



Polytechnic Institute of Leiria

MASTER IN COMPUTER ENGINEERING – MOBILE
COMPUTING

DISSERTATION

*A conversational agent to assist users in public
institutions of Ecuador*

Esteban Eduardo Cando Penaherrera

November 2020



Instituto Politécnico de Leiria

MASTER IN COMPUTER ENGINEERING – MOBILE
COMPUTING

DISSERTATION

*An intelligent conversational agent to assist users in
public institutions of Ecuador*

Esteban Eduardo Cando Penaherrera

Disseration developed under the supervision of
Professor Carlos Grilo and Professor José Ribeiro and
Professor Maria Lapina

Originality and Copyright

This project report is original, made only for this purpose, and all authors whose studies and publications were used to complete it are duly acknowledged. Partial reproduction of this document is authorized, provided that the Author is explicitly mentioned, as well as the study cycle, i.e., Master degree in Computer Engineering – Mobile Computing, 2017/2019 academic year, of the School of Technology and Management of the Polytechnic Institute of Leiria, and the date of the public presentation of this work.

Acknowledgements

Escribo estas líneas para agradecer a todos los que me han ayudado a culminar este proyecto:

A mis profesores y tutores en Portugal, que siempre estuvieron prestos a ayudarme sin importar la situación, la hora, o el día.

A mi familia que siempre ha creído en mí, incluso cuando les he fallado.

A mis amigos que siempre han estado en el momento y lugar indicado, en especial cuando me he derrumbado.

A Male, que me ayudo a encontrar el impulso necesario poder finalizar mis metas, incluso cuando ya no era necesario. Tal vez se lo agradezca en la próxima vida, cuando seamos gatos.

A mis hermanos, a los que quiero entregar un buen ejemplo tal y como yo lo he recibido.

A mi Padre que nunca dejo de apoyarme, incluso cuando él también necesitaba apoyo por todas las adversidades que ha enfrentado

Y a mi Madre que desde que nací, y hasta que cerro sus ojos por última vez, nunca dejo de quererme y enseñarme a ser valiente incluso ante la muerte. La extraño mucho, y cada día un poco más. Cambiaria todo por tenerla un instante y poder volver a decirle cuanto la amo. Desearía tanto poder celebrar esta victoria con ella, pero esto seguro que se fue sabiendo que lo lograría.

Abstract

This project described in this dissertation is being carried out for the Ministry of Social and Economic Inclusion (MIES) of Ecuador based on its current problems. The deficiencies that government entities have to help their citizens with information and paperwork has exceeded their operational capacity. The MIES currently has a Marketing and Information department that is saturated with questions through channels such as call centers, information centers in their offices and social networks outside their office hours.

In recent years there has been an increase in requests for information through digital channels such as its Facebook page, since internet access to places far from the big cities has increased. This has been seen as an opportunity for improvement by creating a chatbot agent that helps citizens 24 hours a day, 7 days a week. Most of the queries can be made through a query to their internal data and this allows the operational burden of the operators to be released to carry out management processes and not just information.

Many public and private companies have resorted to using chatbots to help their users with simple information tasks. They have also relied on these technological tools to create registration and management functions. Today we can use chatbots development tools and platforms such as Dialogflow that allow to create chatbots in a manageable and scalable way according to needs. Facebook has multiple connection tools that allow it to integrate safely and efficiently with its platforms such as Facebook Messenger, allowing customers to create applications that will reach the majority of citizens who have internet access.

Developing a chatbot agent can be a simple and straightforward task, but in which the remaining time must be invested in training and helping the chatbot agent to understand words or idioms of language used in different regions of the same country. In the development of a chatbot, the training cycle is constant and allows the chatbot to increase its ability to understand the users who use them.

In the tests carried out on this chatbot, it was concluded that although the use of these tools allows to release the operational load, it does not avoid the need for users to interact with natural persons, but the requirement is drastically reduced. In the tests it was also obtained that many problems can be obtained by finding a wide range of synonyms and local expressions, but correct training and constant review of the answers provided helps the chatbot to self-train and evolve its answers over time interactions.

Keywords: : Chatbot, Dialogflow, Facebook Developers, Natural Language Processing.

Contents

1	Introduction	12
1.1	Motivation	13
1.2	Goals and Contributions	15
1.3	Document Structure	16
2	Background	17
2.1	Natural Language Processing	18
2.2	Turing test	19
2.3	Bot and Chatbot	19
3	State of Art	21
3.1	Human-Computer Interaction	21
3.2	Evolution of chatbots	22
3.3	Applications	29
3.3.1	Medicine	29
3.3.2	Customer service	29
3.3.3	Learning	30
4	Development Methodology	32
4.1	Solution Methodology	32
4.1.1	Design Thinking	32
4.1.2	Behavior Driven Development	34
5	Solution architecture	37
5.1	User Requirements	37
5.1.1	Functional Requirements	37
5.1.2	Non Functional Requirements	39
5.2	Architecture Solution	40
5.2.1	Communication Architecture	40
5.3	Technologies Solution	41
5.3.1	Frameworks and tools to develop chatbots	41
5.3.2	Facebook developer tools	41
5.3.3	Facebook Messenger Platform	42
5.3.4	Dialogflow plataform	42
5.3.5	Basic concepts	44
6	Development of the proposal	45
6.1	Configuration on Dialogflow	45
6.1.1	Entities	46
6.1.2	Intents	48
6.1.3	Actions and parameters	51
6.2	WebHooks and Connections	55
6.3	Integration with Facebook Developers	57
6.3.1	Access Tokens	58

6.3.2	Analytics	63
6.4	Agent Training	64
7	Agent evaluation	65
7.1	Analytic Testing	65
7.2	Usability Testing	69
7.3	Results	74
8	Conclusion and future work	85
8.1	Future Work	87
Appendices		
A	Tables of Requirements	91
B	Usability Test Form	96
C	Usability Test Results	103

List of Figures

1	Human computer interaction related areas	21
2	Eliza's screen example	23
3	Architecture of TINYMUD	23
4	Home screen for ALICE Chat	24
5	Clippy	25
6	Siri on an Iphone	26
7	Watson Servers on IBM presentation	27
8	Cortana on Android	28
9	Amazon echo with Alexa assistant	28
10	Medical Chatbot	29
11	Customer service chatbot	30
12	Design Thinking Methodology	33
13	Design Thinking Methodology cycles.	34
14	BDD User Story Example	36
15	Architecture proposal	40
16	Basic Configuration on Dialogflow	45
17	Entities of conversational agent	47
18	Entities with regular expressions	47
19	Entities with Fuzzy matching	48
20	Intents of the Agent	49
21	Context configuration of an Intent	50
22	Events configuration of an Intent	50
23	Training Phases of an Intent	51
24	Action and parameters of an Intent	52
25	Response's of an Intent	53
26	Structure of responses	54
27	Facebook Apps configuration	58
28	Basic Configuration on Facebook App	58
29	Subscriptions services from Facebook Messenger	59
30	Basic Configuration on Facebook App	60
31	Basic Configuration on Facebook App	61
32	Start call from Facebook Messenger	62
33	Response test from Facebook Messenger	62
34	Statistics from Facebook App	63
35	Test from Facebook Messenger	65
36	Test from Dialogflow console	66
37	Calls from Facebook Messenger	67
38	Mistakes from Facebook Messenger	68
39	Average request time from Facebook Messenger	68
40	Dialogflow Statistics	69
41	Usability test form example	71
42	Survey statistic: Age	74
43	Survey statistic: Residence	75
44	Survey statistic: Education level	75

45	Survey statistic: Facebook as a query tool	76
46	Survey statistic: Human development bonus	77
47	Survey statistic: Collection places	79
48	Survey statistic: Other programs	81
49	Survey statistic: recommendations	83
50	Survey statistic: Agent Qualification	84

List of Tables

1	Facebook services to suscribe	43
2	Requirements: Feature 1, scene 1.1	91
3	Requirements: Feature 1, scene 1.2	91
4	Requirements: Feature 1, scene 1.3	91
5	Requirements: Feature 2, scene 2.1	92
6	Requirements: Feature 2, scene 2.2	92
7	Requirements: Feature 2, scene 2.3	92
8	Requirements: Feature 3, scene 3.1	92
9	Requirements: Feature 3, scene 3.2	93
10	Requirements: Feature 3, scene 3.2	93
11	Requirements: Feature 4, scene 4.1	93
12	Requirements: Feature 4, scene 4.2	93
13	Requirements: Feature 4, scene 4.3	94
14	Requirements: Feature 5, scene 5.1	94
15	Requirements: Feature 5, scene 5.2	94
16	Requirements: Feature 6, scene 6.1	94
17	Requirements: Feature 6, scene 6.2	94
18	Requirements: Feature 7, scene 7.1	95
19	Requirements: Feature 8, scene 8.1	95
20	Requirements: Feature 8, scene 8.1	95

List of Acronyms

API Application programming interface. 41

BDD Behavior Driven Development. 34

DNI National Identity document. 48–50

FBML Facebook Markup Language. 42

HCI Human-Computer Interaction. 21, 22

MIES Ministerio de Inclusión Económica y Social. 13, 46, 48, 55, 70

NLP Natural Language Processing. 15, 16, 18, 39, 40

NLT Natural Language Toolkit. 18

1 Introduction

Fifty-four years ago, ELIZA born at MIT Artificial Intelligence Laboratory. She was created to demonstrate the superficiality of communication between humans and machines. She is considered the first piece of conversational software. Since then, conversational agents have evolved and in the last years have become part of our daily lives mainly thanks to smartphones. Meanwhile, society has changed, as well as the way we communicate. Globalization produced changes in the world along with technology. More and more tasks, once reserved for humans, are played by computers and robots today. All this giving rise to new ways of doing business and revolutionizing marketing and sales strategies with the user or client.

Nowadays, the standard ways of interacting with clients are through social networks, but it comes with more work for human resources because people usually use them out of the working hours. To solve this problem, big companies like Domino's Pizza, KLM Royal Dutch Airlines, Starbucks or National Geographic implemented on their web pages conversational agents to facilitate contact with their customers.

