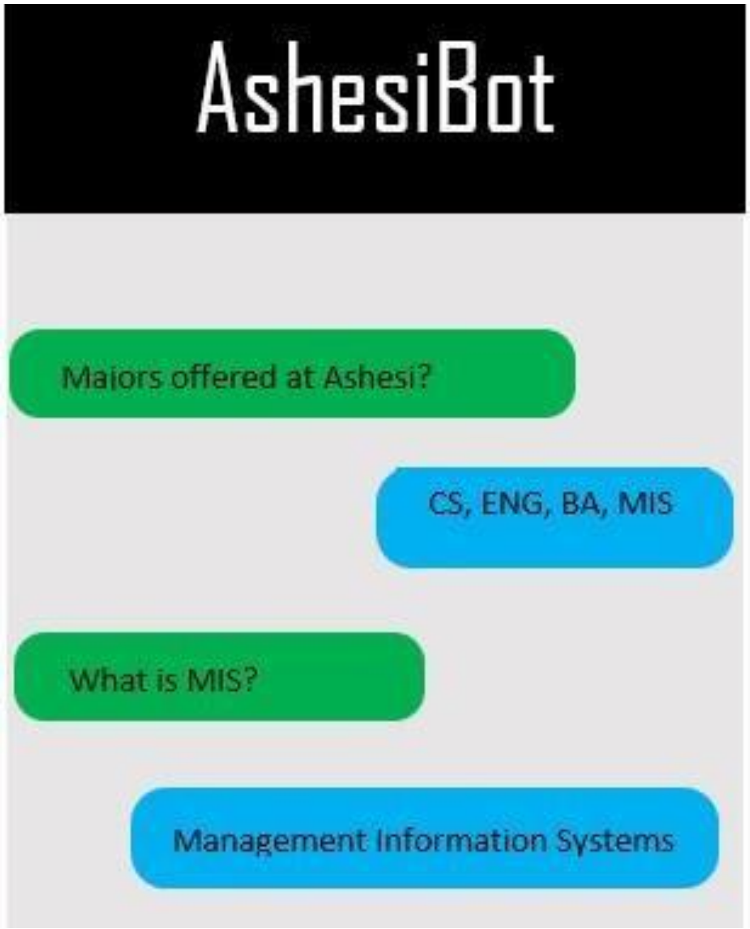
Figure 3. 1: Ashesi software architecture

3.2 The Design of The Architecture

This architecture, as shown above was developed using the three-layer architecture, which allows each layer to be maintained independently (abstracted) from each other. The three-layer architecture consists of the presentation layer, application or business logic layer, and the storage layer. The following section of this chapter gives a detailed information about each layer.

3.2.1 The Presentation Layer

This layer receives input or data from a user and displays the response from the channel. It is designed to be intuitive and clear to the user. More specifically, the presentation layer is the Ashesi Facebook Messenger page (herein AshesiBot), which allows users to ask questions about Ashesi. The diagram below is a prototype of the presentation layer for AshesiBot.



Majors offered at Ashesi?

CS, ENG, BA, MIS

What is MIS?

Management Information Systems

Figure 3. 2: Prototype of the presentation layer

3.2.2 The Application Layer

The second layer in the 3-layer architecture is the application layer, also known as the functional process logic. This layer is responsible for matching the user input to a domain and intent it belongs to and retrieve the domain response needed to meet the user requirement. The different types of modules that makes the application layer are listed below:

A) The Facebook Messenger Event Webhook: This module allows the AshesiBot to receive or push Facebook Messenger events. A detailed information about this module is shown in section 3.5.2

B) The Routing Engine: This module or component is responsible for routing requests from the user to their appropriate functions, allowing a user to get feedback from the bot. See 3.5.3 for more details.

C) **The Domain and Entity Parser**: This module houses the trained model (Rasa NLU), which functions to interpret incoming user input to their appropriate domain or entity. See section 3.5.4 for more information about how the domain and entity module works.

The diagram below (Fig 3.3) shows the application layer:

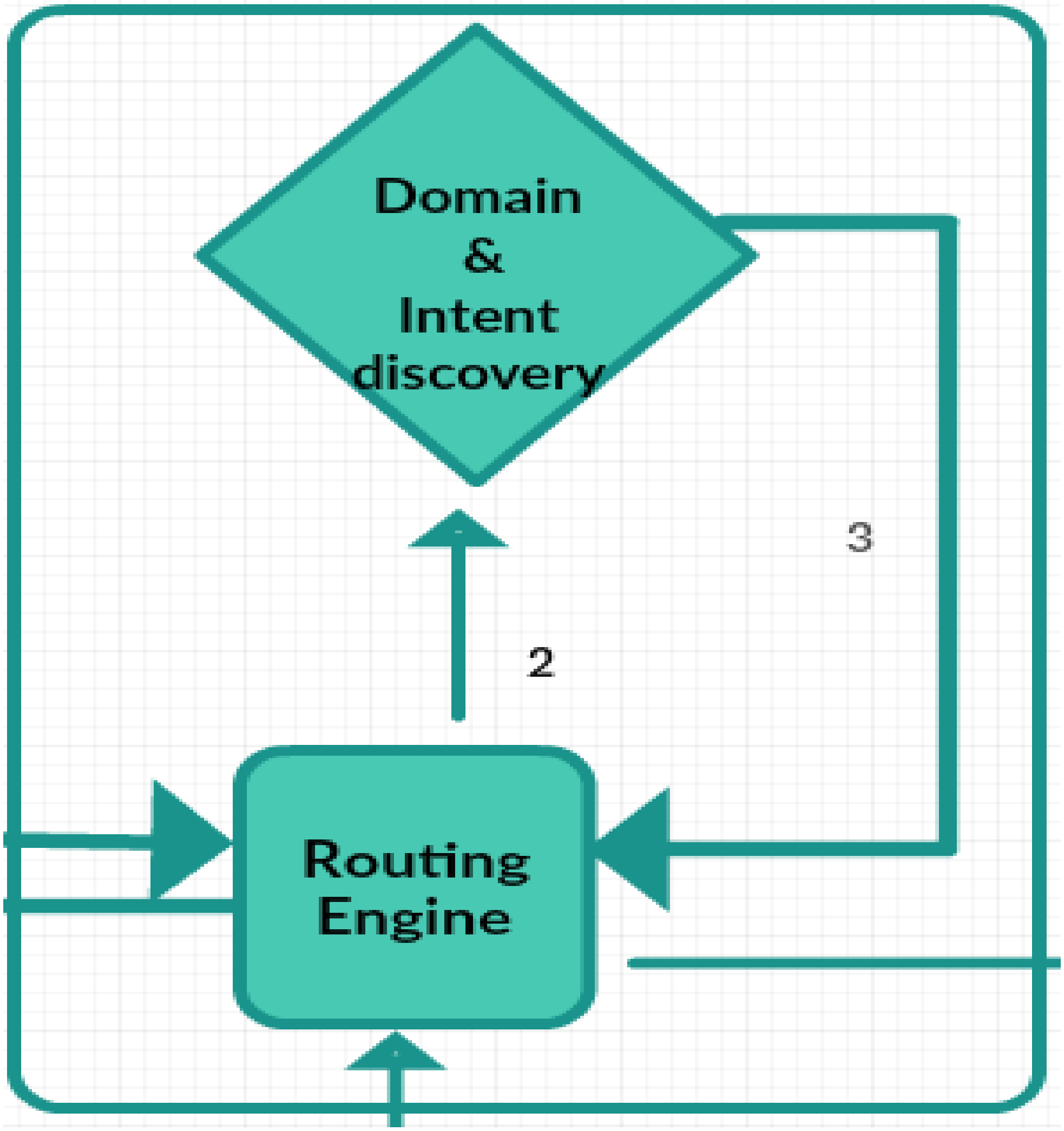
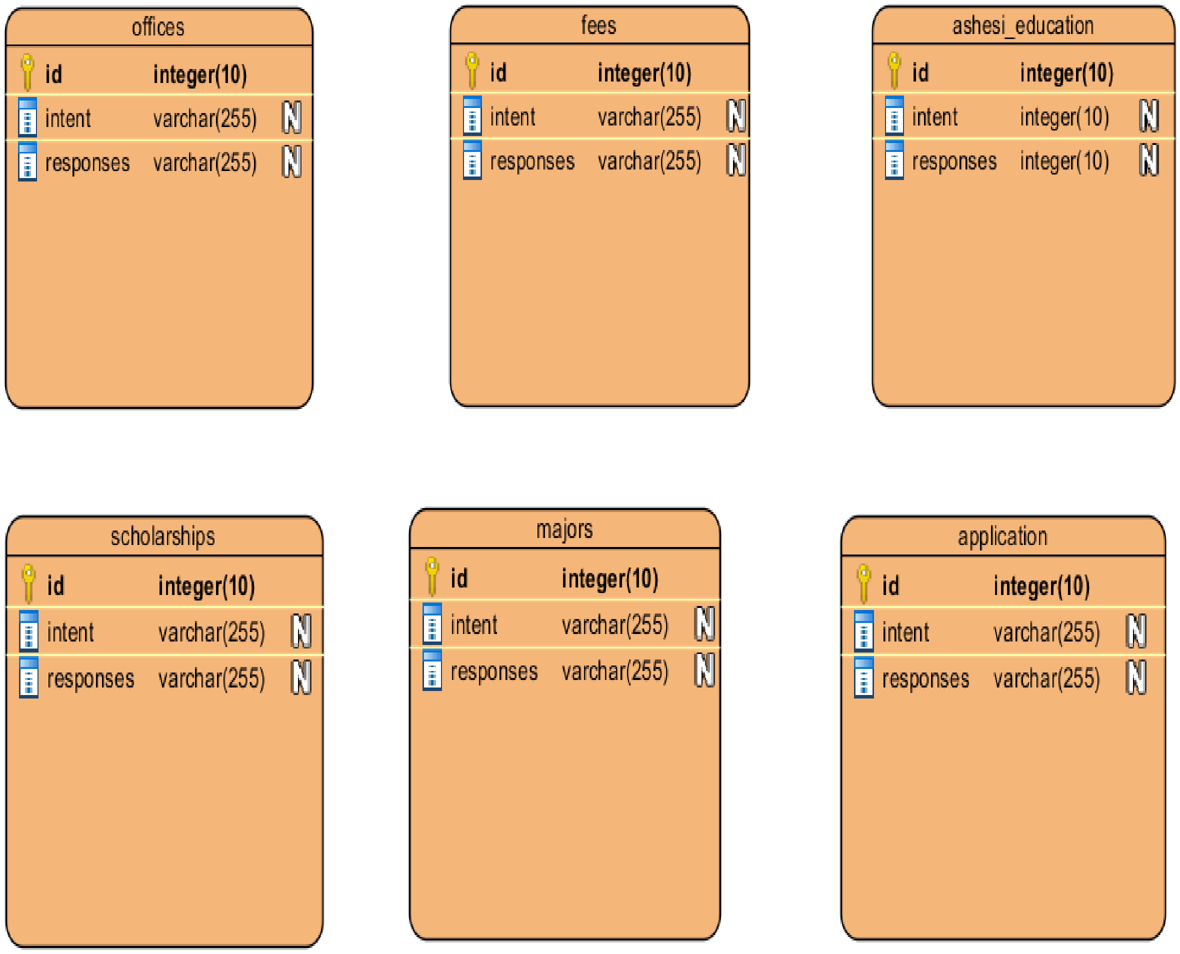


