Adventist University of Central Africa

BANKING BOT

CASE STUDY: EQUITY BANK RWANDA

A research project

presented in partial fulfillment of the

requirements for the degree

BACHELOR OF INFORMATION TECHNOLOGY

Major in

SOFTWARE ENGINEERING

By

IHOGOZA Valentine

May, 2022

ABSTRACT

Research Project for the Bachelor Degree in Information Technology

Emphasis in Software engineering

Adventist University of Central Africa

Title: BANKING BOT (CASE STUDY: EQUITY BANK)
Name of the researcher: IHOGOZA Valentine
Name of the faculty advisor: Dr. SEBAGENZI Jason.
Date completed: May, 2022

The primary objective of this study was to develop a banking bot which will be there 24/7 replying to customers frequently asked questions for the bank to improve their customer satisfaction and for the customers to save their time.

In order to reach the project goals, the analysis of the existing system was done and problems that customers face when they need help were identified. From there, adequate solutions to address the raised issue were offered.

I have used different materials during analysis, design and development of the system.
python and Rasa for the development of the system, Ngrok for deploying a bot temporally on Facebook and Telegram, and for the report, I have used visual studio code.

Observation and interviews were ones of the methods used to gather data that lead to the conceptualization of the problem and gathering of information used in the building of the system.

DECLARATION

I hereby affirm that this project report is based on my original work except for citations and quotations which have been duly acknowledged. I also declare that it has not been previously and concurrently submitted anywhere.

Signature   : _____

Name   : IHOGOZA Valentine

Date   : May, 2022

APPROVAL

I, Dr. SEBAGENZI Jason; hereby certify that this project report has been done under my supervision and submitted with my approval.

Signature…………………………………..
Date………………………

DEDICATION

To the Almighty God,

To my lovely family especially my parents

For their support during all the time of my studies.

To all my friends and relatives, without also

Forgetting my supervisor for his guidance.

This work is dedicated.

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

LIST OF EQUATIONS

LIST OF ABBREVIATIONS

AUCA   Adventist University of Central Africa

NLU   Natural Language Processing

NLP   Natural Language Understanding

USSD   Unstructured Supplementary Service Data

CRM   Customer Relationship Management

FAQ   Frequently Asked Questions

ML   Machine Learning

AI   Artificial Intelligence

CLS   Classification

ACKNOWLEDGMENT

First and foremost, my infinite gratitude goes to the Almighty GOD who has provided life and all needs instead of what I wanted. This is only one in many other miracles He has done to me.

I genuinely offer my recognition to the academic staff of the department of Information Technology and the whole Administration at large at the Adventist University of Central Africa for the work done during my stay at the University, the knowledge that was acquired helped me to conduct this work, and I take this opportunity to thank the whole AUCA staff. My sincere gratitude goes to my supervisor, Dr. Jason SEBAGENZI, for his wise technical advices, suggestions and corrections that made this research project fruitful.

Special thanks goes also to my dear beloved brothers and parents who have always encouraged and supported me during my studies to the completion of this work. No one is truly self-made, hence I pass on a cordial thanks to my colleagues at AUCA, and also from the deepest core part of my heart I thank a Carnegie Mellon University Africa alumni; Isaac Coffie for his guidance.

I am also indebted to EQUITY BANK for allowing me the privilege to do my research in this respectable institution, which provided helpful data.
Finally, I express my gratitude to each and every one who directly and indirectly contributed to make my research project successful today.

God bless you all!

CHAPTER 1

GENERAL INTRODUCTION

Introduction

Banking has become the part and parcel of everyone's life. Almost everyone uses the banking sector to perform their tasks. As the number of people who use banks increases as the number of queries regarding banks' information increases. And the customer care person is in charge of providing the answer to those queries; be it solving them directly or directing them to another person in charge. Chatbots are becoming trending today. They are computer programs that interact with users using natural languages.

In this project, I am going to develop an AI-based Chatbot that can be added to messaging channels and APIs like; a bank's web App, Telegram, Facebook Messenger, Twilio, Slack, Google hangouts, etc. Artificial intelligence(AI or machine learning) chatbots, which uses natural language processing(NLP) technologies to recognize the intent behind the question and reply to the customer's query without any human assistance in order to improve customer satisfaction.

Background of the Study

Nowadays the use of information technology in different domains is increasing and taking an important role in the development of this world. We are in a digital world where almost everything is being done online.

Technology is growing by leaps and bounds, especially in our country. And when it comes to financing, it plays a huge role in enhancing activities. In Rwanda, particularly the traditional way of saving money in "ihembe" or "horn" is becoming history. Use of banks is taking over.

As we all know the population is growing everyday; in 2020 the population of Rwanda was 12.9 million according to *Rwanda Population (LIVE),(2020)*. Of which 36% of adults which is slightly **2.6 million individuals** according to *Access to Finance Rwanda,(2020)*; are using bank services. Rwanda's population is expected to grow by more than half to 17.6 million by

2035, and more than double to 22.1 million by 2050, according to *N.I.S.R. (2022, March 5)*. Can conceive the number of people who will be using bank services in 2050. Thanks to technology, now basic transactions of the bank can be done online; fund transfer, viewing the balance, mini statement, and so on. which is a good thing at least the queue at the retailer table has been reduced. But the big problem that is remaining is the queue at the customer service. Imagine increasing the number of customer service assistants as the number of population and of course banks' customers increases.

The use of powerful algorithms like natural language processing(NLP) and neural networks has shown great potential for enhancing the efficiency, speed and accuracy of tasks performed by customer service assistants. The primary objective of this research is to study the work regarding the field of bank-related customer services and evaluate the best approach which can be used by the system to facilitate their work. Aiming to the objective, this study's puporse is to develop an AI-based chatbot that will be trained and be confident enough to reply to almost all Equity Bank Rwanda FAQs.

Statement of the Problem

The literature assessment revealed that modern financial services are continually seeking to expand their technologies, both to improve customer service and to increase service delivery. This is done to acquire a financial advantage over other banks and to expand its customer base. However, due to a lack of knowledge about how banks operate, we frequently receive many queries.

A domain-specific chatbot will be implemented to assist customer service assistant to improve their service for the sake of overcoming user satisfaction issues. The chatbot will yield coherent communication amid the user and their bank in order to get assistance when needed, such as; answering any queries. The chatbot will allow users to feel confident and comfortable when asking any question. A user doesn't have to be fluent in English due to the simplicity of the natural language used in messages.

Choice and Motivation in the Study

As a Student, it is an opportunity to put in practice the knowledge gained throughout my academic training at Adventist University of Central Africa by solving problems and improving services using technology.

For my University, this study will increase the literature of the institution which will be used for the students as reference for their projects; especially it will serve as a basis to the ones who will deal with related project topics. For academic purpose it will be used for evaluating if the knowledge provided was well acquired. For researchers visiting from outside the study will provide material and reference for their research projects.

For Banking Companies, the new developed system will help the company to improve customer satisfaction.

Objectives of the Study

This project is made up of two objectives, general and specific objectives as mentioned in the following subsections:

General objectives

The main objective of Banking Bot is for Banks to improve customer satisfaction. Where a customer can know almost every information regarding the bank wherever he or she is.

Specific objectives

This study will be focusing specifically on the following objectives:

- To improve the service given to the customers.
- Help customers to minimize and to use time effectively.
- Customers will be able to confidently post their queries.
- To help customers get answers to their queries.
- To reduce the customer service assistant's work.
- To implement a computerized application used to manage the conversation.
- To develop a simple system for the society that could improve the customer satisfaction.

Scope of the Study

This project is mainly focused on the bank and its customers since they are the ones who will be benefiting from the project. Customers will be able to be assisted virtually for some bank related frequently asked questions(FAQ) and how the Banks can use the technology to provide an efficient way of serving Rwandan citizens. The system will be accessed using computers, Smart mobile phones, and the internet; and the geographical scope of the study is limited to the country of Rwanda.

Methodology and Techniques Used in the Study

Research methodology describes how to conduct a research. Many facets are involved in conducting a research. Documents, the workplace, web searches, field notes, questionnaires, and social interaction or interviews are all possible sources of information. To collect data on the existing internal way of assisting customers at a certain bank, the following techniques were used:

Documentation

Documentation involves systematic data collection from existing record, such as books, and Websites and so on. This technique was used in this study while consulting documents online, past memories with related topics on different Bank Companies, and Equity bank's website. The analysis of existing system from information gathered from a review of the documents helped to identify problems and help to assess improvements needed to correct the current system of internal emergency communication.

Observation

Observation refers to the process of recording people's behavior, object and occurrences without questioning or communicating with them. This technique helps me to know really what happens to the customers when they need assistance. With this, a visit to the Equity bank Kisimenti branch Kigali has been done and there are some problems observed.

Interview

Interview is defined as a systematic conversation between an investigator (interviewer) and an informant (interviewee), initiated for obtaining information relevant to a specific study.

To understand deeply the problem domain, interview plan as a data collection technique also helped to achieve the meaning of this project.

The conversation has been conducted and someone named **Mr. MUNEZERO Jean de Dieu** in IT department at Equity Bank Rwanda has been interviewed. He explained to me all the processes, how they technically manage customer's queries.

Some Research questions and their corresponding answers:

1. In which department are you working in this company?

→ IT

2. How long have you worked with this company?

→ 9 years

3. How do you manage customer queries technically?

→ Through Customer Relationship Management system (CRM).

→ Sentiment Analysis

4. How long does it take for a customer who used CRM system to get a reply?

→ Approximately 48 to 72 hours

5. Do you think something can be done to enhance the current System

→ Absolutely

Expected Results

To solve different problems identified in the existing system, the following results are expected:

- The system will be a chat box that will be added to Messaging channels; Telegram and Facebook Messenger to assist users with their queries.

- The conversation will be done online. And it will be a natural language conversation. It would be more as you could interact with a customer service assistant.

- The system will provide accurate information about the bank.

- The system will allow the customers to post any query.

- Properly handle unexpected requests and alert the user if it is unable to provide assistance.

- The system will be able to provide answers in less than 10 sec.

- The system will be able to keep the conversion even when the question asked before was unknown or unclear.

- The system will work 24/7 since there would be no human intervention in answering questions.

- The system will improve the bank's customer services as well as save the customers' time.

Organization of Report

Our study contains five chapters:

First chapter provided a general Introduction with basic information on the research project and the introduction of what we will do.

The **second chapter** will be coming to highlight in details how the existing system of how Rwandan Banking system handle the customers queries and recount the concepts and the domain terms, portray the environment of the system, the techniques that will be used by the system, proposed solution for the new system and in this chapter is where keys words that will be used in new system are defined.

Chapter three entitled Analysis and Design of the new System is ordinarily the logical conception of the new system. It will portray the conceptual process of the solutions proposed to solve the problems of the existing system.

Chapter four will come to point up the technical realization of the application, Technologies Used, presentation of the New System (screenshots and brief descriptions),software Testing, hardware and Software Requirements, and deployment.

Chapter five will conclude and recommend related to results of the project

CHAPTER 2

ANALYSIS OF THE CURRENT SYSTEM

Introduction

This part is meant to provide a brief description about terms that are being used during development of the project. After a brief presentation of Online Banking System given, subsequently a thorough analysis of the existing system like working principle, problems within the system, all its features and characteristics which are relevant to our upcoming work are also described. Before the end of this chapter, proposed solution to solve the mentioned problems will be highlighted.