A conversational agent is a tool capable of processing natural language and that can hold a conversation with the user, whether he/she asks for some information or asks for some action to be carried out. Artificial intelligent chatbots are used in different scenarios such as banking systems, customer services or education, but mainly they have focused on customer services [20]. Most of the questions or queries made on their fan pages or social networks are repeated again and again, turning the response into a monotonous and repetitive action that can be solved by a community manager.

Companies choose to hire staff as community managers to deliver a faster and more accurate response that responds quickly to queries and comments from users on social networks. More and more consumers are using networks to get in touch with brands. Users know that a smart company will provide a timely and transparent response through networks; after all, they are on a platform where all the world is watching. However, a company with twelve million users like KLM, needs a big staff to respond to their users, as is the case of governments and the new ways that they communicate with the people [20].

According to the 2020 census, Ecuador has 13.8 million connected users, with an average 79% Internet penetration in the population. This volume is justified by the number of active cell lines and the government initiatives for access in public areas and the offer of low-cost mobile service providers. The Ecuadorian state has begun to create and provide information on their web pages, taking advantage of this momentum as a new information channel. And, over the years, these channels have migrated to social networks like Twitter, Instagram and Facebook. For example, the Ministry of Economic and Social Inclusion (MIES)

MIES has approximately 200,000 followers on Facebook and around 5,000 followers on Twitter [4].

MIES is in charge of carrying out projects to help the neediest population such as single mothers, elderly people, people with serious illnesses, etc. For the most part, the consultations they make to their community managers are oriented to information about new projects, the requirements to access those projects, the collection sites and the questions are usually the same.

To carry out any type of procedures, inscriptions, inquiries or request for documents, citizens should approach the office closest to their home. This is not just a mobility and time issue. If we consider that the majority of users who use the benefits of the ministry are people with low economic resources, and for whom any help they can obtain is significant, avoiding costs of mobility, food and work time is a priority.

Since most of the users' needs are information requirements or help when solving their doubts, today the ministry manages an information department in the central offices which also has a call center system. This has generated a lot of expectation and reception in the citizens, but has increased the need for help personnel. Each of the 57 offices has at least one information services person and the call center team is made up of 45 people and increasing. In total, an average of 20000 calls are answered through the call center, from Monday to Friday from 7:00 a.m. to 9:00 p.m.

Social networks have become a direct channel for the population to be closer to their authorities, and also to their public institutions. Taking advantage of this large number of connected people and the global health situation promulgated by social distancing due to the global pandemic of Covid 19, we have sought an effective solution using chatbots that help the population with information without having to leave home or go to a public institution.

The implementation of chatbots aims to help the information and support department in the need to hire more personnel to help answer the concerns of citizens. Also reduce the number of necessary personnel in the Call Center, and be ready to help citizens at all hours, every day of the year.

1.1 Motivation

This project was developed to improve the communication processes between customers and the communication department of the Ministry of Economic and Social Inclusion. The current problem in this area is misuse of the communication channel in social networks because it is very slow. The project is aimed to improve the response times to messages sent through social networks, as well as a better use of the human resources by throwing repetitive and informative tasks to an intelligent agent who can respond in a natural and user-friendly

language.

Now, the entire process is done manually: after the user contacts for information, he/she is directed to an operator. The mechanics are repetitive because the information does not change very much over time. This way of working diminishes the performance of the communication division, who wastes time informing the large number of users who look for answers to simple questions or just want to know more about the programs like: location, requirements, projects which they can apply to, available times, etc. They waste human resources for those simple questions instead of using them to communicate programs and help to potential users who have already decided to help. Also, the given information can change from operator to operator, making this a problem.

With a properly trained agent, we can add an essential resource to the information cycle process, since most of the concerns made through messages are from people interested in information. The agent can also perform office functions as validate requirements, promote new projects, or any interactions that are necessary for the core of the business. Implementing an intelligent agent to be part of the team and be part of the communication work will allow them to obtain some advantages such as: a conversational interface, asynchronous conversations or marketing tools.

Unlike creating a mobile app from scratch, the conversational interface of Facebook is one of the most used in the world. Besides, this tool generates a new concept of versatility associated with the brand when the user uses it from his/her cell phone or personal computer. This allows keeping the conversation where it was left if it was interrupted.

Marketing departments are the first to look for new ways to improve and optimize investments, but at the same time, new users may exceed the ability of salespeople to react. A conversational agent arrives to help the operator in charge of social networks or any other communication channel such as a website or email, 24 hours per day, seven days week, even on holidays. The advantages that a conversational agent can provide to the company are vast, varied and continuous over time since the chatbot can evolve. The chatbot may start by giving automatic answers to 30% or 50% of the questions until it can reach or exceed 80% of all queries [7].

1.2 Goals and Contributions

The main objectives of the chatbot are to answer the question if users are beneficiaries of "Bono de Desarrollo Humano", using the citizenship ID of Ecuadorian users as a means of identification. The Agent should also provide information of collection places near the users home, information about other bonus and ministry projects, business hours, other procedures that can be done online and information relevant to Covid 19. All this information should be provided using the Facebook page of the Ministry of Economic and Social (MIES).

The primary goal of this project is to implement a chatbot agent who will be integrated into a social network and solve the questions of the users in natural language, to improve the performance of the business in communication channels and speed up the response times and interactions with users, allowing to evaluate the performance of the agent. In short, the objectives of this work are the following:

- Analyze and compare the tools provided in the current technological market for the development of conversational agents;
- Analyze the structure, the architecture, tools and the design of conversational agents.
- Implement a conversational agent using NLP technology, and integrate it into the Facebook Messenger module;
- Encode and generate the knowledge base that will nurture the responses generated by the conversational agent;
- Publish the agent prototype in a production environment, to analyze its behavior in a real environment;
- Generate a final analysis with the problems found at the time of the investigation and ways of solving them.
- Improve the response processes to doubts and concerns of the users using the conclusions taken from the state of the art.

1.3 Document Structure

This document, in addition to the chapter "Introduction," is organized in the six following chapters:

- Chapter "Background" - This chapter includes general information on topics related to chatbots and NLP, to put the research in context;
- Chapter "State of the Art" - In this chapter, modules are presented to understand a human-machine communication system, the history of chatbots and their evolution, natural Language Processing and how we can train the agents. Also, some useful information about Facebook API, and Dialogflow Tools;
- Chapter "Project proposal" – This chapter analyzes the proposal for the project, the Thinking design methodology, and requirements with the users. In this section, we include the architecture of the project;
- The chapter "Implementing and Testing" - This chapter presents and describes the process of implementing the smart agent and discusses several difficulties encountered during the process;
- Chapter "Discussion of Results" - An analysis is made based on the results obtained;
- Chapter "Conclusion" - This chapter presents the conclusions of the project, the main contributions and some proposals for future work.

2 Background

All sectors, public and private, have evolved over the years. The Ministry of Social Inclusion of Ecuador was created in 1979 under the name of "Ministry of Social Welfare" by President Jaime Roldós Aguilera. Its headquarters are in Quito, the capital of Ecuador, and it has more than fifty offices in most of the main cities of Ecuador. In its early years, there were offices dedicated to the distribution and planning of news related to the projects that were commissioned. These were published on radio, press and television for dissemination among Ecuadorians. Over the years, small Call Centers were invested in, which were in charge of managing direct communication. By the year 2000, with the arrival of the internet and the technological advances of the new millennium, a series of government pages was established that allowed users to know the information of each of their constituents, from the comfort of their homes. The biggest problem was that not all the population of Ecuador had access to the internet, due to its costs and the need for personal computers. However, they have never parted with the past model. Today information is still published through print and audiovisual media thanks to stable channels, but little by little it has begun to digitize and move towards new multimedia channels such as social networks since every day citizens access them more easily. Maintaining old channels such as: help offices, call centers, information department, etc. have increased the workload of the staff. This is because people have more access to telephony and the Internet in the last 5 years. The only way they have found to cope with this massive load of requests and questions has been to bring staff to their workplaces. Although this solution made it possible to solve the problem in the first years, it would be unsustainable in the future. That is why the ministry has started looking for efficient technological alternatives. Smart agents can help people with their concerns, since most questions are similar between users:

- Where are your offices?
- What are your work hours?
- Do I have access to a money bonus?
- Am I a candidate for a program?
- Where can I collect my money?
- What information or documents should I present when I withdraw my money?
- Until when can I withdraw it?

The ministry wants to implement a chatbot agent that helps staff release repetitive workloads, without the need to hire more outside staff. An efficient and effective solution to this problem is undoubtedly a chatbot agent. Smart agents have also undergone a process of improvement over the years. Today we can find this novelty of chatbots in multiple sectors but they are not as new

as most people think. There are two perspectives on the provenance of these chatbots: the first one, holds that chatbots began with the work of the mathematician Alan Turing in the 1950s when he began to develop the first algorithms to make modern computers and artificial intelligence work. He has investigated whether a machine was able to mimic a human's responses by studying a text conversation between a machine and a human.

The second perspective places the origin in the year 1964 with Joseph Weizenbaum, who created the first to be considered a chatbot, Eliza, designed to act as a psychologist / therapist. Through this method, he was trying to demonstrate the superficiality of communication between man and a machines. These investigations fell on a main idea. The algorithms of the time were not implemented to understand natural language, which is the medium through which people communicate [5].

2.1 Natural Language Processing

NLP is the capability of a technological system to understand human language communications. The development of NLP applications is challenging because traditional systems require humans to communicate to them in a specific language that is explicit, unambiguous and highly structured, or through a limited number of clearly enunciated commands. Human speech however, is not always precise. It is often ambiguous, and the linguistic structure can depend on many complex variables, including slang, regional dialects, and context [3].

The history of natural language processing started in 1950. Alan Turing published an article titled "Intelligence" which proposed what is now called the Turing test as a criterion of intelligence. Nowadays, NLP algorithms are based on machine learning algorithms. Instead of hand-coding large sets of rules, NLP can rely on machine learning to automatically learn these rules by a special training using a knowledge base and making inferences. With more data analyzed, the model will be more accurate.