Figure 3. 3: The application layer

3.2.3 Database Layer:

This section describes the layer that stores the data. It is the knowledge base of the AshesiBot. The knowledge or information about Ashesi is organized into relations, also referred to as tables. Each table is a domain which consists of the user's overall intent, and *responses;* the corresponding information or knowledge about the domain and the intent. The database design diagram is below:



offices

id	**integer(10)**	
intent	varchar(255)	N
responses	varchar(255)	N

fees

id	**integer(10)**	
intent	varchar(255)	N
responses	varchar(255)	N

ashesi_education

id	**integer(10)**	
intent	integer(10)	N
responses	integer(10)	N

scholarships

id	**integer(10)**	
intent	varchar(255)	N
responses	varchar(255)	N

majors

id	**integer(10)**	
intent	varchar(255)	N
responses	varchar(255)	N

application

id	**integer(10)**	
intent	varchar(255)	N
responses	varchar(255)	N

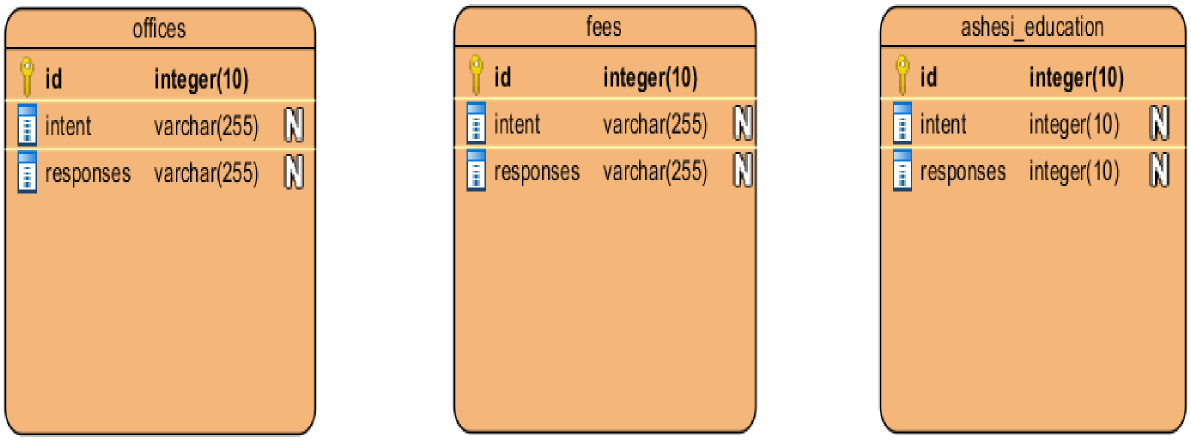


Figure 3. 4: The database layer

3.4 System Modules

This section provides detailed information about the design of each application layer module.

3.4.1 Chatbot Channel or User Interface

To engage in a conversation with the trained agent, the AshesiBot, an interface needs to be provided. The AshesiBot project will, however, make use of the already existing Ashesi Facebook Messenger page as the channel for communication. This is because the page currently has about 37, 000 subscribers, which is a large enough number to target.

3.4.2 Facebook Messaging Event and Webhook

This module is responsible for receiving events from or posting messages to the Ashesi Facebook Messenger page. It contains procedures and callbacks that hook and respond (derived from the name webhook) to all events that are triggered in the Ashesi Messenger page. To successfully configure a webhook with the trained agent, an endpoint needs to be established. In this case, an endpoint at https://ashesibott.herokuapp.com/webhook was configured to receive real-time events triggered in the Messenger page.