Description of the Current System Environment

Historical Background

Equity Bank Rwanda began its operations in 2011 and is registered as a commercial bank by the National Bank of Rwanda. The Bank has its Head Office located in Kigali, with a foot-print of 14 branches and is supported by 2421 agents, 1074 merchants and a network of 21 ATMs. (*Equity, 2022*)

Equity Bank Rwanda is a subsidiary of Equity Group Holdings Plc, a financial services company listed at the Nairobi Securities Exchange, Uganda Securities Exchange, and Rwanda Stock Exchange. In addition to Equity Bank Rwanda, the Group has banking subsidiaries in, Kenya, South Sudan, Uganda, Tanzania, DRC, and a Commercial Representative Office in Ethiopia; with additional non-banking subsidiaries engaged in the provision of investment banking, custodial, insurance agency, philanthropy, consulting, and infrastructure services.

Equity Group has an asset base of over USD 10 billion. With over 14.2 million customers, the Group is one of the biggest banks by customer base in the region. The Group has a footprint of 335 branches, 52,742 Agents and 35,386 Merchants and 720 ATMs. The Group is the largest bank in market capitalization in East and Central Africa. The Banker Top 1000 World Banks 2020 ranked Equity Bank 754 overall in its global ranking, 62nd in soundness (Capital Assets to Assets ratio), 55th in terms of Profits on Capital and 20th on Return on Assets. In the same year, Moody's gave the Bank a global rating of B2 with a negative outlook same as the sovereign rating of the Kenyan government due to the Bank's strong brand recognition, solid

liquidity buffers and resilient funding profile, established domestic franchise and extensive adoption of digital and alternative distribution channels. Equity Group Holdings Plc is regulated by the Central Bank of Kenya.(*Equity, 2022*)

Mission

We offer inclusive, customer focused financial services that socially and economically empower our clients and other stakeholders.(*Equity, 2022*)

Vision

To be the champion of the socio-economic prosperity of the people of Africa.(*Equity, 2022*)

Description of the Current System

The client with an account at the bank wants to ask a query to the bank. The client is on the way to the bank's branch in a certain city/village. Reaching the bank; finds a queue and waits for some minutes. Now it's time to ask a query, the client asks the question to the customer care person. The customer care person has no answer to the asked query or is not capable to answer the query, the customer care person asks the staff in charge. The customer care person takes the client to the staff in charge and gets the help directly from the staff or the customer care came with the exact answer to the asked question. Customer return to their home.
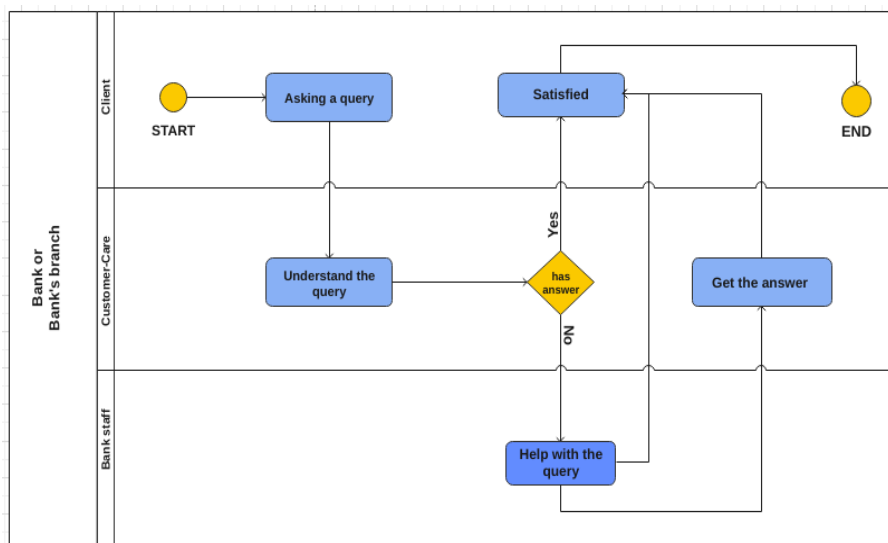
Analysis of the Current System

Users with or without a smartphone or data/internet connection can use mobile banking via the *___# code, which is provided by the Unstructured Supplementary Service Data (USSD). Fund transfers, checking account balances, and producing bank statements are all possible using USSD-based mobile banking. An Access bank's bot which is deployed on WhatsApp, can't handle customers' queries; what it does it gives the choices to select from like press 1 to view your account balance. It is similar to the USSD the only difference is where they are deployed. But the solution that I am bringing in is creating an intelligent bot that will act more as a customer service assistant to answer the frequently asked queries of the customers.

Equity group specifically has Customer Relationship Management(CRM); It is a web-based application integrated on their website, whereby a customer can talk to them by submitting the question and get a response in 48 hours. Sentiment Analysis is a common text categorization

tool that examines an incoming message and determines if the sentiment is positive, negative, or neutral; Equity uses it to analyze the sentiments of their customers on Twitter. Whereby the positive are those who appreciate the bank, negative are those who are complaining about the bad service given and neutral mostly are the questions that customers have and from there the bank knows who needs help more than the other and work accordingly to maintain the brand's good reputation. Another way that the bank uses to assist customers is by responding to the customer's calls and also by helping them physically.

Modeling of the current system

Figure 1: **Modeling of the current system**

Problems of the current system

This System is required to solve the following problems which may occur when it's run out:

- Calling the bank's number and no one picks the call.
- Posting a query to the CRM and waiting for 48 hours not helping always (In case it's urgent).
- It may be difficult for the customer to reach the bank physically.
- The customer care person may have a lot of clients and not knowing how to handle them.
- Finding out that they don't provide the service that you wanted.
- Waiting in the queue and finding out that the other person is the one to help.
- The customer may need to know something after the hours of work.
- Client dissatisfaction may lead to the decline of the bank's good reputation.

Proposed Solutions

The proposed system is a technology based solution which will provide the following solutions according to the problems stated above:

➢ **Easy way of communication between customers and the company**: The system will allow a customer to ask any bank's related query and get the reply.

➢ **Faster Response:** after the machine learning has understood the query , the customer will be able to get a reply in less than a minute.

➢ **A better way of keeping records:** after receiving your response the bank will be able to keep your records for future use. Just in case The system fails to answer your query the person in charge will be able to train the model with that query

➢ **Computer Doing Human Work:** the best part of the system is that the computer will be doing human's work.

System Requirements

Functional Requirements

- The system should allow the user to ask a query.
- The User should use text in natural language while asking a question.
- The system should respond to the asked query.
- The chatbot should provide a text response. In natural language.
- The system should inform the user that it's not a human but a robot if a user asks.
- When the context of prior messages is unclear, the chatbot should be able to keep the discussion going.
- Users will be able to chat with the Chatbot through text commands and it will comprehend what the user is saying through natural language understanding processing.

Non-Functional Requirements

Performance

- The chatbot must be coherent with very little lag in response time for example the chatbot should be able to reply to a user message in less than 5 seconds.

Reliability requirement

- The chatbot must be dependable with next to no faults or bugs.

Usability

- The use of natural language to communicate with the chatbot encourages human-computer interaction.

Efficiency

- Provide error-free responses to queries.

- Proper handling of unexpected input &, and correctly inform the user if it cannot provide assistance.

Maintainability

- The system should be easy to maintain once needed.

Security

- Only the admin can view all the chats from different users.

Availability

- The chatbot must be available 24/7 to assist the customer.

CHAPTER 3

REQUIREMENTS ANALYSIS AND DESIGN OF THE NEW SYSTEM

Introduction

The objective of this chapter is to make retailed review and deepened in the goal to understand the current system and the failing of it in order to propose an adequate computerized solution.

The success or failure of the project is determined by the requirements . They are the follow-on work's basis . They are crucial in achieving the Users' objectives. Requirements are important because they provide the basis for all of the development work that follows. Design states how to accomplish the objective and Analysis specifies what the new system should do.

System Design is the process of creating the architecture, modules, and components of a system, as well as the many interfaces between those components and the data that flows through it. The specs, which must specify both what the proposed system will do and how it will work, will be the phase's result. The new system must be designed based on the user needs and a complete study of the present system. This is where the system is designed. It is the most essential stage in a system's development. (*Dennis et al., 2012*)

In **System Analysis**, a greater emphasis is placed on comprehending the details of an existing or proposed system, and then assessing whether the new system is desirable or not, as well as whether the present system requires improvement. As a result, system analysis is the act of studying a system, detecting flaws, and using the data to provide recommendations for system improvements. (*Rumbaugh et al., 2004*)

In this chapter, we analyzed requirements of the existing system by identifying problems that users face and design the proposed system in order to get solutions to those problems. But before getting into this in details, let us start by discussing the techniques and tools that were used for that end.

Analysis and Design Methodology

Analysis of the new system

This analysis phase answers who will use the system, when it will be used, where the system will be used, and what the system will do. In this stage, the project inventor or researcher investigates the current system which has been described in the previous chapter, identifies opportunities for improvement, and develops a concept for the new system.

Analysis is defined as the procedure by which we break down an intellectual or substantial whole into parts. (*Dennis et al., 2012*)

System analysis is a problem-solving technique that breaks down a system into its component parts in order to investigate how well those parts operate together to achieve their goals. This is a technique for creating new systems. Investigation, design and implementation are all aspects of the systems analysis process.(*Wasson, 2006*)

Unified Modeling Language (UML)

In the realm of software engineering, the Unified Modeling Language is a standardized general-purpose modeling language. UML is a set of graphic notation tools for creating software system visual models. UML is a modeling language that incorporates concepts from data modeling, business modeling, object modeling, and component modeling. It can be used at any stage of the software development life cycle and with a variety of implementation technologies. (*Rumbaugh et al., 2004*)

UML is used in a variety of ways to support a software development methodology but in itself it does not specify that methodology or process.

UML has three building blocks: elements for representing models, relationships that tie elements together and diagrams mean to group interesting collections of elements and relationships.(*Rumbaugh et al., 2004*)

Design of the New System – Diagrams

The first phase of the system development life cycle is System design because it provide a concrete understanding of how the system will operate.