Several libraries can provide algorithmic building block for NLP in real-world applications. We have, for example Apache OpenNLP, which has a machine learning toolkit that provides tokens, sentence segmentation, part-of-speech tagging, named entity extraction, chunking, parsing, co reference resolution. Natural Language Toolkit NLT is a Python library that provides modules for processing text, classifying, tokenizing, stemming, tagging, parsing. Stanford NLP is a suite of NLP tools that provide part-of-speech tagging, the named entity recognizer, co reference resolution system, and sentiment analysis. In Java, we have MALLET who is a Java package that provides Latent Dirichlet Allocation, document classification, clustering, topic modeling, information extraction, and more [3] [12] .

2.2 Turing test

Alan Turing, in a 1951 paper, proposed a test called "The Imitation Game" that might finally settle the issue of machine intelligence. The first version of the game involved no computer intelligence. This first version of the game a man, a woman and a judge must participate in identifying which person is talking to him. Turing then proposed a modification of the game, in which instead of a man and a woman as contestants, there was a human, of either gender and a computer at the other terminal. Now the judge's job is to decide which of the contestants is human, and which is the machine. Later, the game was modified so that there is only one contestant, and the judge's job is not to choose between two contestants, but merely to decide whether the single contestant is human or machine [15].

The Turing Test has been criticized over the years in the science community because the nature of the questioning had to be limited in order to exhibit human-like intelligence. For many years, a computer might only score high if the questioner formulated the questions to be a response "Yes" or "No." When questions were open-ended and required conversational answers, it was less likely that the computer program could successfully fool the questioner, especially if the dialogue goes for an extended time.

2.3 Bot and Chatbot

According to the definition, a Bot is "A computer program that performs automatic repetitive tasks or assignments" [8] To expand on this definition, bots also act as the primary tool for automating communications and engagement with website content on a large scale. Like all the machines, bots can certainly be misused and all the people need to know and prevent this. There are a lot of cases of internet attacks, social media accounts, spam, and more [8]. These "botnets" tend to give bots a bad reputation and inspire negative connotations.

A chatbot is a type of bot designed to interact with humans conversationally, based on its programming. They automate the process of interacting with website visitors and social media followers to create the best user experience. There are a group of points to take care of so that a bot can be considered a chatbot. Chatbot needs an understanding of language. If it does not understand language correctly, then it is just a Bot with rules. Chatbot has a certain knowledge about context and supposed to be an intelligent system to build proper interactions with data taken from the user, industry, context, etc. Chatbots need to interact with the user or with other chatbots. If there is no ability to interact, then it is not a chatbot [6].

Chatbots can be classified according to their nature. For example, we have agents according to their text-type interaction, in which they interact by sending and receiving characters in a particular language. They also interact using buttons or images, which allow a tree-like action. Alternatively, they can also interact with voice commands [16]. According to its application, there are operational agents, who carry out actions or calls to specific functions such as

giving shifts or assigning officials. In the case of customer support, agents stand out in FAQ's, smart forms, or virtual assistants. According to the technology implemented in their responses, there are, for example, decision trees, artificial intelligence, machine learning, or knowledge base [6].

3 State of Art

In this chapter we are going to review a little about the history and technological evolution that chatbots have had in these last decades and how they have become from computer programs, to be present in our smartphones. Chatbots have become part of our daily lives that we can see in schools, medical centers, airports and in many other places that need information support 24 hours a day.

3.1 Human-Computer Interaction

The term human-computer interaction or HCI appeared in the mid-80s to describe a new field of study: a set of processes, dialogues, and actions which a user uses to interact with the computer. Nowadays, that definition should be broader: It is the discipline that studies the design, evaluation, and implementation of interactive computer systems for human use and the phenomena that surround it, in which the areas of computing, design and psychology are intertwined [10].

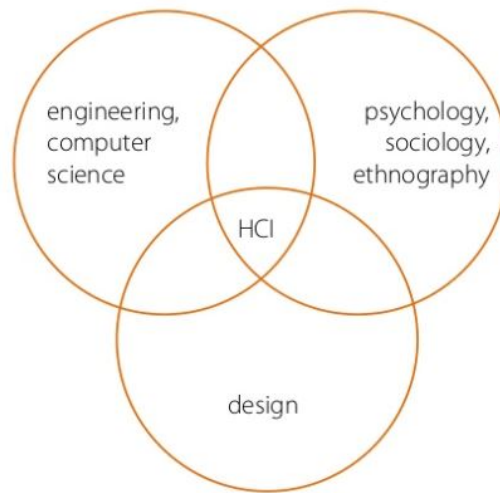


Figure 1: Human computer interaction related areas

The primary objective of HCI is to improve the quality of computer systems through three factors: ease of use, efficiency, and safety. In this context, the system refers not only to hardware and software but also to the entire work environment. An essential idea in HCI is the belief that the user comes first, and the system must be adapted to the needs of the user and not the other way. That is why engineering has seen it necessary to work hand in hand with design and areas such as sociology, to allow HCI that fulfill their main objective

(Figure 1).

To this emerges a general feeling for an active communication device or channel, and there is a discussion about the convenience of an interface in the language of the person who uses it and who is capable of making the computer understand him/her simply and effectively. Computer scientists cooperate in this discussion asking for a language highly stenographic for his concepts of computer management and information [10].

Microsoft CEO Satya Nadella made statements in which he detailed the planned transition to chatbots and natural language user interfaces with previous revolutions, such as the introduction of the graphical user interface, the Web and the mobile Internet. Facebook CEO Mark Zuckerberg proclaimed that chatbots are a solution to the challenge of application overload. If the vision of conversational interfaces is true, significant changes are expected for the HCI field.

3.2 Evolution of chatbots

Intelligent Agents have evolved over the years, aided by technology and advances related to artificial intelligence. From the first Agent known as Eliza, to our days with SIRI, we can see the differences in each of them. At first they were just computer programs that answered pre-programmed questions and did not understand the messages they received. Today, instead they can understand the message they are receiving, the context and even the basic feelings of each conversation. This has allowed them to become our daily assistants, which are in the palm of our hand thanks to our smartphone. Let us detail some of the most famous and well-known intelligent agents.

Eliza 1964

The evolution of intelligent agents begins in 1964 with the creation of ELIZA. Joseph Weizenbaum developed this first bot in the MIT laboratories for IBM. It was implemented to respond to and in English language and it was able to dialogue on any subject (Figure 2). Eliza used labels to categorize the information that fed her [17].

```

Welcome to

          EEEEE LL      IIII ZZZZZZ  AAAAA
          EE      LL      II      ZZ  AA  AA
          EEEEE LL      II      ZZZ  AAAAAA
          EE      LL      II      ZZ  AA  AA
          EEEEE LLLLLL IIII ZZZZZZ  AA  AA

Eliza is a mock Rogerian psychotherapist.
The original program was described by Joseph Weizenbaum in 1966.
This implementation by Norbert Landsteiner 2005.

ELIZA: Is something troubling you ?
YOU:   Men are all alike.
ELIZA: What is the connection, do you suppose ?
YOU:   They're always bugging us about something or other.
ELIZA: Can you think of a specific example ?
YOU:   Well, my boyfriend made me come here.
ELIZA: Is it important to you that your boyfriend made you come here ?
YOU:   He says I'm depressed much of the time.
ELIZA: I am sorry to hear that you are depressed.
YOU:   It's true. I am unhappy.
ELIZA: Can you explain what made you unhappy ?
YOU:   █

```

Figure 2: Eliza's screen example ¹

In her conversation methodology, she had the implementation of pattern comparison that generated an illusion of understanding. For this reason, it could even be used for the user to believe that a psychiatrist heard the user. The response algorithms that were implemented to generate a deceptively original text, and although its author emphasized it, many users who used it for the first time were deceived.

Chatterbot 1989

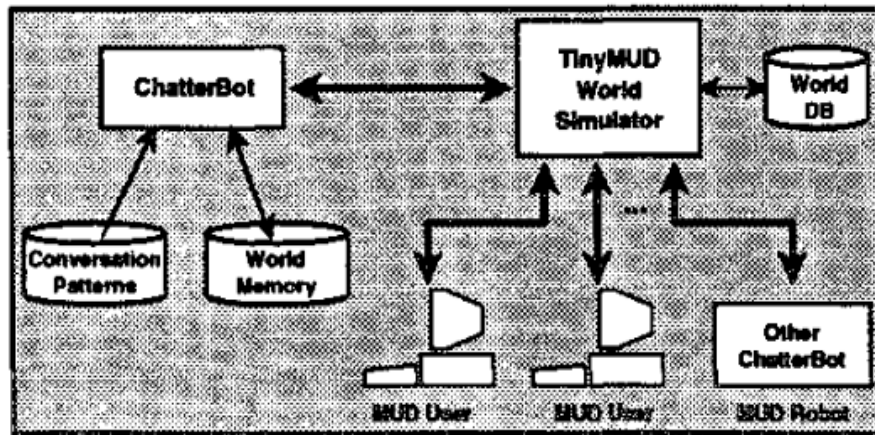


Figure 3: Architecture of TINYMUD ²

¹Image source: <https://medium.com/nlp-chatbot-survey/>

²Image source: <http://www.lazytd.com/lti/pub/aaai94.html>

Chatterbot was a virtual player that was developed for the game “TINY-MUD,” based on ELIZA. The idea was to simulate a real player to talk with another players to solve doubts about the navigation into the virtual world. The success of this Bot was because, at that time, it was not known about the artificial intelligence that the TINYMUD game possessed, allowing human players to assume that all the participants were real [17]. In the architecture of TINY-MUD (Figure 3), the chatbot fed on patterns of conversation and information from the artificial world. The chatbot interacted with other users so that they did not know that he was a chatbot who conversed with them, as if it were just another gamer.

Alice 1995

Artificial Linguistic Internet Computer Entity, known by its acronym as “ALICE,” was also based on ELIZA on its primary engine. She was part of the project called PANDORA, which added to her previous functions to collect examples of natural language on the web, which by 1995 was already beginning to expand.