Additionally, this module involves understanding the event tree or data structure from Facebook Messenger. This will help in taking appropriate action based on the type of event. The following data (in JSON format) is an example of an event triggered when a user (with sender id: 1003774169746657) types in a text (“get me admission contact”) to the AshesiBot. Also, the recipient id, as shown in the event tree, is the AshesiBot. The snippet below shows the event tree from Facebook Messenger:

```
{
    "object": "page",
    "entry": [{
        "id": 43674671559,
        "time": 1460620433256,
        "messaging": [{
            "sender": {
                "id": 1003774169746657
            },
            "recipient": {
                "id": 1003774169746657
            },
            "timestamp": 1460620433123,
            "message": {
                "mid": "mid.1460620432888:f8e3412003d2d1cd93",
                "seq": 12604,
                "text": "Get me admission contact"
            }
        }]
    }]
}
```

Listing 3.4 Shows an Event Tree from Facebook Messenger

3.4.3 The Routing Engine

This module or component is responsible for routing requests from the user to their appropriate functions, allowing a user to get feedback from the bot. This module is the Python Flask app that interfaces with the Rasa NLU. It can therefore be described as the component that feeds the NLU model with data from the user via the chatbot channel. The routing engine is developed based on the concept of REST (Representational Sate Transfer) architecture, where smaller services are exposed to the user.

3.4.4 The Domain and Intent Parser

This module houses the trained model (Rasa NLU), whose functions are to identify and interpret incoming user input to their appropriate domain and intent. Data gathered from the elicitation stage (in chapter 2) was organized according to themes and subject. These themes were translated to the following domains, which is currently supported on this platform. Any domain not in this list is outside the scope of this project and will receive a custom fallback response such as “I’m not trained on this domain”.

A) Admissions

B) Scholarships

C) Application process

D) General information about Ashesi

E) Fees and finances

F) Courses

G) Lecturer and staff office details

H) Greetings and pleasantries

I) Affirmative and negative responses

Included in this section is a sample training dataset and its format. The data below shows the format for training the applications domain with an entity on online application.

```
{
  "rasa_nlu_data": {
    "common_examples": [
   {

    "text": "do you have online application",

    "intent": "application",

    "entities": [

     {

      "start": 12,

      "end": 30,

      "value": "online application",

      "entity": "online_application"

     }

    ]

   }

    ]

 }

}
```

Listing 3.1 Shows a Training Dataset Instance for the Application Process Domain

CHAPTER 4: IMPLEMENTATION

4.1 Overview

This chapter outlines the technologies used to implement the AshesiBot. These technologies include programming languages, libraries, and tools. It also focuses on the implementation techniques and processes such as modularity. The evidence of implementation and challenges faced in implementing the AshesiBot and measures taken to address those challenges are highlighted in this chapter.

4.2 Languages

4.2.1 Python

Python is a high-level interpreted programming language that is widely used in different projects such as web development, data science and Natural Language Processing projects [6]. Python was deemed fit for this project because it is easy to learn and uses a few lines of code to achieve the desired output. Besides this, Python has a large community base, which provides readily available information to speed up software project developments. In this project, the Python language was used to handle the application layer of the AshesiBot, where all logic and application functions are implemented.

4.2.2 PostgreSQL

PostgreSQL is an object-relational database system for storing data. MySQL was implemented first before migrating to PostgreSQL. This migration was necessary because of its seamless integration with Heroku, where the application layer of this project is hosted. Moreover, PostgreSQL has similar syntax with MySQL, which made the transition from MySQL quite easier.

4.3 Libraries

4.3.1 Python Flask

Flask is a microservice python framework for building web applications [7]. In this project, a mini python flask app, which relies on the REST API architecture was implemented to handle the logic and routing of request at the server-side. It functions to receive requests from users, extracts useful parameters from the URL request and then forwards it to the Rasa NLU model to determine the domain and intent of the user request. Upon receiving the domain and intent of the user request from the Rasa NLU, the python flask makes a query to the database to retrieve information that accurately responds to the user’s intent. The database response is finally sent to the user via Facebook Messenger webhook.

4.3.2 Rasa NLU

Rasa Natural Language Understanding (Rasa NLU) is an open source library for identifying domain and intents from a user’s input; in this case, Facebook messenger input [8]. Rasa NLU was chosen because it allows a developer to select his or her preferred pipeline, either TensorFlow or spaCy or some custom pipeline. In addition to this, unlike other NLU platforms like the Dialogflow, the trained model is stored locally (on the developer’s computer or server), which came in handy to mitigate against possible security breaches.

4.4 Tools

4.4.1 Ngrok

Ngrok is a free HTTP tunneling platform for establishing a secured network connection from a local host to the internet and vice versa. Ngrok was used in the testing phase of the application where an https endpoint was created to interface with webhook events from Facebook Messenger.

4.4.2 Postman

Postman is a tool for designing and testing application programming interfaces (APIs). It allows one to send requests to a RESTful API. In this project, Postman was used for testing the flask app endpoints to ensure requests and responses are in a standard format.

4.4.3 Git

Git is a software for tracking versions of software under development at different stages. In this project, git was used for two purposes, to track the system version and to deploy the project on the Heroku server.

4.4.4 Heroku

Heroku platform is a cloud server hosting platform that supports multiple software applications written in different programming languages. The AshesiBot application is hosted on Heroku server because it is free and supports Python projects.

4.5 Components of the application:

4.5.1 Application Functions Component (Flask app)

This component is responsible for listening and forwarding users request to the Rasa NLU domain and intent extractor unit for identification. After identifying the domain and intent from the user request, this component makes a request to the PostgreSQL database to retrieve responses that correctly matches the user’s intent.

4.5.2 Data Storage Component

This component, which consists of PostgreSQL tables store information about the responses or values for the identified domain and its associated intents.

4.5.3 Channel/Client Side

This channel is the platform where the user and the chatbot hold a conversation. In this project, the Facebook messenger was chosen as the channel to handle the conversation with the user. The user sends their request using the native Facebook messenger app on their phones, and the chatbot sends a response to the user using the same channel. The reason behind this is that Ashesi Facebook Page has more than thirty-seven thousand (37,000) followers, a good number to gain motivation to build a chatbot.

4.6 Implementation Techniques and Process

Modularity is one of the techniques that were implemented in this project. The entire software project was broken down into mini-projects; the Rasa NLU, the flask app, and the training datasets. Implementing modularity simplified debugging and maintenance of the project.