Use case diagram

In system analysis a process used to identify, clarify, and organize system requirements is a Use Case. It is made up of a set of possible sequences of interactions between systems and users in a particular environment and related to a particular goal. All system operations that are important to the users should be included in the Use Case. A Use Case is a collection of possible scenarios that are related to a specific goal. (*Dennis et al., 2012*)

A Use Case is the description of the model "View" by the actors in the system. It corresponds to the expected needs of each actor (the WHAT and WHO). Use Cases are used to represent the operation of the system vis-à-vis the User, so this is a view of the system in its external environment. (*Wasson, 2006*)

Elements

Elements and definitions	Symbols
● **Use case**: is a business process model that has been simplified. It is a collection of activities that make up a system. The use case is described from the perspective of the actors involved (i.e. those actors interacting with the system). (*Dennis et al., 2012*).	UseCase1
● **Actor**: When interacting with a system directly, an actor specifies a role that some external entity takes on. It could be a user role or a role played by another system that interacts with one.	Actor1
● **System boundary:** It's a box drawn around the use case to represent the system's edge or border. A system boundary is represented by a rectangle drawn to encompass the internal components of a system in a use case diagram. The actors are any entities outside of the rectangle.	Name
● "**Extend**" relationship: «extend» An extending Use Case is used to describe variations in the normal flow	

of events described by a general use case. Use Case B extends Use Case A when Use Case B describes the behavior of Use Case A under a particular condition. (*Dennis et al., 2012*)	– – – – – –«extend»– – – – –>
● "**Include**" relationship: «include» in a relationship When Use Case D is a behavior or functionality that Use Case C requires, Use Case C incorporates Use Case D. Because that action is required in multiple use cases, it has been built into its own Use Case.	– – – – – –«include»– – – – –>
● "**Association**"relationship: It is symbolized by a line and it acts as an interface between an actor and a use case.	————————

Table 1: **Definitions of use case elements**

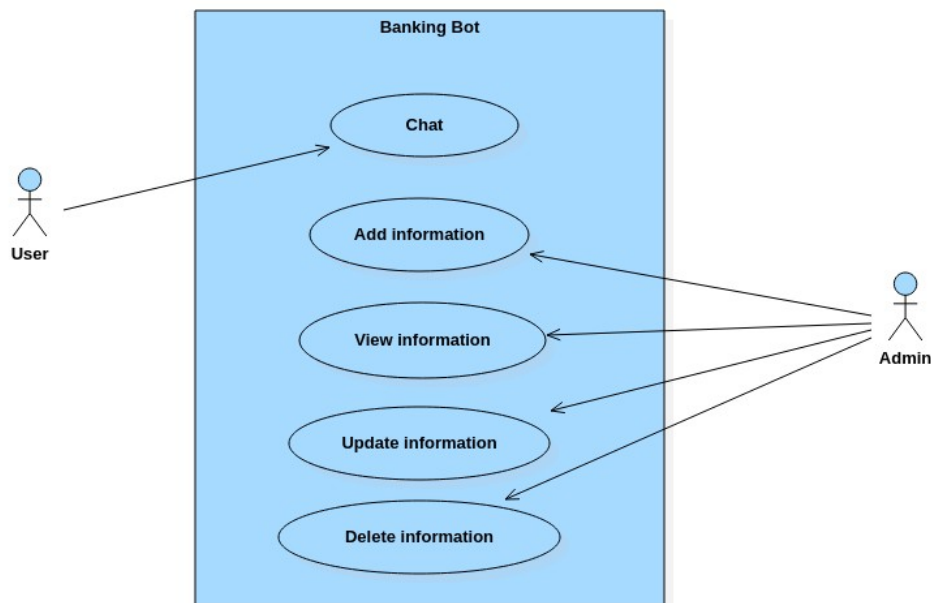**Use case for Equity Bank Chat-bot**

Figure 2: **Banking bot Use Case**

15

Use Case Description

Use Case description describes in details what a use case do, and what it requests in order to be well executed. Each Use Case looks like this:

> **Name:** a name of the Use Case

> **Description**: what the system intends to do

> **Actor**: the actor involved in the Use Case

> **Pre-condition(s)**: the system state before the Use Case can begin

> **Post-condition(s):** the system state when the Use Case is over

> **Main normal flow**: the actual steps of the Use Case

> **Alternative flow**: steps which may happen in case a normal flow fails.

1. **Chat -Use Case Description**

Name: Chat
Actor: User
Description: Help for User to chat with the bot
Pre-condition: the User should have and logged in into his/her Facebook account
Post-condition: the system save all chats that a User had with the bot
Main Flow: 1. User search for the bot on Facebook messenger, 2. User start and greet the bot, 3. Bot greets back to the User, 4. User asks any bank related query, 5. Bot reply to the asked question, 6. repeat 4 and 5 until the User is satisfied, 7. User says thanks or byee to the bot, 8. The bot thank the User too and the conversation is done.
Alternative flow:

A. the bot fails to understand what the user is asking, then the bot reply by asking the user to rephrase.

B. The User rephrase the question.

C. the conversation keeps going until the user is satisfied.

Table 2: **Use case for chatting**

2. Add information -Use Case Description

Name: add information
Actor: Admin
Description: Help the admin to add information
Pre-condition: should have that permission from Equity bank
Post-condition: Should be assured that the information was added.
Main Flow: 1. Admin opens an administrator view 2. Admin add a new intent to the bot, 3. save changes, 4. ask the bot with the added question, 5. Use case end
Alternative flow: A. if the provided intent information is not valid the system shows the error where is not correct. B. Admin provide information again.

Table 3: **Use case for adding information**

3. View information -Use Case Description

Name: view information
Actor: Admin
Description: Admin View the information
Pre-condition: should have that permission from Equity bank
Post-condition: should see the chats
Main Flow: 1. Admin opens an administrator view

| 2. Admin view all the chats that the bot had with different users, |
| 3. Use case end |

<div align="center">Table 4: Use case for Viewing information</div>

4. Update information- Use Case Description

Name: Update information
Actor: Admin
Description: Allows an Admin to update the information
Pre-condition: should have that permission from Equity bank
Post-condition: the system will save updated information
Main Flow:
1. Admin opens an administrator view,
2. Admin views how the bot has replied to the User,
3. correct the bot where it replied incorrectly,
4. save the changes,
5. Use case end

<div align="center">Table 5: Use case for updating information</div>

5. Delete information-Use Case Description

Name: delete information
Actor: Admin
Description: Allow admin to delete the information
Pre-condition: Should have that permission from Equity bank
Main Flow:
1. Admin opens an administrator view,
2. Admin deletes a no longer needed intent,
3. Use case end

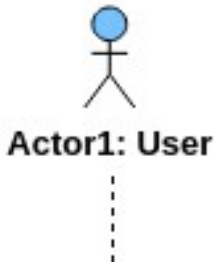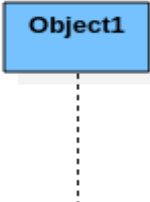<div align="center">Table 6: Use case for Deleting information</div>

Sequence diagram

An interaction diagram that describes in details how operations are carried out is Sequence Diagrams: what messages are sent and when sequence diagrams are organized according to time. The time progresses as you go down the page. (*Rumbaugh et al., 2004*)

The operation's objects are presented in order of when they appear in the message sequence, from left to right. A lifeline is a vertical dotted line that represents the amount of time a thing has existed. Each arrow represents a message on the lifeline of the receiver. The duration of the message's execution is shown by the activation bar. The most popular type of interaction diagram is a sequence diagram, which focuses on the message exchange between a number of lifelines.

A sequence diagram depicts an interaction by focusing on the message sequence that is transmitted, as well as the lifelines' related occurrence specifications.

Here are some notations and their definitions which we are going to use in sequence diagram:

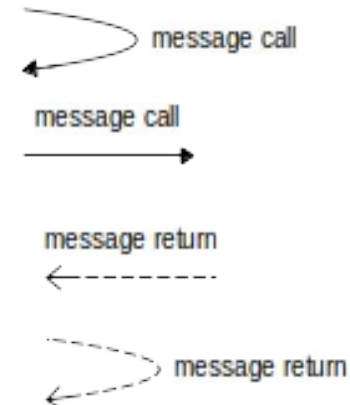| Terms and Definitions	Symbols
An actor: ✓ It can be a person or system but external to the system that drives benefit from sequence. ✓ It partake in the series by sending and / or receiving messages. ✓ It is positioned across the top of the diagram	Actor1: User
An object: ✓ It partake in sequence diagram by sending and /or receiving messages. ✓ At the top of the program that's where it is placed.	Object1
A lifetime: ✓ Symbolize the life of an object during a sequence.	
An activation: ✓ Is a lengthy narrow rectangle positioned atop a lifeline ✓ It symbolize when an object is sending or receiving a message.	

A message:	
✓ It brings information from an object to another one. ✓ A return is labeled with the value being returned and depicted as a dotted arrow, whereas an operation call is labeled with the message being delivered and a straight arrow.	

Table 7: **Table of symbols used in sequence diagram**

Chat

Sequence Diagram for Chat will help the User to chat with our virtual assistant by asking any bank related query and get the reply to the asked question.
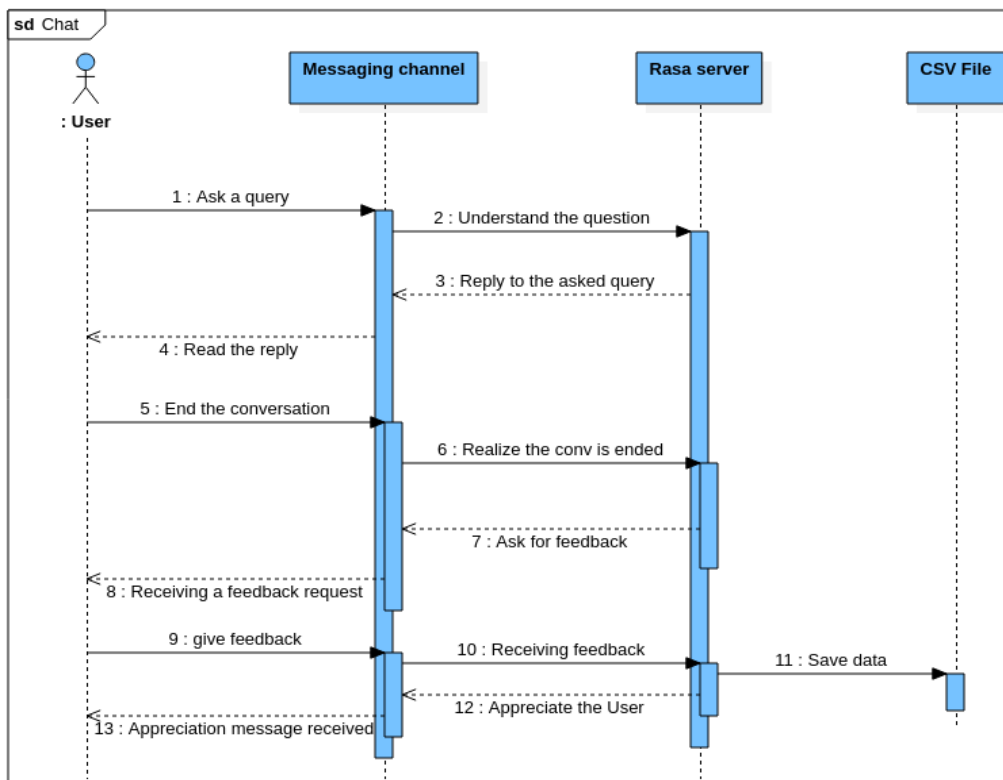