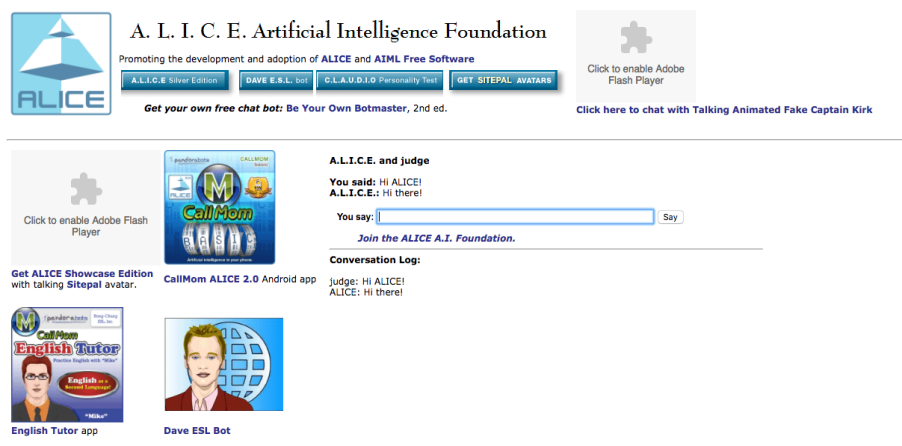


Figure 4: Home screen for ALICE Chat

She was developed by Dr. Richard S. Wallace to pass Turing’s test, which she did not. In 1998 the agent was rewritten in the JAVA language, to use a particular XML schema called AIML or Artificial Intelligence Markup Language. Currently, Alice’s source code is available, and any user can implement their versions of it (Figure 4) [17].

Clippy 1997

Designed by Microsoft, Clippy was the first intelligent personal assistant avail-

able for Windows. Its primary objective was to help users with the real doubts in the tools of Microsoft Office 1997 (Figure 5).

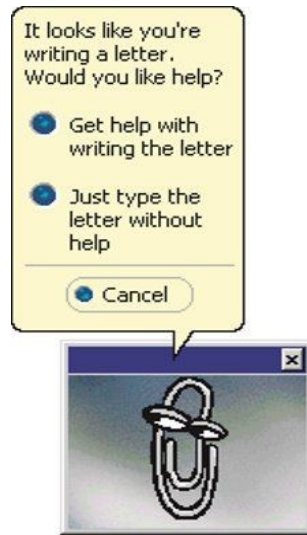


Figure 5: Clippy

One the main features that we can highlight, is the use of triggers to activate events. For example, when some user write the word "Dear," at the beginning of a document, Clippy went out to offer it help to write a letter. Clippy was not accepted by users, who by that time, began to feel observed. It was strange for them that a technological and unknown entity could know their activities [2].

SIRI 2011

Siri was created by Apple to be the virtual assistant of their smartphone, "iPhone". The company Nuance in conjunction with Apple engineers was responsible for creating an agent capable of recognizing voice commands, with amiable qualities. Among the queries that can be answered, we have some about weather, music, contacts, calendar. Also, SIRI can interact with applications like calculator or maps. To promote its use, a side button was implemented that helped access Siri much more quickly (Figure 6).



Figure 6: Siri on an iPhone³

One of the global implementations made it on SIRI were the use of machine learning for all iPhone users to feed the knowledge base to improve their recognition of pronunciation, meanings, language expressions, etc. That makes a big difference with other agents and improves the user experience over the years. Siri was programmed in Objective-C and can recognize more than twenty languages and, among the latest updates, it can be highlighted the contribution that has been received by psychologists to improve the responses that occur [2].

Watson 2011

Developed by IBM (Figure 7), Watson has been the star of the Chatbots in the last decade. It has participated in competitions such as TV program “Jeopardy” in the United States, to demonstrate the cognitive system that feeds it. Watson is part of the DeepQA research project, led by David Ferrucci, and was named after the first IBM president: Tomas J. Watson. Watson can analyze and interpret the proposed data, including unstructured text, images, audio, and video. Also, it can offer recommendations and understand the emotions of the messages sent by the user.

³Image source: <https://www.apple.com/es/siri/>

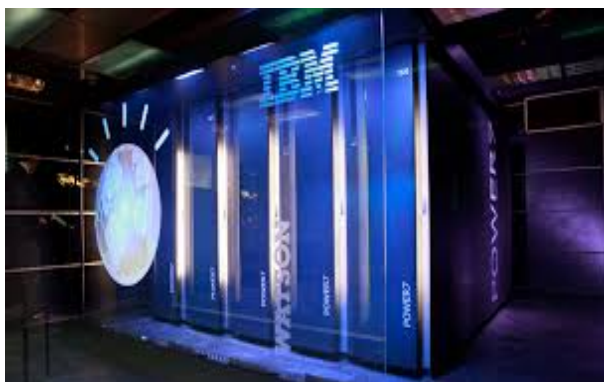


Figure 7: Watson Servers on IBM presentation⁴

The process is simple. Watson decomposes information to create a hypothesis. Then, it performs the first information search and generates possible candidates for the response. These responses are given a score according to different points of view such as mood, reason, user, time, place, date, etc. In the end, a fusion of the equivalent answers is made to determine the definitive answer [17].

Cortana 2014

Cortana is the virtual assistant created for the different Microsoft products such as computers, tablets, and smartphones (Figure 8). She can recognize more than ten different languages. Moreover, with the generation of a lot of API's, this assistant can be connected to different devices such as speakers, domestic houses, or IOT products. Natural language processing comes from Tellme Networks, which was purchased by Microsoft in 2007 [2].

⁴Image source: <https://www.muycomputerpro.com/2019/09/24/>

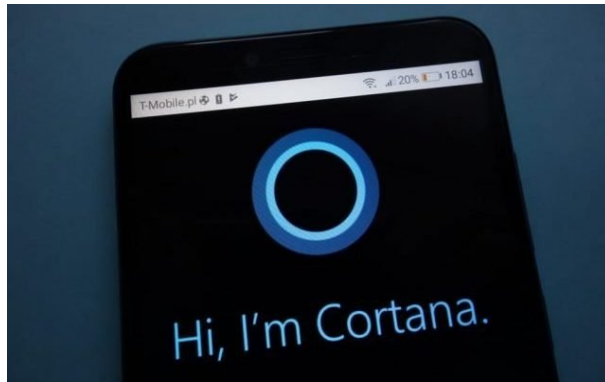


Figure 8: Cortana on Android ¹

Alexa 2014

She is the virtual assistant developed by Amazon, and its main feature is that it can be used with the intelligent speaker called Amazon Echo. This allows us to know information about weather, products, purchases, reminders, and even make calls. She can recognize five main languages, and it is possible to use it from IOS and Android platforms [17].



Figure 9: Amazon echo with Alexa assistant ²

¹Image source: <https://www.microsoft.com/en-us/cortana>

²Image source: <https://developer.amazon.com/es-ES/alexa>

3.3 Applications

Chatbots are currently, in many different work areas, increasing the popularity that they have nowadays. We can find them in basically any business line. Chatbot applications streamline interactions between people and services, enhancing customer experience. At the same time, they offer companies new opportunities to improve the customer's engagement process and operational efficiency by reducing the typical cost of customer service.

3.3.1 Medicine

In China, Microsoft published a chatbot called Xiaoice which is used by around 40 million people and has recorded up to 10 billion conversations. It was designed by developers and psychological experts to create a balance between artificial intelligence and emotional intelligence. This chatbot can memorize and track the emotional state of the users and help them with intelligent and timely dialogues until a human doctor can contact them. On the other hand, IBM uses the Watson services focused on medicine for different uses, such as Consult general medicine and telemedicine. [17]

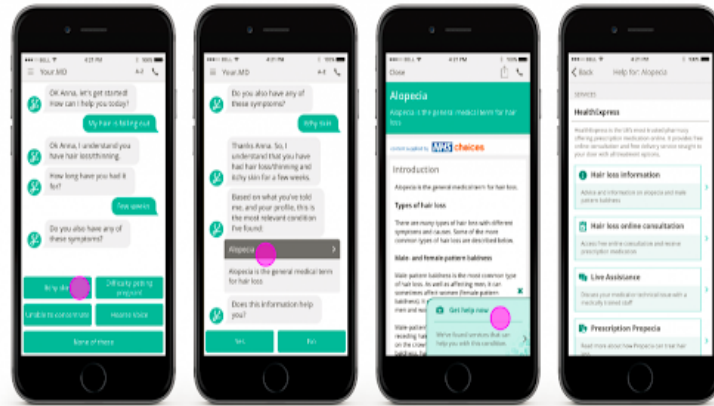


Figure 10: Medical Chatbot ¹

3.3.2 Customer service

According to a survey conducted by the Genesys company, 82 % of the companies consulted in Spain are aware of the boom of new technologies like artificial intelligence. They know how the world is changing every day and how this new form of communications, give new rules to the way of connecting with customers.

¹Image source: <https://news.microsoft.com/>

In this sense, 66% of companies would be thinking of implementing some Bots solution based on artificial intelligence in the coming months.