4.6.1 Implementation Challenges: -

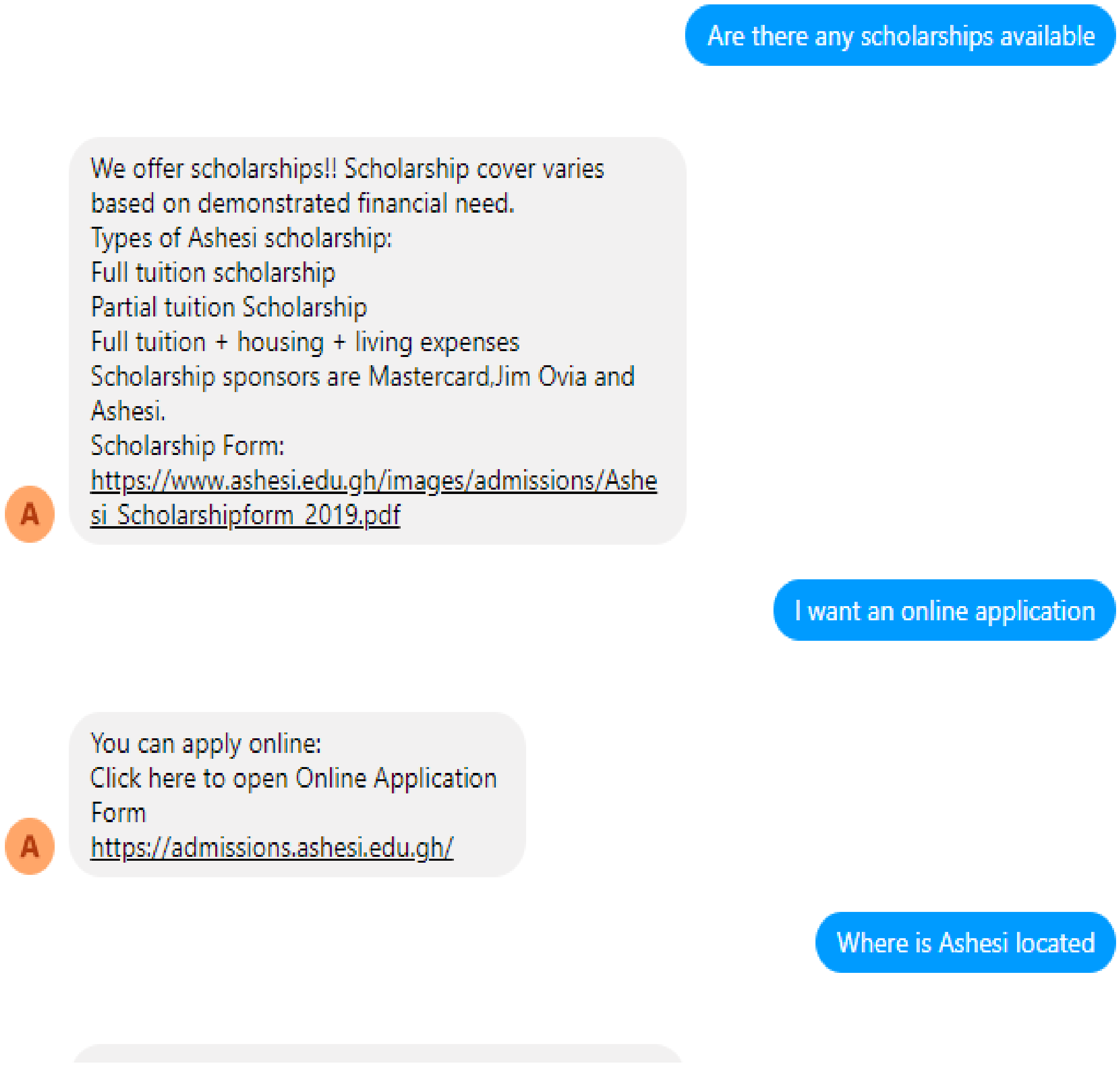
Finding the data to train was one of the biggest implementation challenges. The Ashesi administration could not provide emails related to questions frequently asked by the prospective students. However, the admissions team later issued out some questions that they gathered from these students, but the data they issued out was inadequate. To overcome this, training data was obtained from the Ashesi website that has all the information about Ashesi

Furthermore, there was a challenge handling the conversation part of the chatbot. Rasa core, a trained model for managing conversation was not useful solution due to its rigidness; it requires one to forelay out all the available domains in order, and all the possible ways a user can interact with the chatbot. A better option was to use Rasa NLU, which functions to train the data and identify intents and entities. This problem was still not resolved because there is the need to interface with the NLU model. To overcome this challenge, tools such as Ngrok (working as a server for a local project) and Facebook messenger were used as a channel to allow users to interact with the chatbot.

However, the Ngrok server was not reliable; its connectivity is highly unstable. It works for a few hours then it is interrupted. To address the server instability challenge, Heroku platform offered the solution to deploy the project on their server. This Heroku platform solution made it possible to connect to Facebook Messenger APIs to allow smooth flow of conversations.

4.6.2 Evidence of Implementation

Below are the images of the AshesiBot interacting with a Facebook user:



Are there any scholarships available

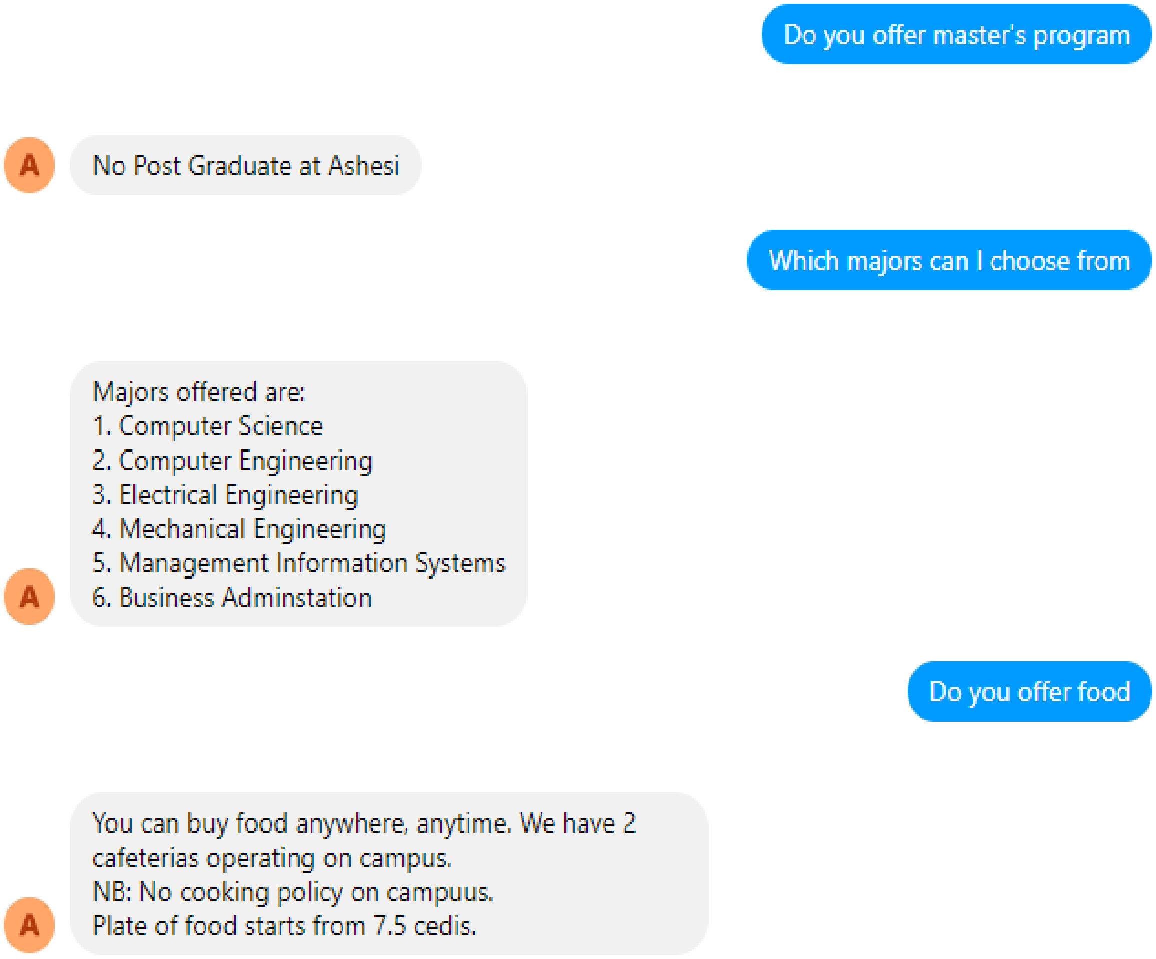
We offer scholarships!! Scholarship cover varies based on demonstrated financial need.
Types of Ashesi scholarship:
Full tuition scholarship
Partial tuition Scholarship
Full tuition + housing + living expenses
Scholarship sponsors are Mastercard,Jim Ovia and Ashesi.
Scholarship Form:
https://www.ashesi.edu.gh/images/admissions/Ashesi_Scholarshipform_2019.pdf

I want an online application

You can apply online:
Click here to open Online Application Form
https://admissions.ashesi.edu.gh/

Where is Ashesi located

Figure 4. 1: Screenshot of AshesiBot Conversation



Do you offer master's program

No Post Graduate at Ashesi

Which majors can I choose from

Majors offered are:
1. Computer Science
2. Computer Engineering
3. Electrical Engineering
4. Mechanical Engineering
5. Management Information Systems
6. Business Adminstation

Do you offer food

You can buy food anywhere, anytime. We have 2 cafeterias operating on campus.
NB: No cooking policy on campuus.
Plate of food starts from 7.5 cedis.