Figure 3: **User Sequence diagram**

Administrator control

Sequence Diagram for the role of administrator in training data, viewing customer's chats.
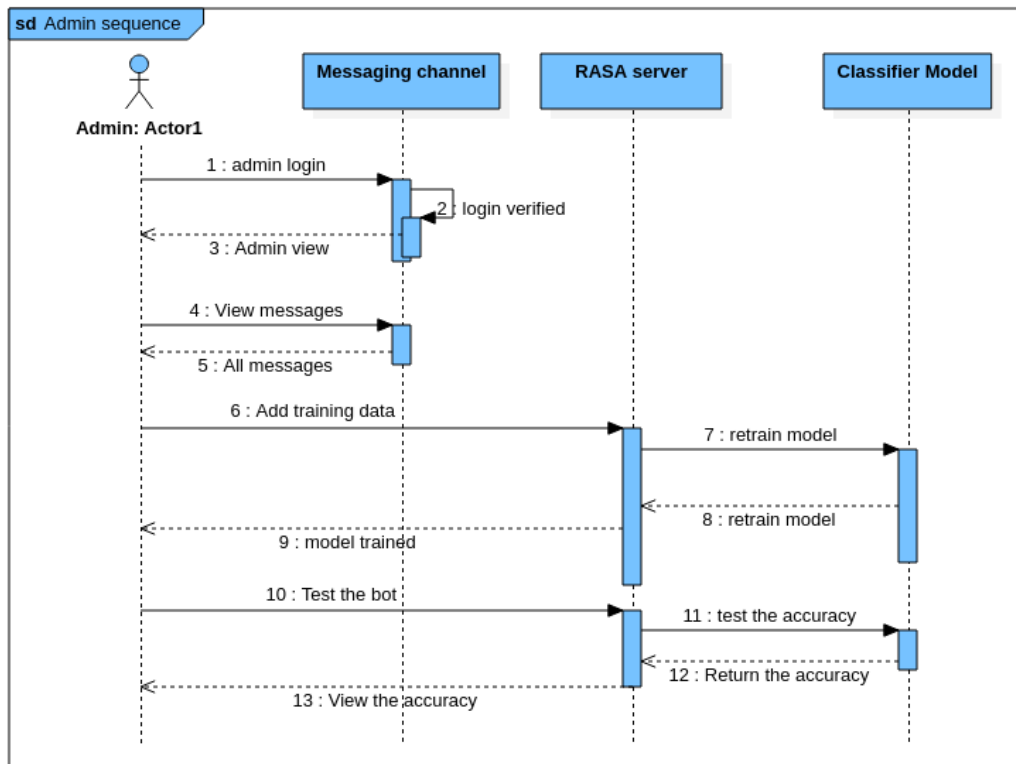

Figure 4: **Admin Sequence diagram**

Architectural design

The primary pieces or functions of a system are represented by blocks connected by lines in a system architecture design. System Architecture takes the conceptual design of an information system and translate it into a physical structure that comprises hardware, software, network support, and processing methods. (*Dennis et al., 2012*).

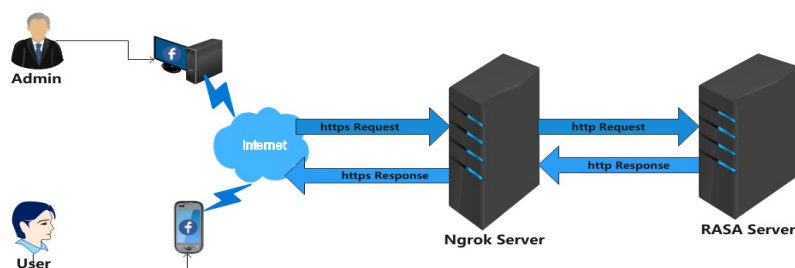The following figure displays sample system architecture of the new application.

Figure 5: **System Architectural Design**

21

Data Processing

This chapter basically explains and looks at how the general system works. It also looks at how the classification is able to compare the parameter input data from the customers with the trained data.

Dataset

One of the best practices for designing NLU training data is by using real data; by collecting them direct from the user. With this chatbot real data was used by training the model. Internet helps a lot with the bank's FAQ(Frequently Asked Questions) of different banks in addition to internet Friends and family has also played a hug role and with the help of Munezero (an equity bank employee with 9 years of experience) I have been able to adjust the data to equity bank specifically. This chatbot will also be creating a dataset that can be used in the future.

n	Parameter	Description	Range	Analysis of data
1	Questions	Customers Queries	A-Z or a-z 0-9	Questions(represented by A-Z, a-z and 0-9)
2	Responses	Customer care's responses	A-Z or a-z 0-9	Text Responses(represented by A-Z and/or a-z

Table 8: **Banking_bot Dataset atributes**

	A	B	C
1	Questions	Responses	
2	Can I transfer my Current Account from one branch to another	you can but once you want to borrow some money	
3	What are the documents required for opening a Current Account of a sole proprietorship firm	Business Current Account	
4	What is the time taken to realize outstation cheques	The time taken to realize outstation cheques is 2 days	
5	How is Equity Bank Premium CA different from a regular CA offered by most banks	Equity has no premium account	
6	Can I access my account by telephone to conduct routine daily transactions	yes, through Eazy banking	
7	what transactions can I perform on my account by telephone and how	You can	
8	What is the initial deposit amount required to open the Current Account	Initial deposit amount required is 1000	
9	What should I do in case I am unable to access my Company's Current Account using NetBanking	raise your request through CRM	
10	How much cash can I deposit per month in my home branch, without being charged	No chargers on deposit	
11	What is the minimum balance that I need to maintain in a Current Account	no minimum	
12	Will both the principal and interest amount of my Fixed Deposit be considered as a guarantee when I'm a applying for a loa	Yes, principal and interest amount of ones fixed Deposit can be considered a	
13	Is there a nomination facility	Yes, there is a nomination facility on the Recurring Deposit account.	
14	Is there a monthly or quarterly interest pay-out option	Interest is paid when the Recurring Deposit account is closed, and on maturi	
15	Can I change the tenure and installment due date, (auto debit date) of my Recurring Deposit	yes you can	
16	For what period can I open a Recurring Deposit	You can open a Recurring Deposit account for a minimum period of 6 month	
17	Can I choose my instalment date	Yes you can negotiate	
18	Can partial Payments be made on Recurring Deposits	Partial payments are not possible.	
19	What is the maximum deposit amount a Recurring Deposit account can be opened with	no maximum	
20	What if certain RD installments are not paid	If six consecutive installments are unpaid, the Bank reserves the right to clos	
	How do Recurring Deposits work	A monthly deposit of a pre-fixed amount is made in the Recurring Deposit ac	

Figure 6: **Sample of Dataset**

n	Parameter	Description	Range	Analysis of data
1	intent	Classes of user input	A-Z or a-z 0-9	Classes names(represented by A-Z, a-z and/or 0-9)
2	User_input	All things that a user can say to the bot	A-Z or a-z 0-9	Text input(represented by A-Z) numeric input(represented by 0-9)
3	Entity_name	Name of an entity extracted	name	Value name(a user has provided a name)
4	Entity_value	Value of an entity	A-Z or a-z	The exact users name extracted(represented by A-Z and/or a-z)
5	Action	Bot's action name	utter_a-z or utter_a-z_0-9	Action name(represented by utter_a-z and/or 0-9)
6	bot_reply	Bot's responses	A-Z or a-z	Text output(represented by A-Z and/or a-z)

Table 9: **Banking_bot generated dataset attribute**

Figure 7: **Sample of generated dataset**

Algorithm Discussion

Before we dive deep in the algorithm let's first understand how chatbot works:

What are Chatbots?

Chatbot/bot is a computer program that simulates a natural human conversation. Users communicate with a chatbot like how they would talk to a real person via the chat interface or by voice.

There are three main types of chatbot:

Rule-based chatbots

This is a straightforward type of chatbots. People interact with these bots by using pre-defined options and clicking on buttons. To give a propriate answers these chatbots require people to make a few selections. As an outcome, these bots have a longer user journey, and they are the slowest to guide the customer to their goal. (Følstad et al., 2022)

Intellectually independent chatbots

Machine Learning (ML) is used in these bots to help them learn from their users' inputs and requests. Intellectually independent bots are taught to recognize and respond to certain keywords. They learn to grasp more and more queries as time goes on. Any conversational AI chatbot can be said to learn and train through experience. (*Følstad et al., 2022*)

AI- powered chatbots

AI-powered bots amalgamate the best from Rule-based and Intellectually independent. AI-powered chatbots perceive free language, but also have a predefined flow to make sure they solve users' problems. They can recall the context of the conversation and the user's preferences. These chatbots can jump from one point of conversation scenario to another when needed and address random user requests at any moment.(*Følstad et al., 2022*)

These chatbots utilize Machine Learning, AI, and Natural Language Processing (NLP) to understand people.

Natural language processing (NLP)

Natural language processing(NLP) refers to the branch of computer science and more specifically, the branch of artificial intelligence or AI, concerned with giving computer the ability to understand text and spoken words in much the same way human beings can (*IBM Cloud Education, 2020*). While this level of understanding is still quite a way off, it is one of the goals of Artificial Intelligence—for the computer to accurately understand English (and other languages) and be able to extract meaning from the words. (*D.Booth, 2018*)
And we'll be developing an AI-powered chatbots.

NLP in Chatbots

NLP comprises mainly with two major functionalities, The first is "Human to Machine Translation" (Natural Language Understanding (**NLU**)), and the second is "Machine to Human translation"(Natural Language Generation (**NLG**)).

Natural Language Understanding (NLU)

The NLU unit is in charge of converting the user utterance to a predefined semantic frame according to the system's conventions, i.e. to a format understandable for the system. This is made up of a task of slot filling and intent detection. For example, the intent, could be a greeting, like Hello, Hi, Hey, or it could have an inform nature, for example I like African food, where the user is providing some extra information. Depending on some interests, the slots could be very diverse, like the user name, price, start time, destination city etc. As we can see, the intents and the slots are defining the closed-domain nature of the Chatbot. The slot filling task and intent detection task is seen as a sequence tagging problem. For this fact, the NLU component is usually implemented as an LSTM-based neural network with a Conditional Random Field (CRF) layer on top of it. The model issued is a sequence-to-sequence model using bidirectional LSTM network, which fills the slots and predicts the intent in the same time. On the contrary, the model is doing the same using an attention-based RNN. To achieve such a task, the dataset labels consist of: concatenated B–I–O (Begin, Inside, Outside) slot tags, the intent tag and an additional end-of-string (EOS) tag. As an example, in a restaurant reservation scenario, given the sentence Are there any French restaurants in Toronto downtown?, the task is to correctly output, or fill, the following slots: {cuisine: French} and {location: Toronto downtown}. (*D.Booth, 2018*)

The following image shows the classification process for intent classification using Neural Network as,
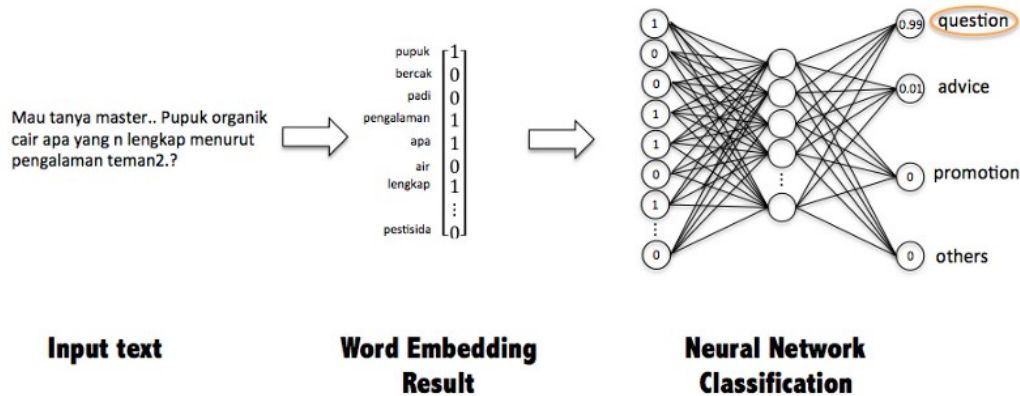
Figure 8: **Classification Process using Neural Network**

The NLU Pipeline

The NLU pipeline defines all the steps that will be used in the pipeline to extract intents and entities. It starts with text as input and passes through all steps until it has entities and intents as output.(*Rasa, 2022*)

Figure 9: **NLU Process**

There are various types of components that you can expect to find in a pipeline. The main ones are:

- Tokenizers
- Featurizers
- Intent Classifiers and Entity Extractors

26

Tokenizers

The first thing is to break an utterance into smaller chunks of text, known as tokens. This must happen before text is featurized for machine learning. There are different types of tokenizers in use like; WhitespaceTokenizer, JiebaTokenizer, MitieTokenizer, SpacyTokenizer, etc(*Rasa, 2022*). In this project I have used WhitespaceTokenizer.