Gartner also says that 85% of interactions with customers will be managed with bots in the year 2022. However, there are already significant advances. For example, Spain is now positioned in Europe as the first place regarding the capacity of companies to respond to repetitive questions, according to the cited study by ServiceNow. [18]

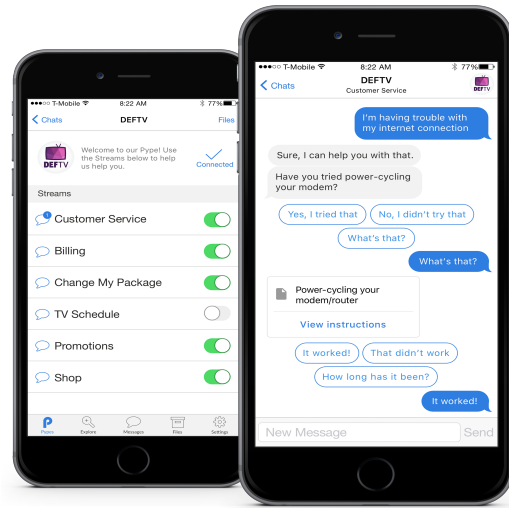


Figure 11: Customer service chatbot ²

3.3.3 Learning

Nowadays, there are many similar partnerships between corporations and educational institutions that try to make learning methods more efficient. In 2016, Bill Gates announced that the Bill and Melissa Gates Foundation would invest more than 240 million dollars in a tech project. It personalizes institutional learning using Artificial Intelligence. Facebook has also followed Bill Gates's example and joined the world-famous Summit Learning project. [11]

The project that involves Bill and Melissa Gates Foundation, Facebook and Summit public school, uses bots for basic lecturing. The idea is that chatbots can serve as virtual advisers, who will adapt their work and curriculum to the students' abilities. This way, the students can follow at their own learning pace. In the Summit Learning project, chatbots serve as vertical tutors. They engage in a dialogue with each student and determine the areas where they are falling

²Image source: <https://www.servicenow.es/products/virtual-agent.html>

behind. Then, chatbots use this data to compose an entirely personalized learning program that focuses on troubling subjects. Their job is also to follow the students' advancement from the first to the last lesson, check their assumptions, and guide them through the curriculum. [1]

4 Development Methodology

Based on the research carried out and the proposed goals for this project, the development of a chatbot prototype has been completed, using the “Dialogflow” development framework, and as a means of entry for “Facebook Messenger” interactions. This allows to contribute with the minister of social inclusion to improve of its processes of delivery of information to their users. In addition, the use of a methodology for the development of agile technologies is analyzed.

4.1 Solution Methodology

In order to perform all the tasks of this research, it is necessary to apply a Work Methodology, which is applied to all areas of this dissertation. Each of the phases of Design Thinking are oriented towards a common objective: Identify the client’s requirements and solve them through continuous feedback loops. All this will help the development team to implement the ideas and resolutions avoiding errors when reaching the production environment. We have decided to use BDD as a common language to describe the requirements requested by the client, and that will be implemented for this chatbot project. Each requirement has been captured as user stories to be solved by the chabot. This will also help us evaluate the chatbot in a future chapter.

4.1.1 Design Thinking

Design Thinking [19] is not an exclusive methodology of designers: all the great innovators in literature, art, music, science, engineering, and business have practiced it. The unique characteristic about Design Thinking is the work processes of the designers that allow to systematically extract, teach, learn and apply in the prototype of an intelligent agent to solve the most complicated problem that exists when you need to obtaining knowledge from processes in a disruptive way.

Different stages must be completed to extract the necessary information and generate a functional prototype as we can see in the figure 12:

1. Empathise;
2. Define;
3. Ideate;
4. Prototype;
5. Test;
6. Implement.

It is essential to understand that the methodology and its six phases are not always sequential. A specific order must not be followed and can often occur

in parallel. The method suggests repeating iterative, finding a balance between the prototype and the requirements. Given that, we should not understand the phases as a hierarchical process or step by step. This methodology emphasizes understanding them as phases that contribute to an innovative project, rather than sequential steps of a product. The team responsible for using the Design Thinking procedures continuously evaluates the results to analyze, question, and improve their initial assumptions, understandings, and results. The results of the final stage of the work process help to improve the knowledge of the problem, help us to better determine the parameters of the problem, redefine and, perhaps most importantly, provide us with new ideas so that we can see any viable alternative, which could not have been available with our previous level of understanding.

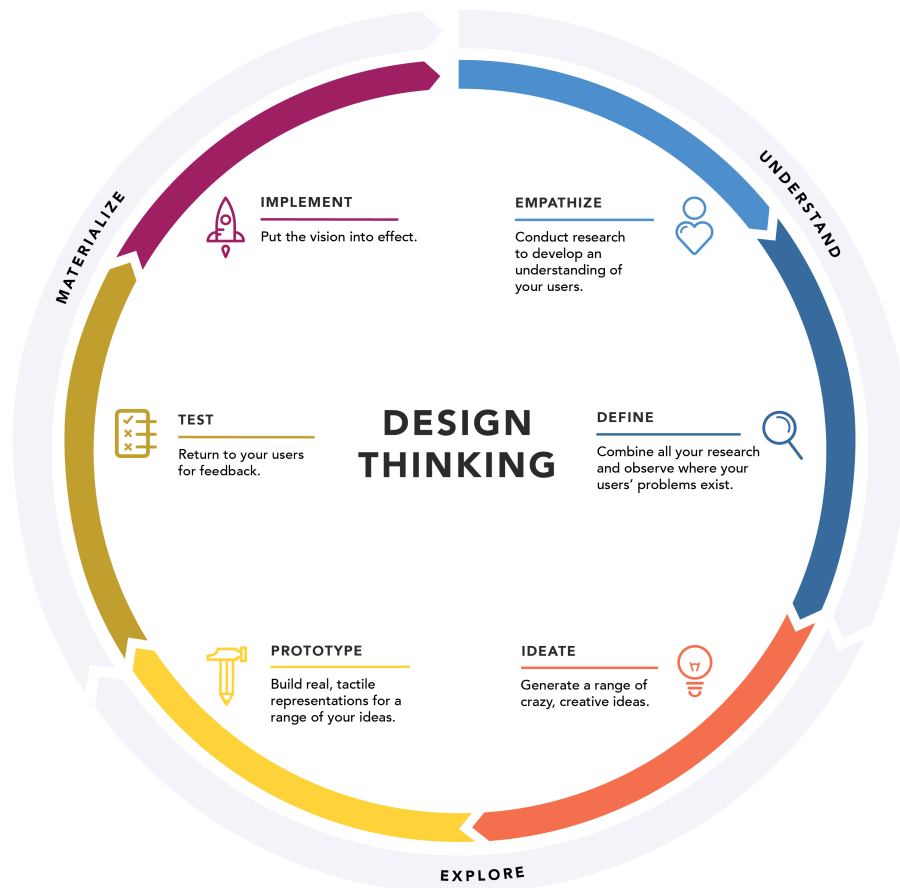


Figure 12: Design Thinking Methodology ¹

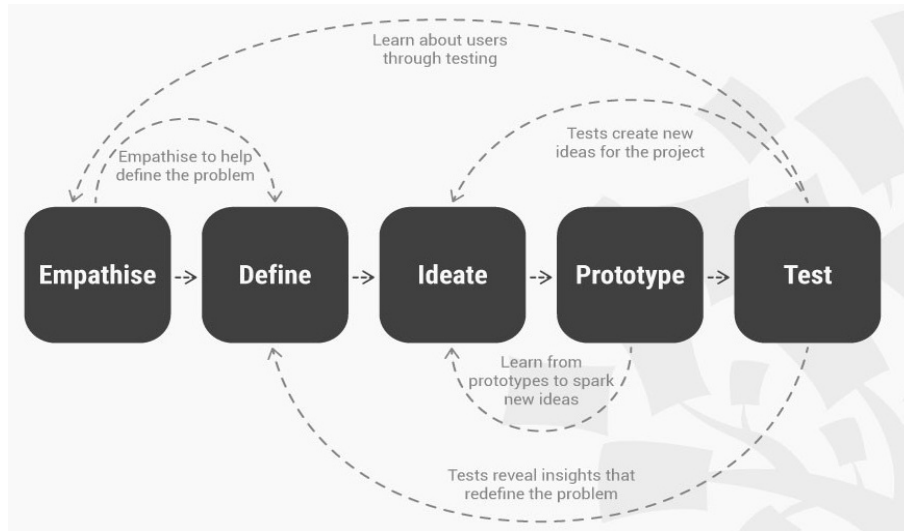


Figure 13: Design Thinking Methodology cycles.²

In our project we have continuously conducted a series of interviews with the Marketing department of the company Couching team. The purpose of the interviews has been related to their daily work and the processes they execute in each of their social networks. We have conducted the interview personally, in a neutral environment and away from work to understand the perception of their daily processes. In a field evaluation, we have participated in their processes silently to observe and empathize with them.

The purpose of the interviews were focused on finding how they relate to users and determining the best benefits they can obtain from the implementation of an intelligent agent to lighten the operational load of users.

4.1.2 Behavior Driven Development

Behavior Driven Development, commonly called BDD, is not a testing technique, such as Test Driven Development or TDD.

DBB proposes to define a common language for the users and technicians. It is to use as a fundamental part of the development and testing. Both approaches could coexist, since an acceptance test could be specified, and then refined in unit tests when coding that functionality in the different layers. The clear advantage of BDD is that developers focus on seeing what has to work. Besides, testing analysts can learn more about feature acceptance tests.

¹Image source: <https://www.designthinking.es/inicio/>

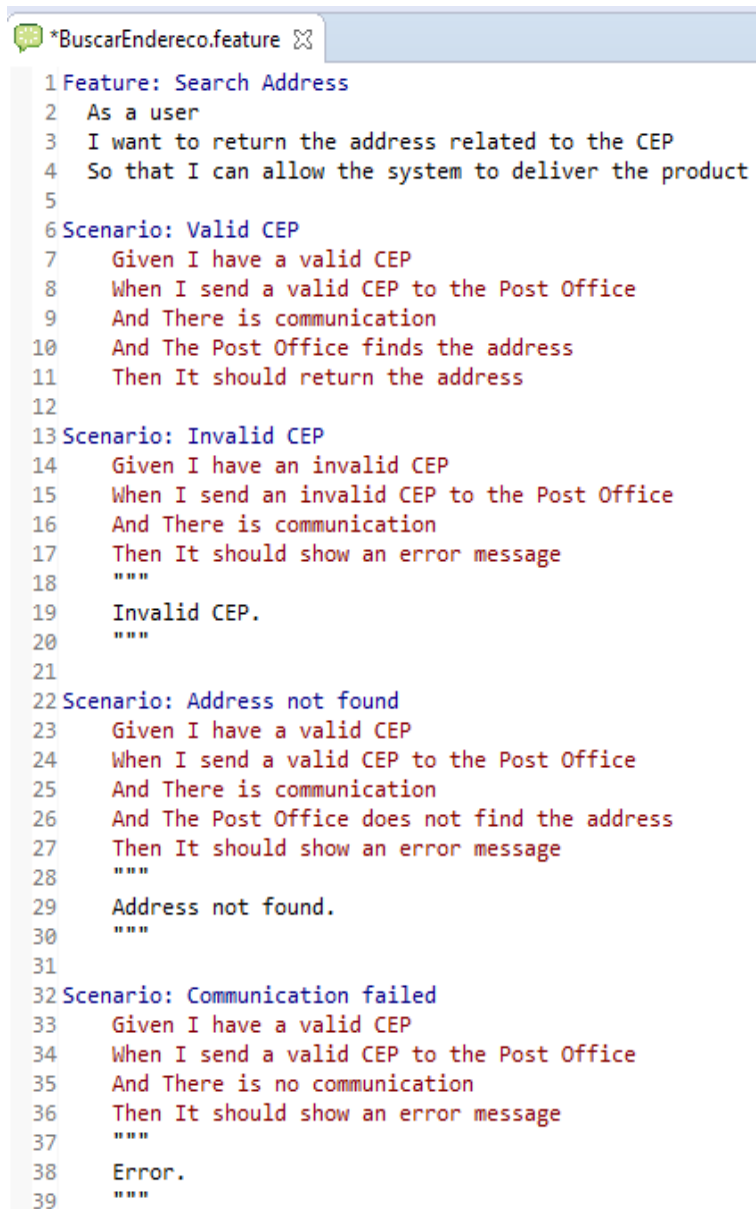
²Image Source: startupbootcamp.org

This strategy fits well in agile methodologies since they usually specify requirements as user stories. These user stories must have their acceptance criteria, and hence acceptance tests emerge, which can be written directly in the Gherkin language. We must take into account specific parameters to use requirements in BDD:

- Each requirement must become a "User story," and specify examples;
- Each example must be a scenario of a user in the system;
- Be aware of the need to define "the specification of a user's behavior" instead of "the unitary test of a class."

The team must write together with the users the necessary scenarios in addition to discussing the terms of acceptance. The tester would be in charge of ensuring that they are well written, complete, with the coverage according to the defined test strategy. Typically, these tests will be stored in ".feature" files, which should be versioned next to the source code of the product. The most common structure used includes the following points:

- Feature: It is the name of the functionality we are going to test, the title of the test;
- Scenario: There will be one for each test that you want to specify for this functionality;
- Given: This attribute marks the preconditions;
- When: The actions to be executes in each scenario;
- Then: Here, we specify the expected result and the validations to consider.



```
*BuscarEndereco.feature
1 Feature: Search Address
2   As a user
3   I want to return the address related to the CEP
4   So that I can allow the system to deliver the product
5
6 Scenario: Valid CEP
7   Given I have a valid CEP
8   When I send a valid CEP to the Post Office
9   And There is communication
10  And The Post Office finds the address
11  Then It should return the address
12
13 Scenario: Invalid CEP
14   Given I have an invalid CEP
15   When I send an invalid CEP to the Post Office
16   And There is communication
17   Then It should show an error message
18   ""
19   Invalid CEP.
20   ""
21
22 Scenario: Address not found
23   Given I have a valid CEP
24   When I send a valid CEP to the Post Office
25   And There is communication
26   And The Post Office does not find the address
27   Then It should show an error message
28   ""
29   Address not found.
30   ""
31
32 Scenario: Communication failed
33   Given I have a valid CEP
34   When I send a valid CEP to the Post Office
35   And There is no communication
36   Then It should show an error message
37   ""
38   Error.
39   ""
```

Figure 14: BDD User Story Example

5 Solution architecture

It is important for the implementation of a right architecture to know correctly the requirements presented by the client and analyze them in a technical way. This must be supported by the technologies used. In this chapter we will analyze the requirements raised with the client in each of the interviews, as well as we will propose a viable architecture and technologies that allow them to be developed and solved.

5.1 User Requirements

To determine a correct architecture of the solution, we broke down and analyze the requirements requested by the client. We sought to find a robust architecture that supports our agent and allows it to grow in the future according to the reported needs. Based on the proposed architecture, we have chosen technologies that support and contribute efficiently.

5.1.1 Functional Requirements

In order to know the basic requirements that we need in the project, weekly meetings were planned that allowed us to know the needs of the users. The ministry of social inclusion has detected four basic functionalities that it wants to attend to and that users ask constantly:

1. Personal information about previously obtained bonuses.
2. Places and available hours in which they can come to receive the payment.
3. What requirements do they need to collect it?
4. What places are close to their home to collect it?
5. Information and requirements of other programs that the ministry has.
6. Information related to symptoms and recommendations of Covid 19.
7. Basic information of the ministry such as hours and locations near their home.
8. Information on the procedures that can be carried out online in other public institutions.

Most of the requirements are public information and available on the website <https://www.inclusion.gob.ec/>, but because it is not found directly, it cannot be easily located. This page, in addition to providing the required information, has topics such as projects, reports, technical data among others. The ministry, aware of this problem has begun to carry out notification and help campaigns, but users prefer to use direct contact through Facebook Messenger. Through analysis and conversations with those in charge of the communication department, we have converted these 8 specific requirements into functionalities and

scenarios that the chatbot can understand and carry out its response processes.

The information should be obtained from a private server where the answers to the required data are stored. In this first instance of the agent, we fed the default and suggested questions and answers. In future instances, we hope to hear the response of users to the agent in order to provide feedback with doubts and questions that may arise in their daily use. The ministry wishes to evaluate the annual costs that are incurred for the use of the platforms taking into account that the server services will be provided by them. In addition, the information in the database should not leave the private cloud due to state policies.

Feature 1 : As a user, I want to know if I am a beneficiary of the "Bono de Desarrollo". In Table 1 in Appendix A we have the ideal flow for a user in which his ID has been correctly validated and we have the personal information for the Development Bonus. We also analyze the flows in which the ID does not correspond to the check digit. The alternate flows are detailed in Table 2 through Tables 5 in the Appendix A.

Feature 2: As a user, I want to know the places where I can go to collect money. In Table 6 in Appendix A, we have the ideal flow to find the places closest to the user's home. The list of Provinces and cities has been delivered by the ministry. Not all states and cities in Ecuador are presented as there are no service centers in all cities. For these cases, we also analyze flows in which they do not have a center close to home. The alternate flows are detailed in Table 7 and Tables 8 in the Appendix A.

Feature 3: As a user, I need information about programs that MIES is working on. In Table 9 in Appendix A, we detail the scenario for the search for information on state programs. The list of programs and details of requirements for each one will be provided by the information department. The alternate flows are detailed in Table 10 and Table 11 in the Appendix A.

Feature 4: As a user, I ask for information about Covid 19. Table 12 in Appendix A, lists a suggested option for citizenship information. This functionality is not intended to work as a symptom triage. We wish to present general information on symptoms and recommendations. This information will be taken from public and world entities. If users wish to be contacted for any of these symptoms or for more individual recommendations, the information of the public health ministry will be delivered for their contact. The alternate flows are detailed in Table 12 through Table 14 in the Appendix A.

Feature 5: As a user, I ask for information offices and their hours. Table 15 in Appendix A, presents the requirements when looking for general information from the ministerial institution. This information is general and has no derivative flows.

Feature 6: As a user, I ask for information about other procedures that are available online. In Table 17 in Appendix A, we present the information on other programs related to the ministry of economic and social inclusion. This information is public and does not present divergent flows.

Feature 7: As a user I start the conversation. In Table 19 in Appendix A, we present a mandatory initial flow. At the beginning of a conversation with the agent, a menu of options that the user can use should be presented. The user must also be informed that the chatbot is a conversational agent. In flows in which the questions are not understood, the options must also be notified and reiterate that it is a non-human operator.

Feature 8: As a user, I said something that the chatbot did not understand. The agent must have a basic answer when the requested information is not understood. This will also help us train you in future occasions.

5.1.2 Non Functional Requirements

The intelligent agent has been implemented under the following guidelines:

1. No private information of users should be stored by any public cloud on the internet.
2. The NLP analysis of the information fed by the agent should be performed using Dialogflow. This information will not be stored by public policies.
3. The database, from which all the information is obtained, is private and belongs to the Economic and Social Ministry.
4. No additional infrastructure requirement may be requested that the Ministry does not have.
5. For this project, the Free Tier of Dialogflow and Facebook Developer should be used to avoid incurring additional expenses.
6. The security of the servers will be in charge of the systems department and any request should be made by official letter to them.
7. The response capacity to transactions is established by the systems department of the Ministry.
8. All system functionality and business transaction must respond to the user in at least 10 seconds.
9. The system must be able to operate adequately with up to 1,000 users with concurrent sessions.
10. The system should not be transactional. It should only allow the consultation of information.
11. The Agent should respond to the Latin American Spanish language.

5.2 Architecture Solution

To determine a correct architecture, we have analyzed the requirements requested by the client. We need a means of inputting information, an intermediate channel to understand the messages delivered by the user and help to our chatbot to decide which action is calling for the user and a layer to connect with a database where the information is located for building the responses to be delivered. The architecture are detailed below.

5.2.1 Communication Architecture

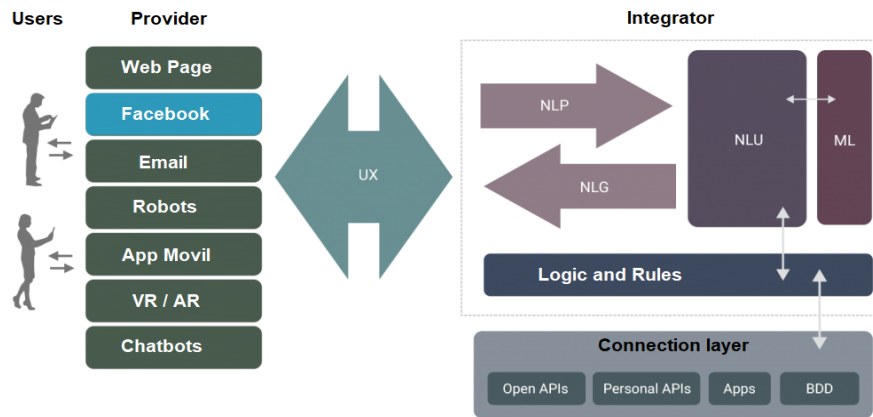


Figure 15: Architecture proposal

The architecture used in this project was implemented by Dion Hinchcliffe in 2016 and consists of the following layers [6]:

Provider: This layer is defined as the interface through which the user communicates with our bot. In our project, we are using Facebook Messenger as a provider. In the experience that the user perceives when interacting with chatbots, we can focus on two types of elements: Interface and Writing. The first helps to display the content within the channel and, by default, the Provider provides it, for example, Facebook and its Messenger interface. The second refers to how the bot communicates with the user using: texts, images, buttons, videos, events, and, in short, the whole series of tools that controls answering all that the bot send to the user.