Figure 4. 2: Screenshot of AshesiBot Conversation

Chapter 5: Testing and Outcome

5.1 Chapter Overview

This chapter explores the procedures and techniques used for testing AshesiBot. This enables assessing and authenticating the software against the functional and nonfunctional requirements as discussed in chapter 2. Moreover, it is useful for finding errors and bugs in the software project and getting feedback from users. These set of activities help to enhance the final product's quality. This chapter includes statistics or outcome from the test cases carried out. Eleven (11) randomly selected people (based on their availability and convenience) interacted with the chatbot and their feedback was solicited for via Google Form. This form captures their experience after chatting with the bot. Below is the information the project sought to collect:

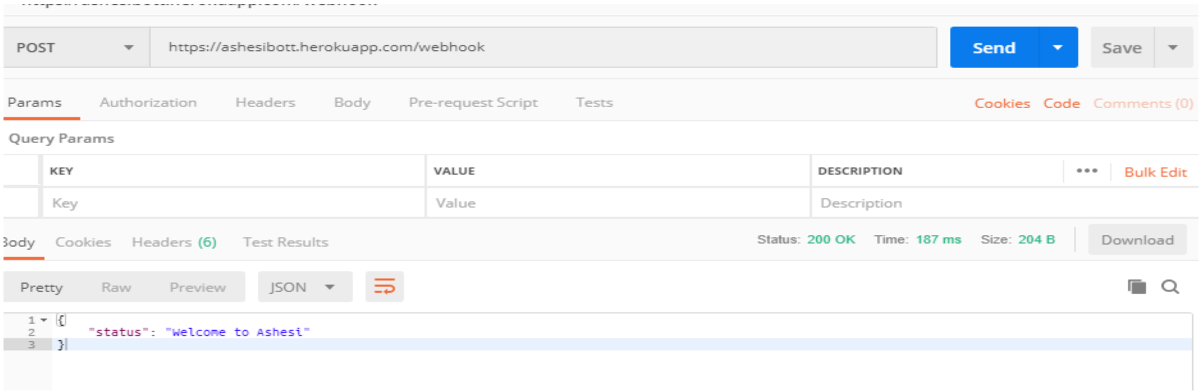
- User rating on the naturalness of their interaction with the chat bot.
- User rating on the accuracy of the chatbot.
- User rating on the latency of the chatbot.
- User's experience during their interaction with the chatbot
- If this was the user's first time to interact with any kind of automated chatbot.

5.2 Development Testing

Development testing is a testing technique where testing is carried out during the development of the software to expose defects such as incorrect computation, system crashes and bugs in the software by the development team. Moreover, it is carried out to identify the system behaviors that are not harmonized with the software requirements and non-requirements. It is also done to ensure that the users are included in the process of software development by gaining their feedback at every milestone. Each function was tested after development to check if it is serving its intended purpose.

Test Case 1: The Flask Endpoint

To test the Application Programming Interfaces (APIs) defined in the routing engine module, POSTMAN tool was used for testing the flask app endpoints. This is to ensure requests and responses are in a standard format. The standard format expected here is JSON. The result displayed in the snippet below shows that the APIs are working as expected. Below is the snippet:



POST https://ashesibott.herokuapp.com/webhook

Status: 200 OK Time: 187 ms Size: 204 B

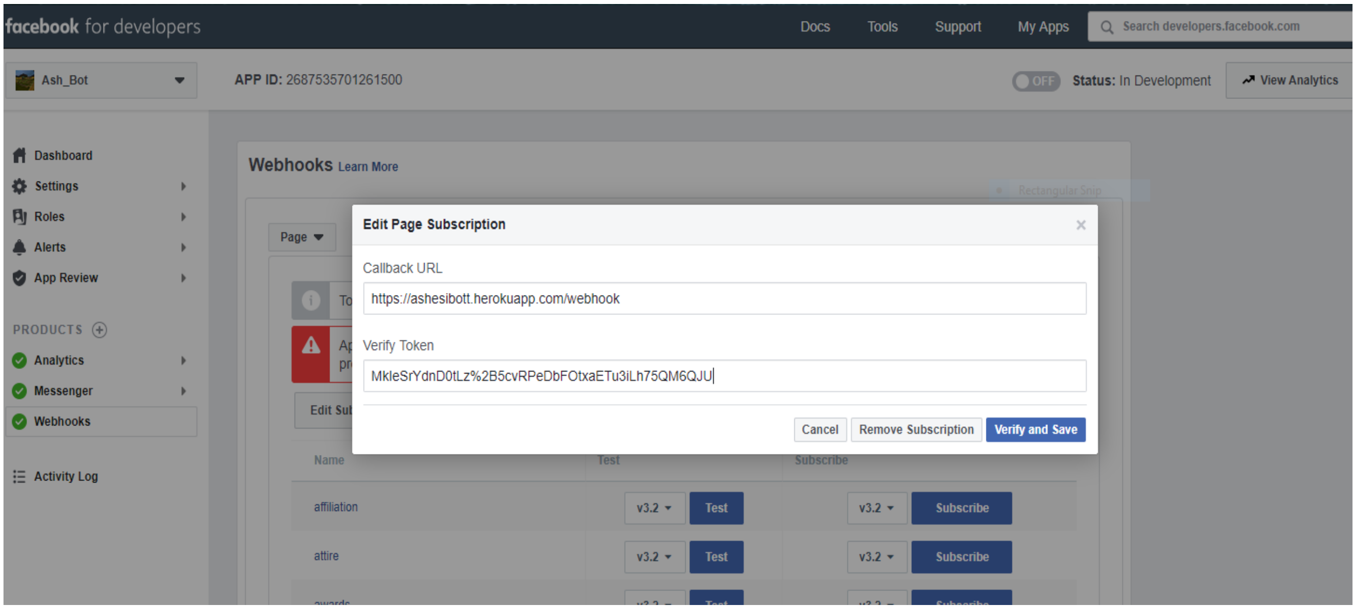
JSON

```
{
    "status": "Welcome to Ashesi"
}
```

Figure 5. 1: Test result for the flask app

Test Case 2: Integration Between Facebook Webhook and Heroku Endpoint

Additionally, the Facebook Messenger Webhook was tested to ensure it can communicate well with the Flask endpoint hosted at Heroku. The figure below (Figure 5.2) shows a user establishing a connection from Facebook Messenger Webhook with Heroku.