WhitespaceTokenizer

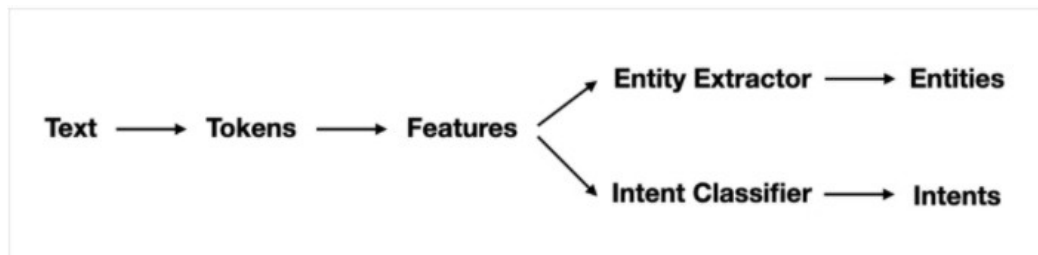WhitespaceTokenizer it's a Tokenizer using whitespaces as a separator. generate a token for all whitespace separated character sequence.

Any character not in: a-zA-Z0-9_#@& will be substituted with whitespace before splitting on whitespace if the character fulfills any of the following conditions:

- the character follows a whitespace: " !word"→ "word"
- the character precedes a whitespace: "word! " → "word"
- the character is at the beginning of the string:"!word" → "word"
- the character is at the end of the string: "word!"→"word"

Note that:
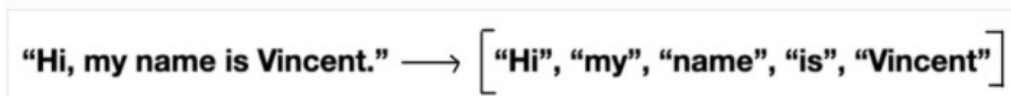
- "wo!rd"→ "wo!rd"

In addition, any character not in: a-zA-Z0-9_#@&.~:∨?[]()!$*+,;=- will be substituted with whitespace before splitting on whitespace if the character is not between numbers:

- "twenty{one"→"twenty","one" ("{"` is not between numbers)
- "20{1" →"20{1" ("{"`*is* between numbers)

Note that:

- "name@example.com"→"name@example.com"
- "10,000.1"→"10,000.1"
- "1 - 2"→"1","2"(*Rasa, 2022*)

Figure 10: **Tokenization**

Featurizers

Featurizers generate features in form of numbers for machine learning models. Text featurizers is divided into two different categories: sparse featurizers and dense featurizers. Sparse featurizers are featurizers that return feature vectors with a lot of missing values, e.g. zeros. As

those feature vectors would normally take up a lot of memory, we store them as sparse features. Sparse features only store the values that are non zero and their positions in the vector. Thus, we save a lot of memory and are able to train on larger datasets.(*Rasa, 2022*)

There are two types of features that all featurizers can return: sequence features and sentence features. The sequence features a size matrix (number-of-tokens x feature-dimension). Every token in the sequence has a feature vector in the matrix. We can use this to train sequence models. A matrix of size represents the sentence features (1 x feature-dimension). It contains the entire utterance's feature vector. Any bag-of-words model can benefit from the sentence properties. As a result, the appropriate classifier can choose which traits to employ. It's worth noting that the feature-dimensions for sequence and sentence features don't have to be the same. (*Rasa, 2022*)

In this project I have used **CountVectorsFeaturizer** which fall in the sparse featurizers category. In short it Creates **bag-of-words** representation of user messages, intents, and responses. The example will be shown later.

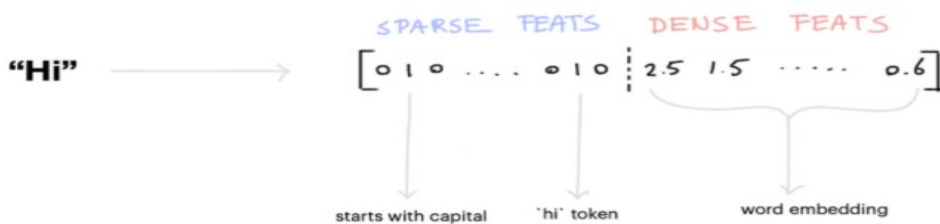The diagram below shows how the word "Hi" might be encoded.

Figure 11: **Word Featurization**

Other than features for tokens, we also generate features for the entire sentence. This is sometimes also referred to as the CLS token. The sparse features in this CLS token are a sum of all the sparse features in the tokens.
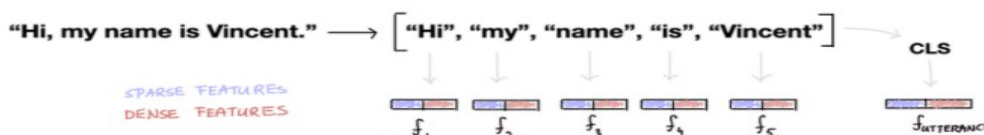

Figure 12: **Sentence Featurization**

Intent Classifiers and Entity Extractors

Once features for all of the tokens and for the entire sentence are generated, we can pass it to an intent classification model and entity extractor model. It classify intents and extract entities. There are different intent Classifiers: MitieIntentClassifier, LogisticRegressionClassifier, SklearnIntentClassifier, etc and in Entity Extractors we have: MitieEntityExtractor, SpacyEntityExtractor, CRFEntityExtractor, RegexEntityExtractor , etc. In this project DIETClassifier was used.

DIETClassifier

Dual Intent Entity Transformer (DIET) used for intent classification and entity extraction. To match the intent, DIETClassifier use Neural Network machine learning algorithm feedforward networks to be specific
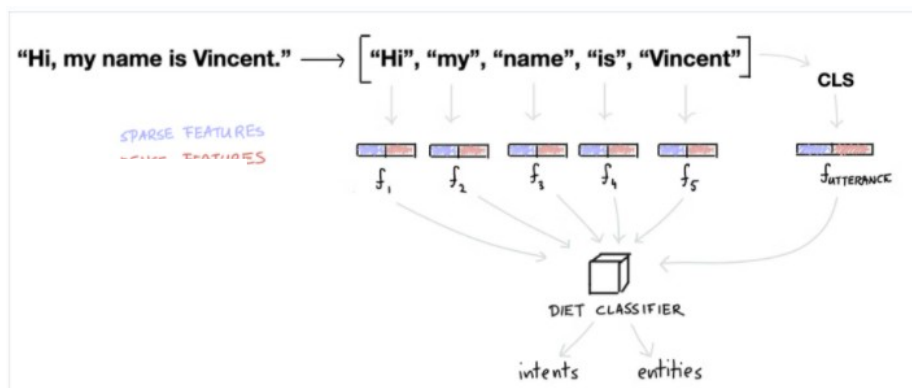
Figure 13: **DIET Classifier**

Interaction: Message Passing

The NLU pipeline is a series of component These components are trained and processed in the order they are listed in the pipeline. This means that a pipeline configuration can be thought of as a linear series of steps that the data needs to pass through. The diagram below gives an overview of what happens when the Message is refined.

Figure 14: **Message Processed**

Natural Language Generation(NLG)

Natural Language Generation (NLG) is a branch of Artificial Intelligence (AI) that tries to bridge the gap between machines and humans in terms of communication. In general, technology absorbs non-linguistic input and converts it into human-readable representations such as reports, papers, and text messages. (D.Booth, 2018)

Here is how NLU and NLG works together,

Key:
NLP - Natural Language Processing
NLU - Natural Language Understanding
NLG - Natural Language Generation

Figure 15: **NLU and NLG**

Let's take a look on how decision engine works, which is simply being referred to as **Core Policies**. How the prediction is made after classifying intents and entities.

Policies

Your assistant uses policies to decide which action to take at each step in a conversation.
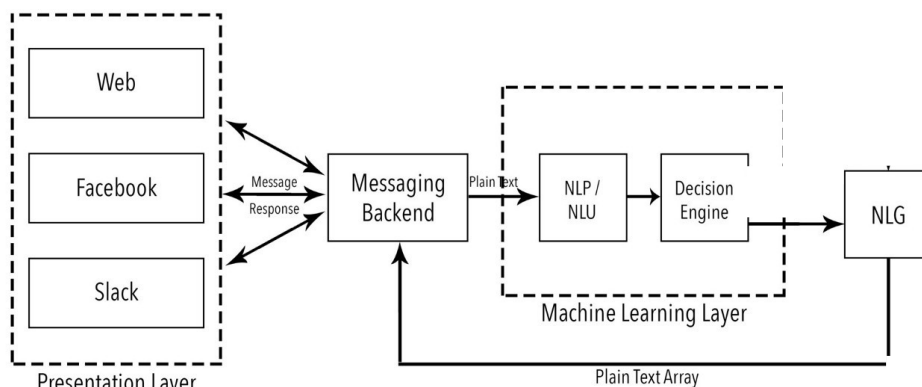
1. The **RulePolicy** controles conversations that match predefined rule patterns. It makes predictions based on any rules you set.
2. The **MemoizationPolicy** examine if the current conversation matches any of the stories in your training data. If so, it will predict the next action from the matching stories.
3. The **TEDPolicy** uses machine learning to predict the next best action. Long Short Term Memory (LSTM) machine learning algorithm was used.

These policies works in a priority based hierarchy. If we count upon the standard settings, then the **RulePolicy** is considered before the **MemoizationPolicy** which in turn is considered before **TEDPolicy**. (*Rasa, 2022*)There are many policies like UnexpecTED Intent Policy, Augmented Memoization Policy, Rule-based Policies, etc.

Machine learning algorithm used

The most popular algorithms used by conventional Chatbots are Naïve Bayes, Decision Trees, Support Vector Machines, Neural Networks (NN), Markov Chains, Long Short Term Memory (LSTM) and Natural Language Processing (NLP). Classification algorithms are used by chatbots primarily to identify the intent in phrases. This helps in deriving context pertaining to that intended by the user in the input. Topic Modeling using algorithms such as LDA and feature extraction (NLP) is useful in extracting the overall topic or domain of the conversation. Naïve Bayes is useful for determining how confident that chatbot is in its prediction, and Decision Trees are more appropriate for determining how the chatbot arrived at the classification prediction that it did. NN and LSTMs are excellent algorithms to process textual data efficiently. And they are the excellent algorithm they were used in this project.(*V. et al., 2020*)

$$p(x=v|C_k)=\frac{1}{\sqrt{2\pi\sigma_k^2}}\,\mathrm{e}^{\frac{-(v-\mu_k)^2}{2\sigma_k^2}}$$

Equation 1: **Artificial Neural Network Classifier**

Where σ^2 is the variance of the values in x, and μ is the mean of the values in x. Therefore, Artificial Neural Network, assume we have training data which comprises of continuous attribute x, we shall fragment the data by class.