The Integrator: The integrator is the critical piece of a chatbot. It delivers all the tools of NLP, Machine Learning, and the Artificial intelligence of the chatbot. Within the integrator, we find the Natural Language Processing and refers to all process that the machine performs to obtain, identify and analyze

the message sent by user. Once processed it communicates with the Natural Language Understanding, Which, according to the intention identified in the message, executes or performs one or more required flows.

Connection layer: This layer integrates all external connections, such as APIs, Database, Webhooks, necessary to perform external tasks or queries. This channel is a means of output, storage and consultation.

5.3 Technologies Solution

Nowdays, developing a chatbot with tecnological tools requires training and time. Specially when we are creating flows of questions and answers. We can upload documentation and let dedicated platforms like Watson use their artificial intelligence to search for answers, mainly with keywords. Or we also have other types of platforms such as Dialogflow that allow direct control of each of the question and answer options and allow the agent to train as it interacts with users.

5.3.1 Frameworks and tools to develop chatbots

For the creation of an agent which can imitate human conversation and can offer personalized services to hookup them, we look for a chatbot builder. A chatbot builder, or a chatbot development platform, is an application through which anyone can construct a smart, conversational agent for popular messaging platforms. Some have visual, drag-and-drop builders for low-code users, while others leverage machine learning to ensure the chatbot does not just represent a company with pre-learned responses but also learns from every interaction. The inclusion of APIs and SDKs are also typical for enterprise chatbot builders. A chatbot publishing platform, on the other hand, is the channel through which users can access the chatbot. The most popular channels are Facebook Messenger, Slack, or WhatsApp. In the market, we have a bunch of platforms, but for this research, we try to take some powerful Bot at low cost. We identified three which can handle our requirements: Microsoft Bot Framework, Wit.ai, Dialog Flow. [14] After comparing the most popular frameworks, we decided that Dialogflow is right platform for this research.

5.3.2 Facebook developer tools

Facebook achieved its popularity as a social network capable of containing applications made by third parties, thus allowing the conduct of business from it. To help developers, Facebook create a group of tools which are grouped in the development framework called Facebook AP, or what they have called "Facebook for developers". The most outstanding in Artificial intelligence are Wit.ai and AR Studio. For Corporate Tools we have Canvas API, FB Analytics, Business SDK, Authentication, Messenger. They also have some other very popular tools like: Gameroom, Live broadcasts, Graph QL, Live API FB, Instagram

Business and more. Most of these are REST-based interfaces that allow access to user profile data, friends, photos, and user events through the use of GET or POST messages. The queries are made using a language called Facebook Query Language and have a markup layout similar to HTML that allows it to be interleaved with the Facebook Markup Language. Using FBML developers can access different points of the network such as the profile, profile actions, canvas and feeds. This language also includes support for AJAX and Javascript.

Undoubtedly, one of the most used tools on Facebook is its Messenger, which has become one of the most used channels on the web. It is for them that Facebook Developers also has services to be able to connect and obtain useful information about events and messages sent through its channel.

5.3.3 Facebook Messenger Platform

When a user starts a conversation using Facebook Messenger, a detailed flow begins. Facebook Messenger connects to a webhook URL provided by the developers. It also has a return webhook that awaits the information that will be returned to the user. This allows developers to create guided conversations for people to perform an automated process or develop an app that serves as a link between smart agents and Facebook Messenger.

Facebook Messenger presents many options to capture messages; among them, we highlight the use of Webhooks. The Messenger platform sends events to the webhook to notify the Bot when a variety of interactions or events occur, even when a person sends a message. The Messenger platform sends webhook events as POST requests for further development and decision making.

From the services presented in Table 1, we will subscribe to "Messages", "Postbacks", "Reads", "Echoes". These services are the minimum necessary to connect with our Dialogflow agent in order to create communication flows.

5.3.4 Dialogflow platform

Dialogflow is a chatbots creation tool capable of understanding natural language and providing infrastructure to recreate conversations and build dialogues in order to interact with the user in a fluid way. It belongs to Google since its purchase in September 2016.

Dialogflow stands out among its competitors due to the wide range of conversation interfaces that it includes: Google Home, wearables, cars, phones, etc. It currently supports more than 14 languages and is increasingly able to cope with the use of abbreviations and spelling errors.

A basic conversation flow between the user and our chatbot consists of the following steps:

1. The user gives an input; the input can be a voice command, a request, or merely a question.

messages	Subscribe to received message events.
messaging_account_linking	Subscribe to account linking events.
messaging_checkout_updates (beta)	Subscribe to updated payment events.
message_deliveries	Subscribe to message delivered events.
message_echoes	Subscribe to echo message events.
messaging_game_plays	Subscribe to the instant game events.
messaging_handovers (beta)	Subscribe to protocol events "Handover."
messaging_options	Subscribe to the plugin acceptance events.
messaging_payments (beta)	Subscribe to payment events.
messaging_policy_enforcement	Subscribe to policy application events.
messaging_postbacks	Subscribe to the postback received events.
messaging_pre_checkouts (beta)	Subscribe to events prior to payment.
message_reads	Subscribe to message read events.
messaging_referrals	Subscribe to the reference events.
standby (beta)	Subscribe to the events of the channel waiting for the "Handover" protocol.

Table 1: Facebook services to suscribe

2. The Dialogflow agent (or natural language compression module) extracts each of the parameters from that input.
3. The agent returns the response (previously programmed) that corresponds to the input.

To have a more concrete idea of how an agent works, we must explain the second step in more detail.

First, it is essential to know that an agent is composed of several intents. An intent categorizes, defines, and maps the user's intention to react with an action or response.

When a device with our integrated chatbot receives a voice command, the conversion to text is first carried out, then that text is processed through NLP to extract each of the parameters and finally it is carried out. It then performs a procedure called ML Ranking (Machine Learning Ranking) to find out what intent the command corresponds to. One of the positive points in favor of Dialogflow is the ability to offer the possibility of creating simple chatbots easily, as well as offering multiple interfaces for the creation of more complex chatbots that can be adjusted to our needs and requirements.

Dialogflow offers the possibility to create our chatbots from a graphic interface. In a few clicks, we can have our first chatbot. We have at our disposal a menu from which we can program our intents, define synonyms, establish keywords, contexts, etc. We can even select the interface in which we want to

integrate our chatbot. On the other hand, we can develop our chatbot with code in Node.js language. With this variant, we have the possibility of exploiting to the maximum the capacities that the tool offers us. Specifically, in addition to being able to incorporate all the features explained above, we have within our reach to get our agent to carry out actions, either with other Google tools, such as Google Calendar, Gmail, BigQuery, or with third-party applications.

5.3.5 Basic concepts

To start structuring our chatbot on the Dialogflow platform, we must first review some basic concepts that will be named throughout the implementation process.

Agents: An agent is a chatbot. In simple projects, one agent will be enough to group all the required flows. However, in larger projects, you could have several agents for the same project (brand, store, account), each of them consolidating a particularity or common intention of the conversation. For example, our agent could be the "Customer Service Agent".

Intents: They are the client's intentions or, in other words, what he wants to communicate. Here we define the possible questions or requests of the clients and what possible answers we want to give them. For example: if the client ask for the status of an order.

Entities: The entities are the tools available to extract parameters or variables from what the user has said. For example, when a user says, "I want to check the status of the order with ID 0237908," we will use the entities to extract the value of the order ID the user is referring to. Dialogflow uses internal functionalities to recognize the parameters automatically and with a high percentage of success.

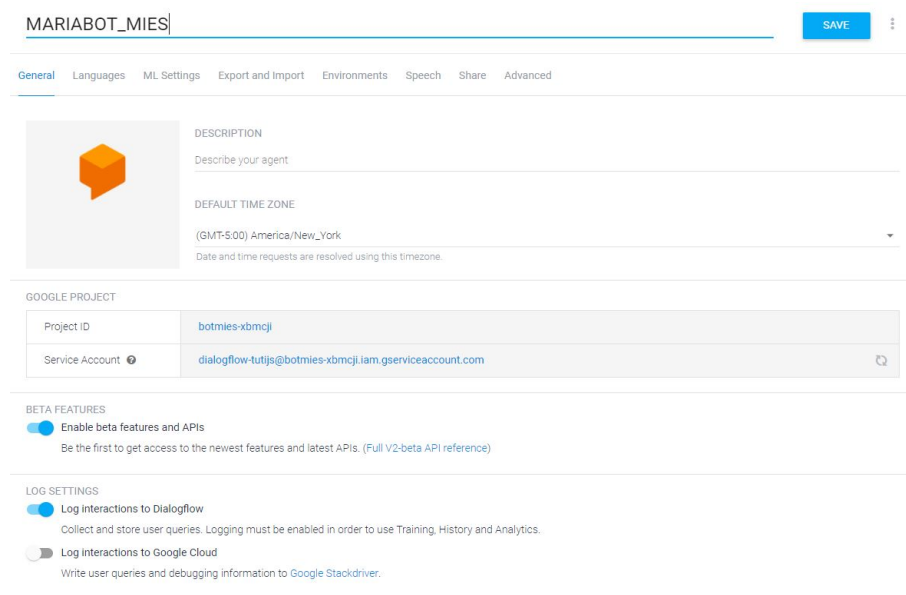
Actions: Actions are what we do when the intention of a user is detected. When what the user says corresponds to an Intent, then the parameters are extracted and passed to the action that was defined for that intent, which can be a predetermined response or a specific action.

6 Development of the proposal

Throughout the chapters we have analyzed the requirements and the technologies necessary to implement them. In this chapter we focus on how we have developed the solution of our project and the different difficulties that we have encountered throughout the implementation process.