APP ID: 2687535701261500

Status: In Development

Callback URL

https://ashesibott.herokuapp.com/webhook

Verify Token

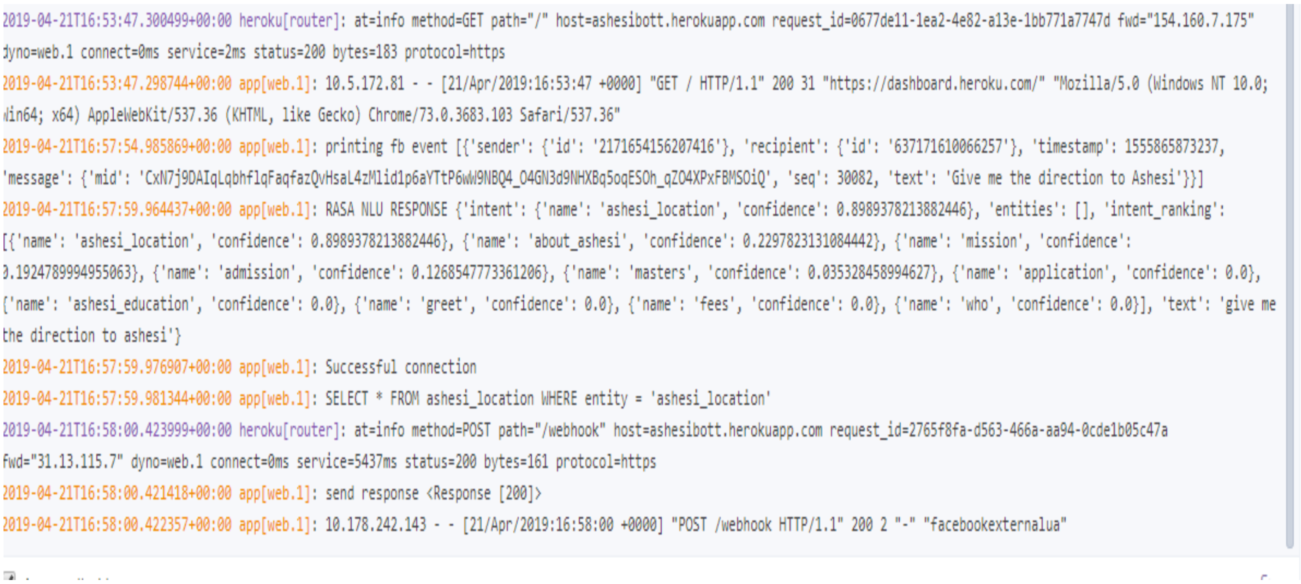
MkleSrYdnD0tLz%2B5cvRPeDbFOtxaETu3iLh75QM6QJU

Name	Test	Subscribe
affiliation	v3.2	v3.2
attire	v3.2	v3.2

Figure 5. 2: Facebook messenger webhook in integration with heroku test case

Test Result

Due to the successful integration between the Flask endpoint and the Facebook Messenger webhook, any interaction between a user and the Ashesi page is forwarded to the Heroku server for processing. Find below a snippet of the Heroku logs.



```
2019-04-21T16:53:47.300499+00:00 heroku[router]: at=info method=GET path="/" host=ashesibott.herokuapp.com request_id=0677de11-1ea2-4e82-a13e-1bb771a7747d fwd="154.160.7.175"
dyno=web.1 connect=0ms service=2ms status=200 bytes=183 protocol=https
2019-04-21T16:53:47.298744+00:00 app[web.1]: 10.5.172.81 - - [21/Apr/2019:16:53:47 +0000] "GET / HTTP/1.1" 200 31 "https://dashboard.heroku.com/" "Mozilla/5.0 (Windows NT 10.0;
Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/73.0.3683.103 Safari/537.36"
2019-04-21T16:57:54.985869+00:00 app[web.1]: printing fb event [{'sender': {'id': '2171654156207416'}, 'recipient': {'id': '637171610066257'}, 'timestamp': 1555865873237,
'message': {'mid': 'CxN7j9DAIqLqbhflqFaqfazQvHsaL4zMlid1p6aYTtP6wW9NBQ4_O4GN3d9NHXBq5oqESOh_qZO4XPxFBMSOiQ', 'seq': 30082, 'text': 'Give me the direction to Ashesi'}}]
2019-04-21T16:57:59.964437+00:00 app[web.1]: RASA NLU RESPONSE {'intent': {'name': 'ashesi_location', 'confidence': 0.8989378213882446}, 'entities': [], 'intent_ranking':
[{'name': 'ashesi_location', 'confidence': 0.8989378213882446}, {'name': 'about_ashesi', 'confidence': 0.2297823131084442}, {'name': 'mission', 'confidence':
0.1924789994955063}, {'name': 'admission', 'confidence': 0.1268547773361206}, {'name': 'masters', 'confidence': 0.035328458994627}, {'name': 'application', 'confidence': 0.0},
{'name': 'ashesi_education', 'confidence': 0.0}, {'name': 'greet', 'confidence': 0.0}, {'name': 'fees', 'confidence': 0.0}, {'name': 'who', 'confidence': 0.0}], 'text': 'give me
the direction to ashesi'}
2019-04-21T16:57:59.976907+00:00 app[web.1]: Successful connection
2019-04-21T16:57:59.981344+00:00 app[web.1]: SELECT * FROM ashesi_location WHERE entity = 'ashesi_location'
2019-04-21T16:58:00.423999+00:00 heroku[router]: at=info method=POST path="/webhook" host=ashesibott.herokuapp.com request_id=2765f8fa-d563-466a-aa94-0cde1b05c47a
fwd="31.13.115.7" dyno=web.1 connect=0ms service=5437ms status=200 bytes=161 protocol=https
2019-04-21T16:58:00.421418+00:00 app[web.1]: send response <Response [200]>
2019-04-21T16:58:00.422357+00:00 app[web.1]: 10.178.242.143 - - [21/Apr/2019:16:58:00 +0000] "POST /webhook HTTP/1.1" 200 2 "-" "facebookexternalua"
```

```
2019-04-21T16:57:54.985869+00:00 app[web.1]: printing fb event [{'sender': {'id': '2171654156207416'}, 'recipient': {'id': '637171610066257'}, 'timestamp': 1555865873237, 'message': {'mid': 'CxN7j9DAIqLqbhflqFaqfazQvHsaL4zMlid1p6aYTtP6wW9NBQ4_O4GN3d9NHXBq5oqESOh_qZO4XPxFBMSOiQ', 'seq': 30082, 'text': 'Give me the direction to Ashesi'}}]

2019-04-21T16:57:59.964437+00:00 app[web.1]: RASA NLU RESPONSE {'intent': {'name': 'ashesi_location', 'confidence': 0.8989378213882446}, 'entities': [], 'intent_ranking': [{'name': 'ashesi_location', 'confidence': 0.8989378213882446}, {'name': 'about_ashesi', 'confidence': 0.2297823131084442}, {'name': 'mission', 'confidence': 0.1924789994955063}], 'text': 'give me the direction to ashesi'}

2019-04-21T16:57:59.976907+00:00 app[web.1]: Successful connection

2019-04-21T16:57:59.981344+00:00 app[web.1]: SELECT * FROM ashesi_location WHERE entity = 'ashesi_location'
```

```
2019-04-21T16:58:00.423999+00:00 heroku[router]: at=info method=POST path="/webhook"
host=ashesibott.herokuapp.com request_id=2765f8fa-d563-466a-aa94-0cde1b05c47a
fwd="31.13.115.7" dyno=web.1 connect=0ms service=5437ms status=200 bytes=161 protocol=https

2019-04-21T16:58:00.421418+00:00 app[web.1]: send response <Response [200]>
```

Listing 5.0

5.4 System Testing

System testing is a type of testing where components that make up a complete system are integrated and tested. This ensures that every component interacting with another component is tested for compatibility and correctness. Eleven randomly selected people were asked to interact with the chatbot after which they can provide feedback about the performance of the chatbot. Below is the user rating of the AshesiBot:

User Test Case 1: Naturalness of the Conversation

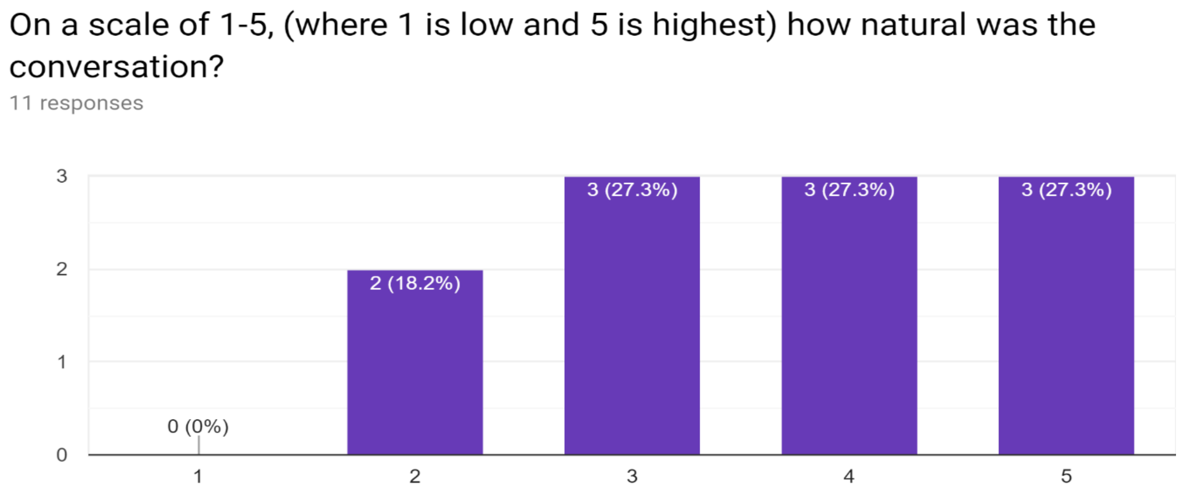
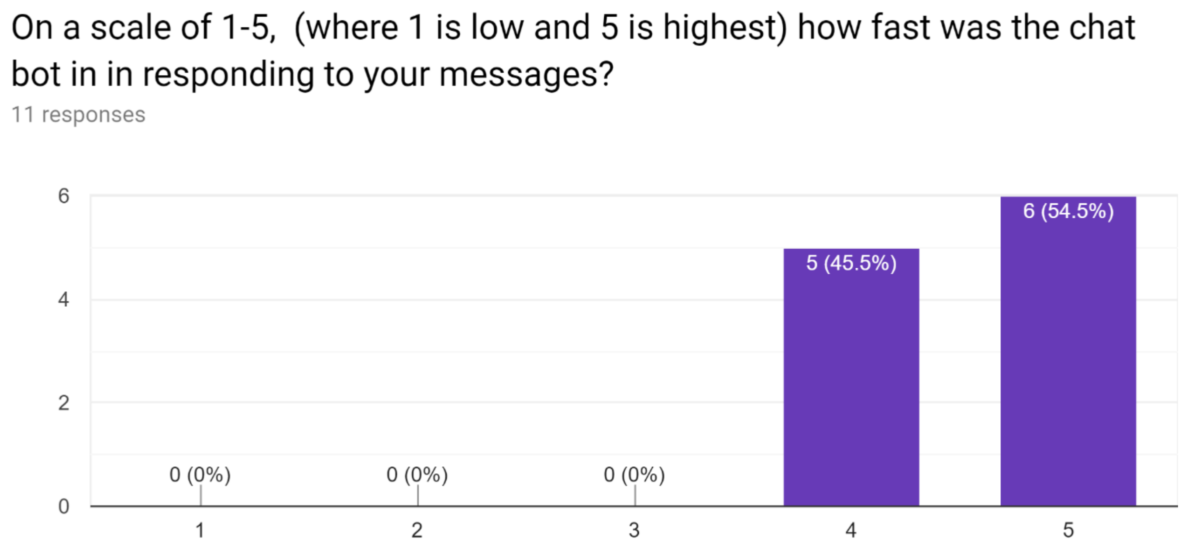


Figure 5. 3: Results of the Naturalness of the Conversation data

Test Case 1 Result:

The chat above indicate that more than 80 percent of the participants responded that the AshesiBot was able to engage in a natural conversation. However, steps to be taken to improve the naturalness of the chatbot is outlined in the recommendation section below.

User Test Case 2: The Speed of AshesiBot



On a scale of 1-5, (where 1 is low and 5 is highest) how fast was the chat bot in in responding to your messages?

11 responses

Figure 5. 4: Results of the speed of AshesiBot

Test Case 2 Result

Based on the survey carried out, all the participants accepted that the AshesiBot latency is high. Until this proposed solution, the latency or the time it takes for users to get their questions responded to was very low (an average of a week). Thankfully, the use of AshesiBot increases latency (response rate) to about few seconds.