After, we compute the mean μ and the variance σ^2 of x per class. We let μ_k be the mean of the values in x for class C_k, then it follows that we let σ_k^2 be the variance of the values of x for class C_k. Thus, assume we have gotten some observation values x_i. Therefore, we have the probability density for x_i for class C_k as . We put x_i to the Gaussian distribution equation with parameters μ_k and σ_k^2.

In essence, the Artificial Neural Network classifier is an extension of the traditional Naïve Bayes classifier and the method takes each data point, and allocates it to whichever class it is nearest to. Since Gaussian Naïve Bayes is derived from Naïve Bayes probabilistic model basing on Bayes theorem, we have to first analyze Naïve Bayes probabilistic model. Given a mathematical dataset x = $(X_1, \ldots, X_n)$ to be classified Naïve Bayes allocates to an sample a discrete probability $p(C_k|\ X_1, \ldots, X_n)$. For k-classes in the dataset. To study this multivariate distribution might need a big quantity of data. Therefore, to simplify the assignment of learning, we assume that the features are conditionally independent from each other given the class. Therefore leading to the application of Bayes' theorem.

$$p(C_k|x) = \frac{p(C_k)\, p(x|C_k)}{p(x)}$$

Equation 2: **Bayes theorem**

By conditional probability, the numerator is the *joint probability distribution $p(C_{k,x})$* and may be factored through chain rule,

$$p(C_k, x) = p(x_1|x_2, \ldots\ldots, x_n, C_k)\, p(x_n|C_k)\, p(C_k)$$

Equation 3: **Chain rule**

Thus, by using the hypothesis of conditional independence of features, i.e. each feature x_i is conditionally independent from all other features x_j for $j \neq i$, we deduce

$$p(x_i|x_{i+1}, \ldots, x_n, C_k) = p(x_i|C_k)$$

Hence taking us to the expression of the joint probability model as

The equation above is a Naïve Bayes probabilistic model. To build a classifier, we ought to integrate a decision rule, like what we have with linear regression classifier. For our NB classifier, we implement the ***argmax*** function.

$$y = argmax_{k \in \{1,\ldots K\}}\left(p(C_k)\prod_{i=1}^{n} p(x_i|C_k)\right)$$

Equation 4: **Linear Regression**

Modeling and predicting

The entire project's main goal is to answer the customers queries with the towering accuracy. In order to attain this various classification algorithms will be tested. This segment contains all results acquired from the study and introduces the best performer according to accuracy metric. Several algorithms typical for solving supervised learning problems throughout classification methods. DIET Classifier, Mitie Intent Classifier, Logistic Regression Classifier, Keyword Intent Classifier and Sklearn Intent Classifier were tested. They are named as per those names in Rasa but in the background they use different machine learning algorithms. DIET Classifier uses Artificial Neural Networks, Mitie Intent Classifier uses Naïve Bayes, Logistic Regression Classifier uses Leaner Regression Classifier, Keyword Intent Classifier uses Decision Trees and Sklearn Intent Classifier uses support Vector machine.

Accuracy of algorithms

The results have been obtained by applying different classification algorithms. Table 9   holds accuracy of the different algorithms that we applied on our dataset.

classifier	Accuracy (%)
DIET Classifier	94.5
Mitie Intent Classifier	89.7
Logistic Regression Classifier	70.4
Keyword Intent Classifier	79.9
Sklearn Intent Classifier	77.6

Table 10: **Accuracy of different algorithm**

Confusion Matrix

The Confusion Matrix is a visualization of the performance of an AI model, by measure of its intent recognition. In other words, it shows you if your AI model is able to distinctly recognize similar expressions under different intents, or if it is "confused" by it. DIETClassifier uses Neural Networks (NN) in the background.(*Rasa, 2022*)
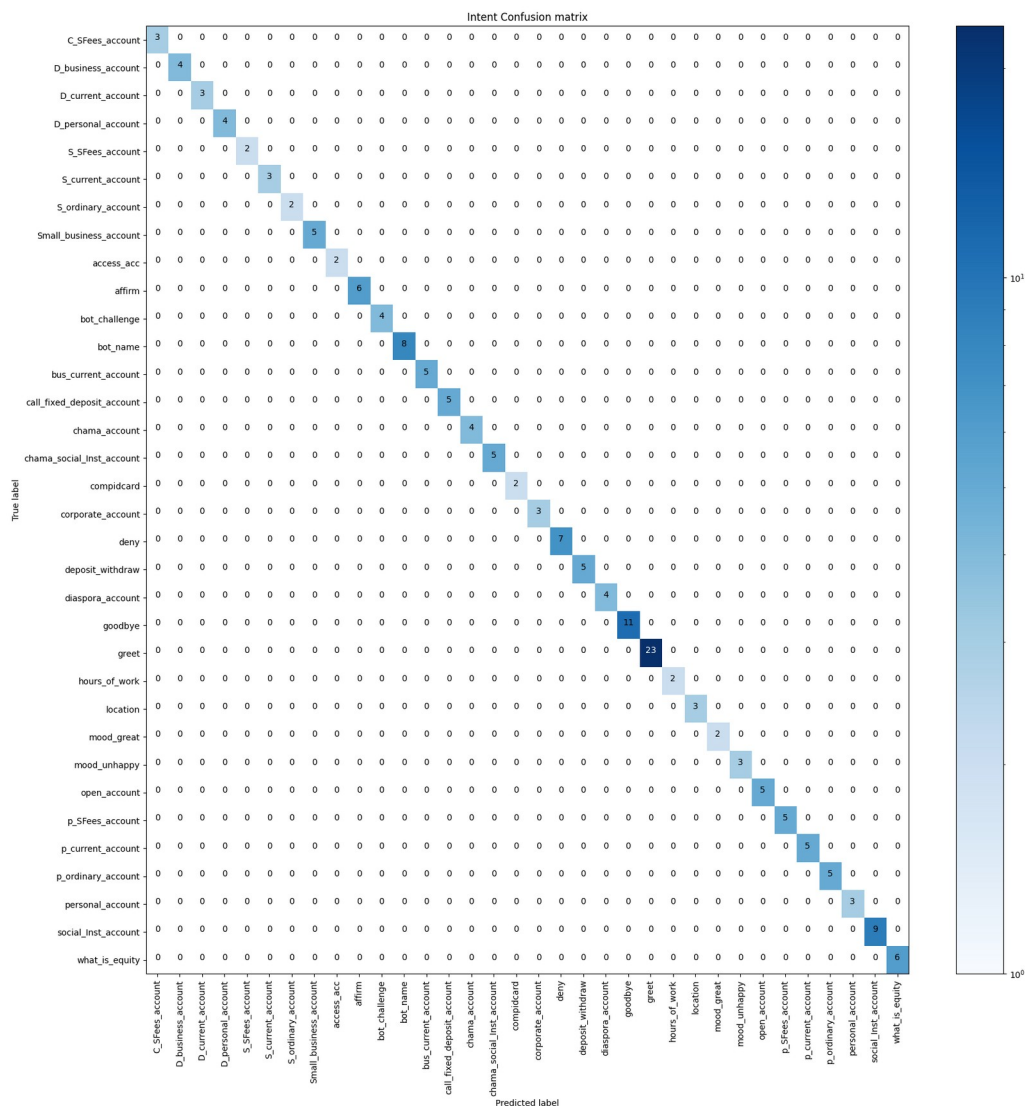

Figure 16: **DIETClassifierIntent Confusion Matrix**

The histogram allows you to visualize the confidence for all predictions, the incorrect predictions are displayed by red bars and correct predictions are displayed by blue bars.(*Rasa, 2022*) Bettering the quality of your training data will move the blue bars of the histogram up the plot and the red bars of the histogram down the plot. It should also avail in downsizing the number of red bars of histogram itself.
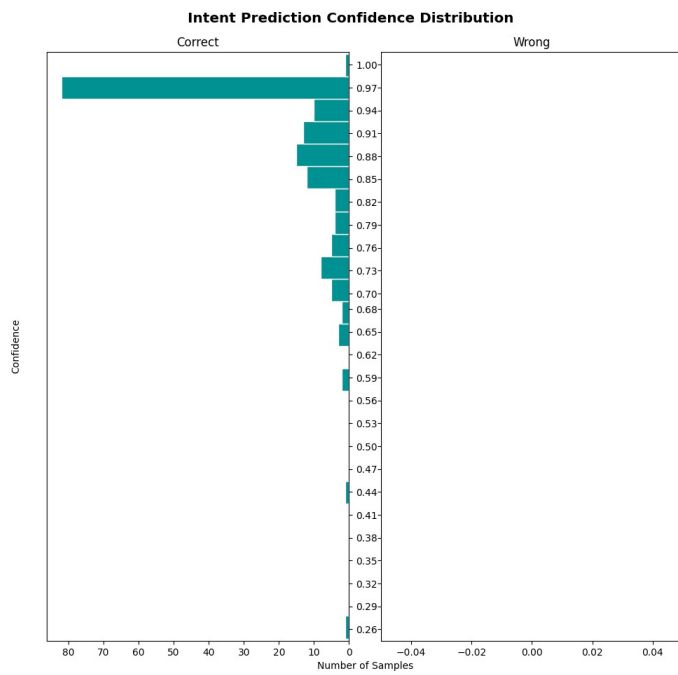
Figure 17: **Intent prediction confidence distribution**

CHAPTER 4

IMPLEMENTATION OF THE NEW SYSTEM

Introduction

This chapter recounts the development of the "**Banking bot System**", in this chapter, the analysis in the third chapter will be used in developing a new system, the technologies used in implementation will be described, some of the developed system interfaces and different tests will be done.

Technologies and Tools Used

To develop this application the following different technologies and tools were used:

Software Technology	Product
Server	RASA server, Ngrok
Source Code Editor or IDE(Integrated Development Environment)	Visual Studio Code 1.33.0
Chat-bot framework	RASA
Programming Languages	Python

Table 11: **Technology used**

RASA

Rasa is a python framework that help us to build any kind of Chatbot. It is based on NLU(Natural Language Processing) which offer the possibility to understand what the user want. Rasa is framework for developing AI powered, industrial grade chatbots. Its incredibly powerful, and is used by developers globally to fabricate contextual assistants chatbots.(*Rasa, 2022*)

RASA Server

The Rasa server gives endpoints to retrieve trackers of conversations as well as endpoints to modify them. Additionally, endpoints for training and testing models are provided. (*Rasa, 2022*)

Ngrok

ngrok is a cross-platform application that enables developers to expose a local development server to the internet with minimum effort. The software makes your locally-hosted web server appear to be hosted on a subdomain of ngrok.com, meaning that no public IP or domain name on the local machine is needed.

Visual Studio Code

A programming tool, also known as a software development tool, is a program or application that helps software developers design, debug, maintain, and support other programs and applications.
Visual studio code (VS Code) is a source code editor for programming in languages including C++, C#, Java, Python, PHP, Go, Perl, and others, as well as runtimes like.NET and Unity. Microsoft unveiled Visual Studio Code on April 29, 2015 at the 2015 Build conference. (*del Sole, 2017*)

Microsoft's Visual Studio Code offers debugging, embedded Git control, syntax highlight, intelligent code completion, and code refactoring for Windows, Linux, and macOS. It's also customisable, with users being able to alter the editor's theme, keyboard shortcuts, and options.