User Test Case 3: Accuracy of AshesiBot Responses

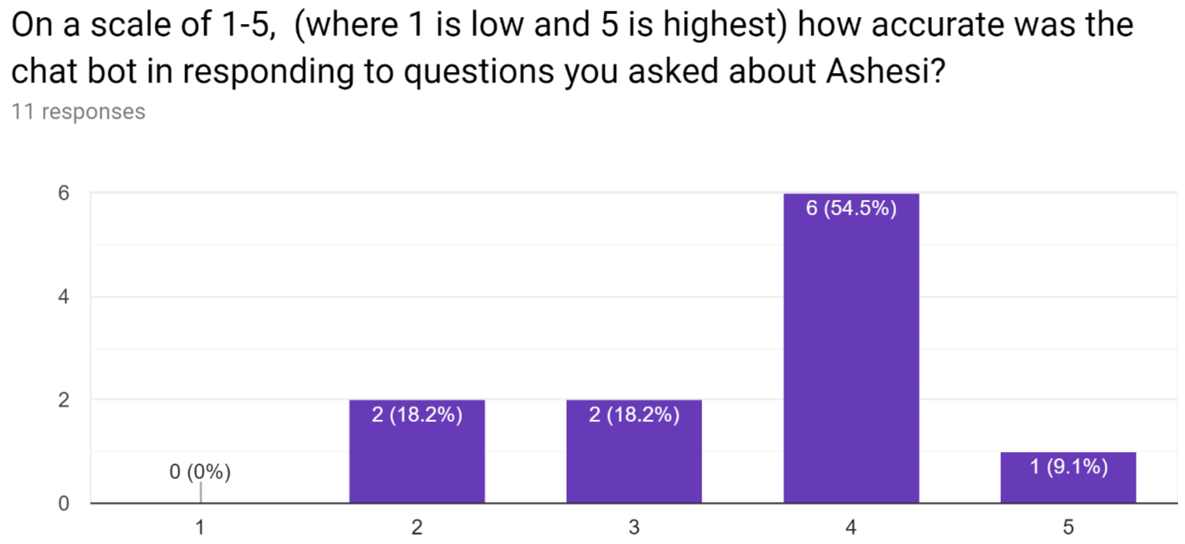


Figure 5. 5: Results of accuracy of AshesiBot responses

Test Case 3 Result

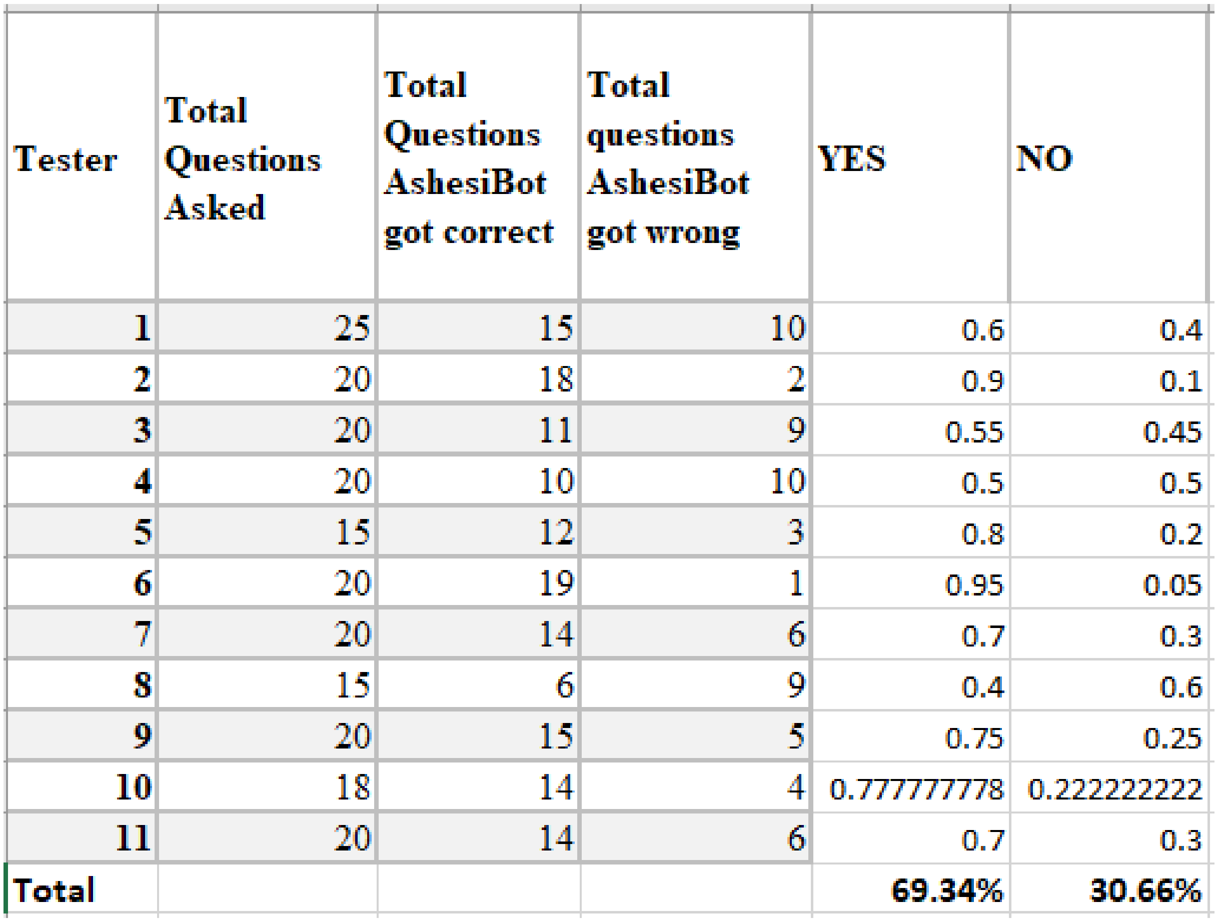
From the diagram above, about 55% of the test users confirmed that the AshesiBot has a very good accurate response while only 9.1 percent responded it has a high response rate. Clearly, there is the need for improvement of the bot’s responses and the recommendations to achieve 100 percent excellent rating are outlined in the recommendation section below.

User Test Case 4: Familiarity with Chatbots

Approximately sixty-four percent of the randomly selected people who participated in the survey have interacted with a chatbot before. Studies shows that, there is a huge difference between people who have interacted with chatbots before and people who haven’t interacted with them [9]. People who have knowledge about chatbots or have interacted with a chatbot before tends to structure their questions differently in a way that accommodates the limitations of chatbots in general. Seventy-two percent, (majority) of the people enjoyed interacting with the chatbot. This could be interpreted as they already knew the general limitations of chatbots.

The following evaluation was carried out to understand the reasons behind the wrong answers given by the AshesiBot. A critical analysis of looking at each response given by the chatbot in response to the user’s request helped to collect this information below. Below is the table that shows the statistics of the number of questions each user asked, number of questions the bot answered correctly and the ones it failed.

Table of chatbot results:



Tester	Total Questions Asked	Total Questions AshesiBot got correct	Total questions AshesiBot got wrong	YES	NO
1	25	15	10	0.6	0.4
2	20	18	2	0.9	0.1
3	20	11	9	0.55	0.45
4	20	10	10	0.5	0.5
5	15	12	3	0.8	0.2
6	20	19	1	0.95	0.05
7	20	14	6	0.7	0.3
8	15	6	9	0.4	0.6
9	20	15	5	0.75	0.25
10	18	14	4	0.777777778	0.222222222
11	20	14	6	0.7	0.3
Total				**69.34%**	**30.66%**

From the table, the AshesiBot could not answer all the questions correctly. On average, the trained agent accurately answered 70 percent of the user's questions. On the flip side, the chatbot was not able to understand the meaning of the key words given by the user, accounting for the 30 percent inaccurate responses. Also, the syntax of the questions coupled with the morphological structure of the user's sentences, and questions which were entirely out of scope accounted for this 40% loss.

Chapter 6: Recommendation and Conclusion

Chapter Overview

This chapter marks the end of the project. It highlights the conclusion and recommendations for future work.

Conclusion

To this end, a conversational agent –AshesiBot –was implemented to improve the information retrieval system at Ashesi. The main objective of this proposed solution is to allow for quick (measured in time) and efficient (measured in terms of complexity) retrieval of information about Ashesi University. It was discovered at the testing phase of the project that the AshesiBot had tremendously reduced the time a user waits for a response on Ashesi Facebook messenger from an average of a week to a few seconds.

Recommendations

There is room for improvement of the AshesiBot. While a higher proportion of the requirements were implemented, there is a need to review and make improvements. Anyone interested in taking this project up is hereby recommended to undertake the following measures. Firstly, this project made use of TensorFlow to train the dataset. However, TensorFlow requires a large number of training data (about 2000 and above) to be able to function properly. Therefore, interested researchers are encouraged to add more data on the topics covered with different syntax and semantics to improve the accuracy of the chatbot's responses.

Additionally, the chatbot, currently does not cover all the domain and information one might want to know about Ashesi. Therefore, it is recommended to interested researchers to add more data that adds or broadens the scope of this project so user can have conversation with the chatbot across several domains. Moreover, the chatbot as at the time I write this report is supported only on Facebook Messenger platform. Researchers can increase the coverage of this project by investigating the feasibility of other social media channels like Twitter, Skype or even via Email.

Finally, one of the goals of the non-functional requirements of the chatbot is to engage in a natural conversation as any normal human conversation will do. The use of conversational techniques like context management and tracking function can be beneficial in tracking previously asked questions. These recommendations, if implemented will not only make the trained agent, the chatbot, smart but also, make it more robust to handle conversations organically.

References

[1] Godson M. D'silva, Sanket Thakare, Sharddha More, and Jeril Kuriakose. 2017. Real world smart chatbot for customer care using a software as a service (SaaS) architecture. Retrieved March 31, 2019 from https://doi.org/10.1109/I-SMAC.2017.8058261

[2] Bhavika R. Ranoliya, Nidhi Raghuwanshi, and Sanjay Singh. 2017. Chatbot for university related FAQs. Retrieved March 31, 2019 from https://doi.org/10.1109/ICACCI.2017.8126057

[3] Anon. 2018. The University of Sydney: Exploring AI and implementing Chatbots to achieve operational efficiencies. 2018. Retrieved March 31, 2019 from

https://educationblog.microsoft.com/2018/03/the-university-of-sydney-exploring-ai-and-implementing-chatbots-to-achieve-operational-efficiencies/

[4] Ashay Argal, Siddharth Gupta, Ajay Modi, Pratik Pandey, Simon Shim, and Chang Choo. Intelligent Travel Chatbot for Predictive Recommendation in Echo Platform. 2018. Retrieved March 31, 2019 from https://ieeexplore.ieee.org/document/8301732

[5] Dongkeon Lee, Kyo-Joong Oh, and Ho-Jin Choi. 2017. The chatbot feels you - a counseling service using emotional response generation. Retrieved March 31, 2019 from https://ieeexplore.ieee.org/document/7881752/

[6] Andreas C. Müller and Sarah Guido. 2017. *Introduction to machine learning with Python: a guide for data scientists*, Beijing: OReilly.

[7] Armin Ronancher. Flask. Retrieved April 3, 2019 from http://flask.pocoo.org/

[8] Anon. Rasa NLU: Language Understanding for chatbots and AI assistants. Retrieved April 3, 2019 from https://rasa.com/docs/nlu/

[9] Mohit Jain, Pratyush Kumar, Ramachandra Kota, and Shwetak N. Patel. 2018. Evaluating and Informing the Design of Chatbots. (June 2018). Retrieved April 8, 2019 from https://dl.acm.org/citation.cfm?id=3196735