VS Code helps you be instantly productive with syntax highlight, bracket-matching, auto-indentation, box-selection, snippets, and more. Intuitive keyboard shortcuts, easy customization and community-contributed keyboard shortcut mapping let you navigate your code with ease. VS Code also integrates with build and scripting tools to perform common tasks making every day works faster.

VS Code includes enriched built-in support for Node.js development with JavaScript and Typescript, powered by the same underlying technologies that drive visual studio. VS Code also includes great tooling for web technologies such as JSX/React, HTML, CSS, SCSS, Less and JSON.(*del Sole, 2017*)

37

Python

Python is a widely-used, interpreted, object-oriented, and high-level programming language with dynamic semantic, used for general-purpose programming. It was created by Guido van Rossum, and first released on February 20, 1991. (*Lutz, 2013*)

Python is frequently used to create websites and applications, automate operations, and analyze data. Python is a general-purpose programming language that may be used to develop a wide range of programs and is not specialized for any particular problem. It has become one of the most widely used programming languages due to its versatility and beginner friendliness. It was the second-most popular programming language among developers in 2021, according to a survey performed by industry analyst firm RedMonk.

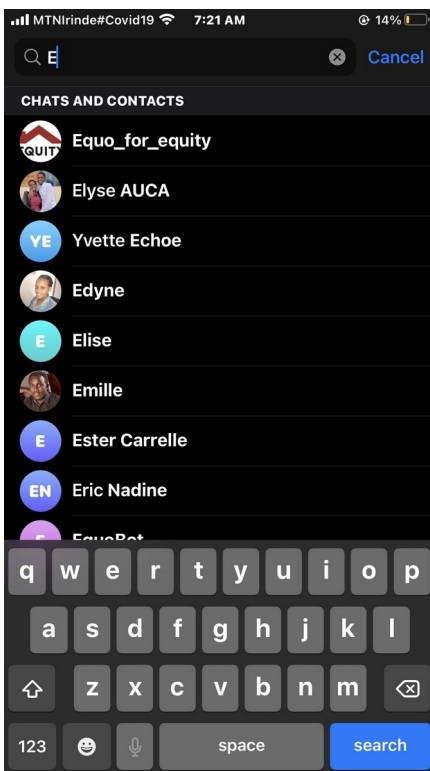## Graphical User Interface of banking-bot system
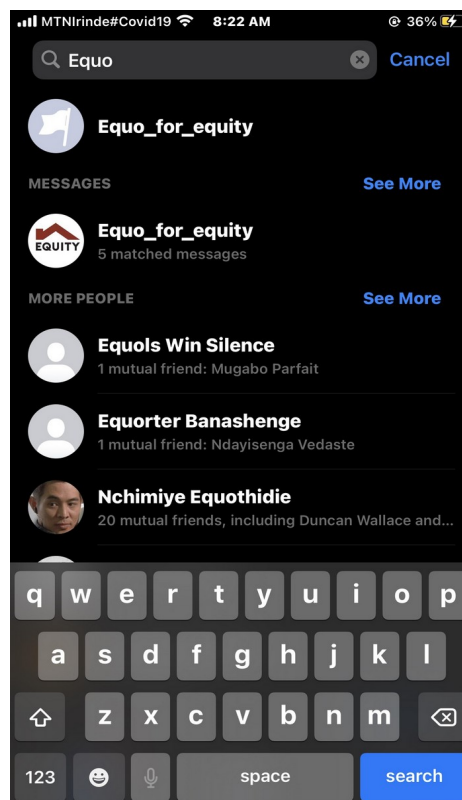

Figure 18: **Telegram search**

Figure 19: **Messenger search**

Images above shows the Telegram and Facebook messenger interface where you simply type "**Equo_for_equity"** in search bar and after the bot shows up you click on it and start the conversation.
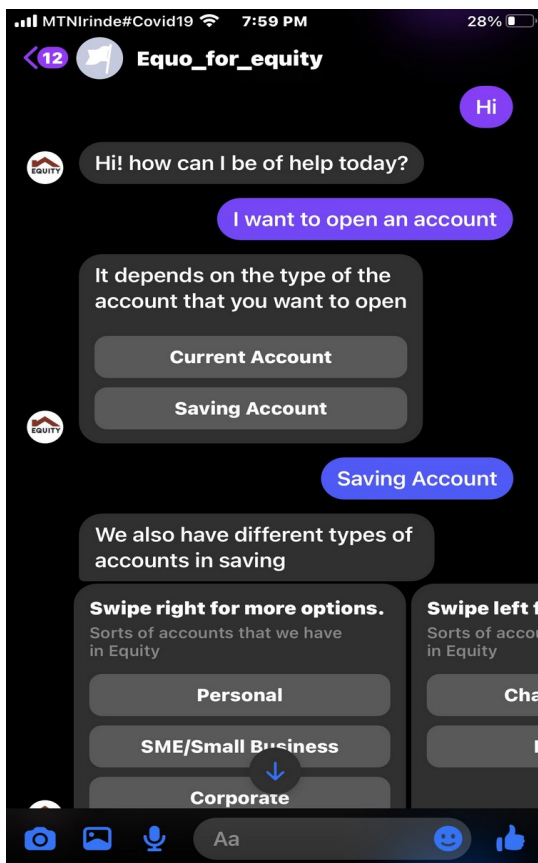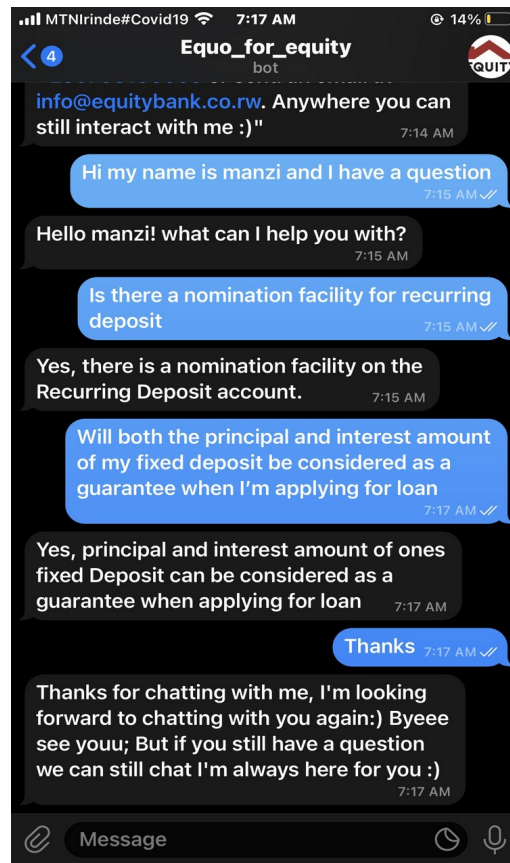
Figure 20: **Messenger chats**

Figure 21: **Telegram chats**

The images above shows how the user chats with the bot, where by he/she starts with a greeting message as he/she could greet the customer care person at the bank, he/she might include his/her name in the greeting message and the bot in it's reply it has to echo the name that the user mentioned. And then the conversation keeps going; a user asks whatever the question he/she has and the bot reply accordingly for both Telegram and Facebook messenger.

The bot has quick replies in the form of buttons where a user can click on one of the responses and get the reply without typing a lot of things. And once the conversation is done the bot thank you and tells you to feel free to ask another question if you have one.
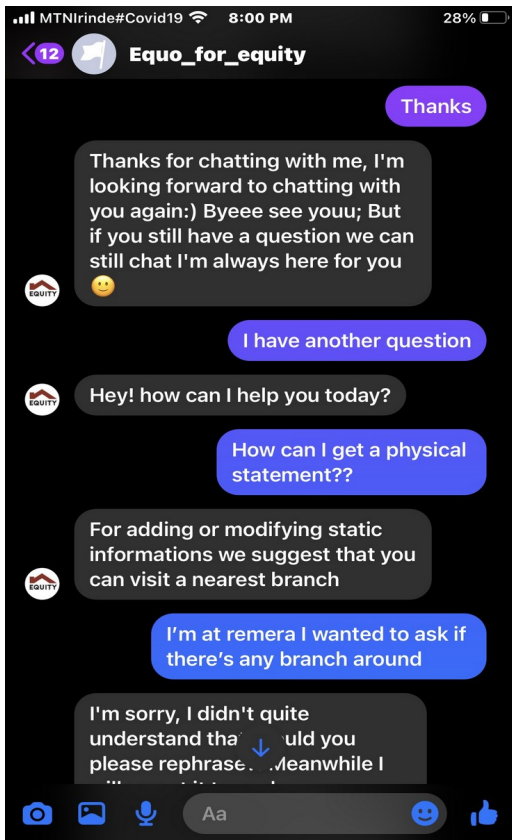
Figure 23: **Conversation continues**
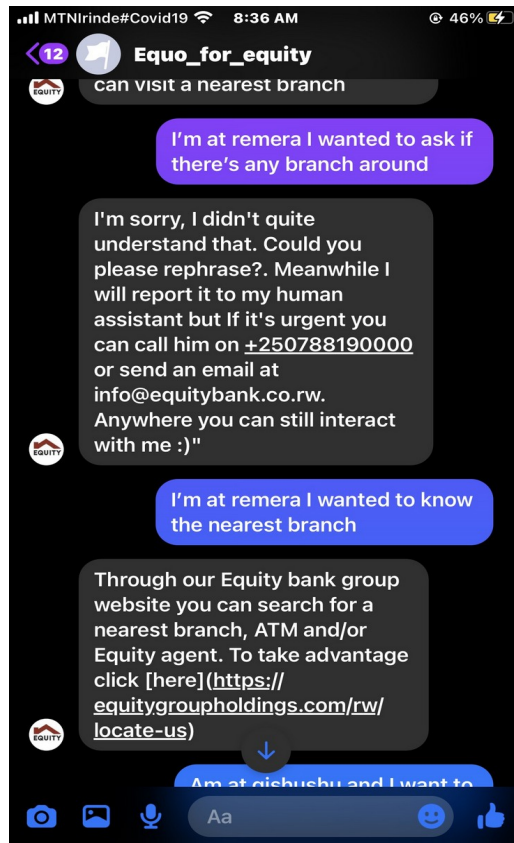
Figure 22: **Bot Fallback**

Images above show how the bot behave in different situations. In case there's another question that you want to ask the bot; Equo_for_equity is flexible enough to allow you to ask as many questions as you want. And also in case there's a fallback; the bot fails to understand what the user is trying to ask, It asks you to rephrase the question for it to understand.

Figure 24: **Equo_for_equity Facebook page**

Above image shows the bot's page on Facebook, You can also chat with it from there.

Figure 25: **Admin's View**

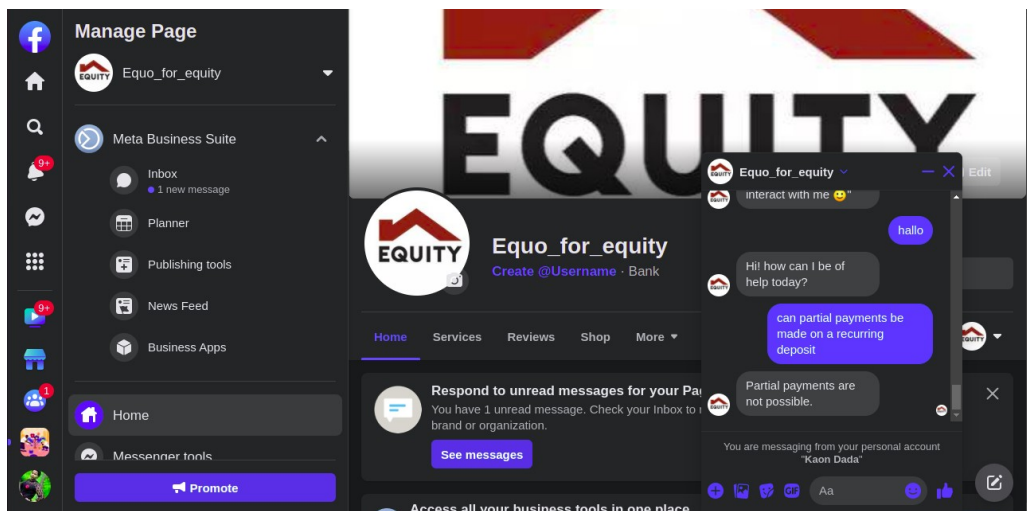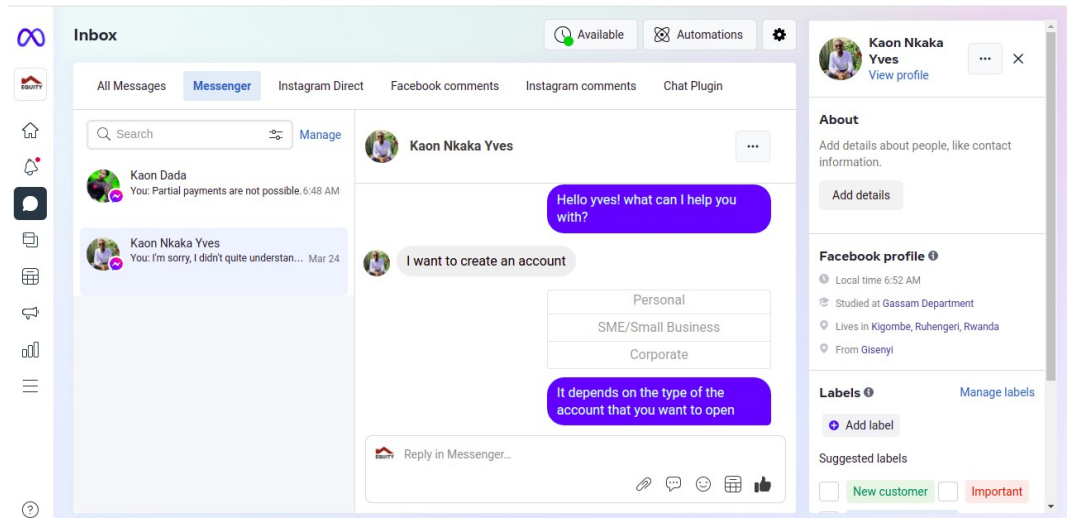This image shows Equo_for_equity admin's view , where by the admin can view all the conversations that the bot had with different users and from there He/She can update the bot's performance.

Software Testing

Software testing is the paramount step in software development. If software is launched without being tested, it can lead to catastrophic accidents. As the project is a bit wide reaching, at all times we need testing to make it successful. If each components work accurately in all respect and gives rightful output for all kind of inputs then project is said to be successful. So to make the project successful, it needs to be tested. The testing done here was System Testing checking whether the user requirements were fulfilled.

Testing Levels

The system was developed and tested in phases. The different units developed required individual testing in order to reveal the errors in them easily. The fundamental testing levels are;
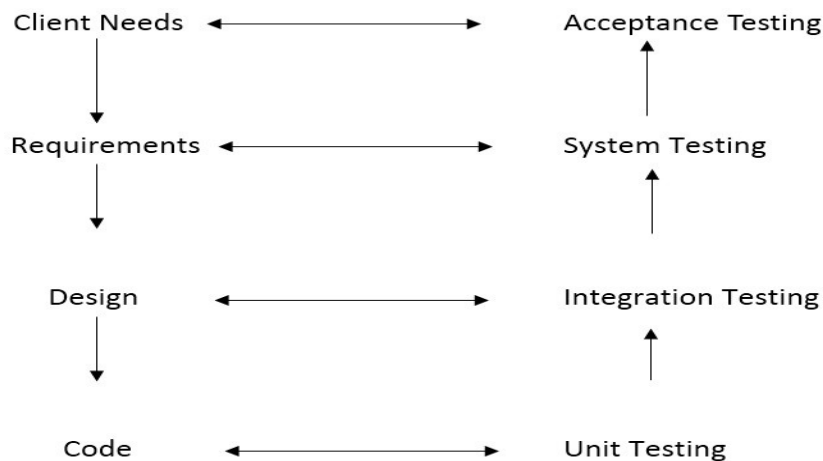
The following are some software testing

User Acceptance Testing

When a software system is developed, the users have to either accept it or reject it. When they reject it, it means the requirements were wrong or the developers did not get enough requirements. To ensure that the system conforms to the requirements, the clients are involved at all development stages by getting their feedback.

System Testing

In this section, various tests were carried out to prove the functionality of the system. Some parameters used in system testing are application and service testing.

Integration test: is the stage in software testing in which individual software modules are combined and tested as a group. This test is useful to check the assembly of the different part of the software. It is also a progression of tests, in which the software and hardware components are collected and tested until the entire system is tested. The application modules have been successively tested until completion to ensure that the whole constituted by the assembled software components answers to the required functional and technical specifications.

Validation test: The last test phase has the role of corroborating the software in its external environment. The product has been put in final situation in order to corroborate if it perfectly answers to the needs expressed in the first phase. The validation test is important, since it is necessary to corroborate if the setting up of the application corresponds to the expressed needs. The application has been tested in its entirety, and it is in this way that i noticed that the progress of operations done corresponds to the functional specifications.

42

Unit Test: Unit testing is a process to guarantee the proper functioning of particular software or a portion of a program. It is a method by which individual units of source code, sets of one or more computer program modules together with associated control data, usage procedures, and operating procedures, are tested to determine if they are fit for use. In other words every small component that can be compiled with the goal to know that every unit matches to its specifications, and to know if there are logical mistakes. Indeed, the unit test is an efficient means that permits to detect the maximum possible mistakes. The application has been checked with the unit test at each piece of the code written.

System Test and Performance

Finally, the whole classification system was tested by going over the various functionalities and processes in the system over and over again For example the reliability of the bot was tested by attempting to insert queries which are not related to bank, sending emojis, sending images and voice notes the bot reply correctly with a fallback message that asks the user to rephase the question for it to understand. The system was run in different messaging platform, The Messaging platform used are Facebook Messenger and Internet Telegram. The results show that the system runs faster and well in Facebook Messenger than other messaging platforms.

Software and Hardware Compatibility Requirements

Client-side requirements:

✓ CPU: Single Core 2.4 GHZ/ Duo Core or higher Recommeded

✓ RAM: 512 MB/1 GB Recommended

✓ Operating System: Windows XP or Higher Recommended

✓ Hard Drive: 5 Gigabytes/6 Gigabytes or more Recommeded

✓ Network: Broadband Recommended

Server side requirements:

✓ Operating System (Ubuntu 18.04/20.04)

✓ Network card : 1GB/second;

✓ RAM : 4GB (Recommended 8GB+)

✓ Hard disk space : 20GB/50GB Recommended free disk space

✓ CPU: 2CPUs (4CPUs+ Recommended)

✓ Python 3.8.10, Rasa 3.0.8, ngrok 2.3.40

CHAPTER 5

CONCLUSION AND RECOMMENDATION

Conclusion

We are at the end of the research project Banking-bot. The main objective of the current project is to provide solutions to problems faced by bank's customers when they need help.

After analyzing how the process of getting help from the customer care were done at Equity Bank and how customers are being managed, I developed this application to make it easy for the customers to access their bank's information. This application also will help the bank to improve their customer satisfaction.

To validate this project, dissimilar methods, technologies and tools have been used. UML has been used for the analysis and design of the new system; observation, documentation and interviews have been used as tools for a good understanding of the existing system and analyzing problem statement to meet the requirements. The conception of this new system indeed contributed to the improvement of the customer satisfaction at Equity Bank.

Finally, I remain available and humble to acquire contributions and to meet demands that could be sent to me to partake to the perfection of this work.

Recommendations

This work was focused on the way how the customers asks queries at EQUITY BANK.

I would recommend EQUITY BANK to take into consideration my project because I have analyzed every process that took place when their customer needs help and I analyzed all problems regarding that process. In the long run if I get to have financial support I would create a physical robot that could be placed at the bank with the functionalities of this bot and some additional features, for instance; how to fill a create account form and others, etc. And I recommend also to EQUITY BANK to use this system and see how the application will be used and what it does.

Finally, I encourage and welcome all contributions from whoever that can improve this work.

REFERENCES

Books

Dennis, A., Wixom, B. H., & Roth, R. M. (2012). Systems analysis and design (5th ed.). Don Fowley.

Wasson, C. S. (2006). *System Analysis, Design, and Development: Concepts, Principles, and Practices* (1st ed.). Wiley-Interscience.

Rumbaugh, J., Jacobson, I., & Booch, G. (2004). *The Unified Modeling Language Reference Manual Second Edition* (2nd ed.). Wesley.

Alan Dennis, Barbara Halley Wixon, David Tegardern (2015) System Analysis and Design An object oriented approach with UML Wiley.

D. Booth, J. D. B. (2018). *Natural language processing.* Joseph D. Booth.

Følstad, A., Araujo, T., Papadopoulos, S., Law, E. L., Luger, E., Goodwin, M., & Brandtzaeg, P. B. (2022). *Chatbot Research and Design: 5th International Workshop, CONVERSATIONS 2021, Virtual Event, November 23–24, 2021, Revised Selected Papers (Lecture Notes in Computer Science)* (1st ed. 2022 ed.). Springer.

del Sole, A. D. (2017). *Visual Studio Code Succinctly.* Van Haren Publishing.

Lutz, M. (2013). *Learning Python, 5th Edition* (Fifth ed.). O'Reilly Media.

Websites

Rwanda population (LIVE). (2020). Worldometer. Retrieved March 15, 2022, from https://www.worldometers.info/world-population/rwanda-population/#:~:text=The%20current %20population%20of%20Rwanda,the%20latest%20United%20Nations%20data.

Access to Finance Rwanda (AFR). (2020). Finiscope financial inclusion. https://www.bnr.rw/fileadmin/user_upload/2020_Rwanda_Finscope.pdf

N.I.S.R. (2022, March 5). *RPHC4: Population Projections | National Institute of Statistics Rwanda*. National Institute of Statistics Rwanda. Retrieved March 5, 2022, from https://www.statistics.gov.rw/publication/rphc4-population-projections

Equity. (2022). Equity Group Holdings. Retrieved March 28, 2022, from https://equitygroupholdings.com/rw/about-equity

IBM Cloud Education. (2020, July 2). *Natural language processing (NLP)*. Ibm.Com. Retrieved April 15, 2022, from https://www.ibm.com/cloud/learn/natural-language-processing

V., V., Cooper, J. B., & J., R. L. (2020). Algorithm Inspection for Chatbot Performance Evaluation. *Procedia Computer Science, 171*, 2267–2274. https://doi.org/10.1016/j.procs.2020.04.245

Rasa, R. (2022). *Tuning Your NLU Model*. Rasa. Retrieved May 2, 2022, from https://rasa.com/docs/rasa/tuning-your-model/

APPENDICES