6.1 Configuration on Dialogflow

The basic configurations of Dialogflow are carried out on the online platform that they provide from their website <https://dialogflow.cloud.google.com/>. First, get set up with a DialogFlow Account. Then we can create an Agent by clicking ‘Create Agent’ in the column menu to our left. We need to give a name to the Bot. In this case, we call it ChatbotNLP. It is important to select right the time zone and language of our agent.



The screenshot displays the Dialogflow console interface for configuring a new agent. At the top, the agent name 'MARIABOT_MIES' is entered in a text field, with a 'SAVE' button to its right. Below this is a horizontal menu with tabs: 'General', 'Languages', 'ML Settings', 'Export and Import', 'Environments', 'Speech', 'Share', and 'Advanced'. The 'General' tab is selected. On the left side of the 'General' tab is a placeholder image of an orange cube. The main content area contains the following sections:

- DESCRIPTION**: A text input field with the placeholder 'Describe your agent'.
- DEFAULT TIME ZONE**: A dropdown menu currently set to '(GMT-5:00) America/New_York', with a note below stating 'Date and time requests are resolved using this timezone.'
- GOOGLE PROJECT**: A table with two rows:

Project ID	botmies-xbmcij
Service Account	dialogflow-tutjs@botmies-xbmcij.iam.gserviceaccount.com
- BETA FEATURES**: A toggle switch for 'Enable beta features and APIs' is turned on. Below it is a link: 'Be the first to get access to the newest features and latest APIs. (Full V2-beta API reference)'.
- LOG SETTINGS**: Two toggle switches are present. The first, 'Log Interactions to Dialogflow', is turned on, with a note: 'Collect and store user queries. Logging must be enabled in order to use Training, History and Analytics.' The second, 'Log Interactions to Google Cloud', is turned off, with a link: 'Write user queries and debugging information to Google Stackdriver.'

Figure 16: Basic Configuration on Dialogflow

As we can see in Figure 16, when we are creating an agent, we will automatically have a basic initial configuration. It was not necessary to modify it for this project. The project ID, and the Service Account allow us to connect with Facebook Developers. We can also create development environments that help us with the management of large projects. In this case, we have generated two environments: Development and Production. ML Settings allows us to configure the Agent’s training and learning mode. The agent have generated errors at the time of training automatically.

6.1.1 Entities

The agent needs different entities to help us understanding the context used by the user. Entities are parameters defined as the type of information we want to extract from user input. For example: "City". City can be understood if the user call about: "Quito", "Guayaquil", "My Town", "My house". All entitys can be understood like synonyms or names or nicknames, etc. That is very important to understand because it can vary from user to user. One user can call his town : "Quito", but others can call it like "the capital", "quito", "kito" or the oficial name: "San Francisco de Quito". In the Figure 17, we have a look of the seven custom entities that allow us to help the agent. These are:

1. Cedula (ID): personal and non-transferable identification number.
2. Ciudades (Cities): Population where a group of people lives.
3. Dígito (Number): Identity card validation number.
4. Programas Mies (Ministry programs): Events and benefits that MIES owns and accredits people according to their needs.
5. Provincias Ecuador (Provinces of Ecuador) : Territorial administrative division in which Ecuador is organized.
6. Sintomas (Symptoms): Physical problem that a person presents, which may indicate a disease or condition.
7. Tramites (Procedures): Any request or delivery of information that people can make to the MIES.
8. Telefono (Phone): Entity in charge of the validation of telephone numbers.
9. Correo electronico (email): Entity in charge of the validation of email.

All them are in Spanish because the users that interact with the agent are people from Ecuador, whose principal language is Spanish.

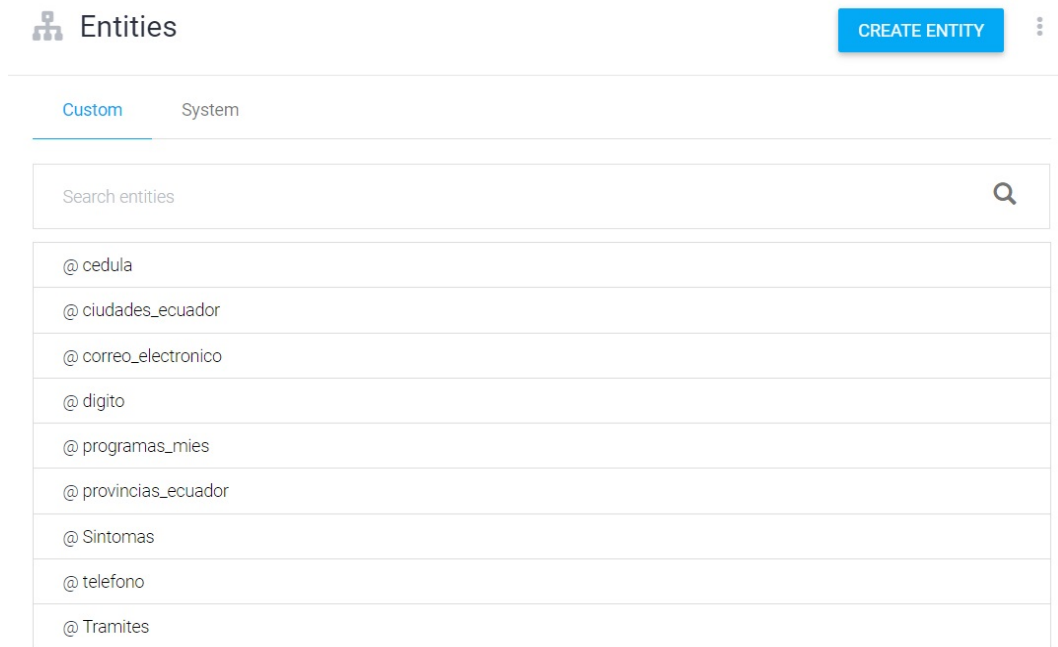


Figure 17: Entities of conversational agent

In our entities it can be seen that there are no entities for the cases as 'synonyms'. It is not necessary to create this type of entity, since Dialogflow has filters and schemas as regular expressions that help us to validate the information and relate it. This allow us to find correct information for identity cards, verification digits, telephone numbers, emails, or any alphanumeric value that need to be validated, as shown in Figure 18.

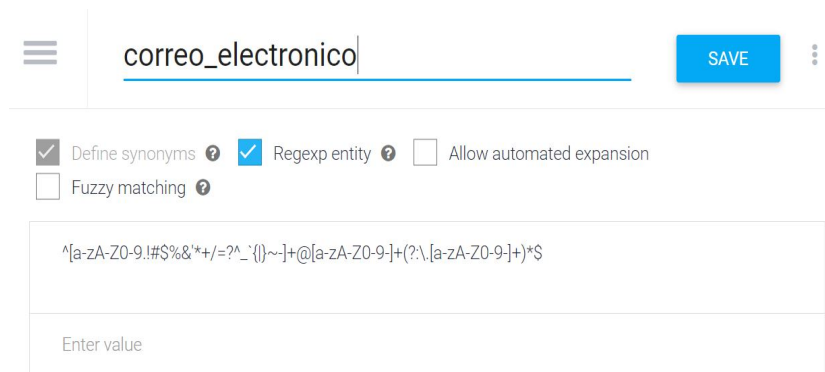


Figure 18: Entities with regular expressions

In Figure 19, we also have examples of entities with "Fuzzy Match" type. When used, this feature allows us to avoid an exact match in the entrances of the entities. For example, if we have a city with the name of "Santo Domingo de los Colorados", it will allow to find the entity only with "Santo Domingo", "santo dommingo", or even with some basic spelling errors. Fuzzy Matching should not be enabled for entities that are used to match strict values such as IDs or National Identification Numbers.

ciudades_ecuador

SAVE

☐ Define synonyms ? ☐ Regex entity ? ☐ Allow automated expansion

☒ Fuzzy matching ?

Search entries

1 OF 3

Cuenca
Gualaico
Paute

Figure 19: Entities with Fuzzy matching

6.1.2 Intents

To cover the requirements, as can be seen in Figure 20, we have created 13 intents that respond to the requested requirements. Each of them has its own flow of response and interact with the created entities. These intents are:

1. Beneficiario (Beneficiary): Intent which identify if the user has an economic bonus.
2. Cedula (DNI) : Intent that validates the user authentication using DNI.
3. Lugar cobro (Collection locations) : Intent which identifies the closest place to the user's residence.
4. Otros bonos (Other Bonus): Intent which provides information about other economic benefits offered by the MIES.
5. Sintomas (Symptoms): Intent which provides information about symptoms and recommendations for COVID 19.

6. Horarios (Schedule): Intent which provides information about schedule of closest offices to the user's residence.
7. Tramites (Procedure): Intent which provides information about some on-line procedures.
8. Salida Positiva (Positive output): Intent that identifies a farewell from the user, with positive or appreciative characteristics.
9. Salida Negativa (Negative output): Intent that identifies a farewell from the user, with negative or derogatory characteristics.
10. Verificar digito (verify digit): Intent that validates the user authentication using validation number in the DNI.
11. Provincia (Province): Intent that identifies the user's province of residence.
12. Seleccion programa (Province): Intent that identifies the program the user wants to select.
13. Seleccionar tramite (Procedure): Intent that identifies the process that the user wishes to consult.

By default, we also have a welcome intent which can start the interaction or "wake up" the agent, and another intent in cases where the agent does not understand the information entered by the user.

Figure 20: Intents of the Agent

In each of the intents we have configuration options that allow us to control the conversation flows. For example, in Figure 20 we have a context in a second

intent, which let us control and understand previous ideas in the conversation like the DNI when validating the verification digit entered by a previous intent.

The screenshot shows the configuration interface for the 'verificar_digito' intent. At the top, there is a header bar with a menu icon, the intent name 'verificar_digito', an information icon, a 'SAVE' button, and a vertical ellipsis. Below the header, the 'Contexts' section is expanded, showing two input contexts: 'Cedula-followup' and 'Cedula', each with a close icon. To the right of these is a text input field labeled 'Add input context'. Below the input contexts is an 'Add output context' field with a close icon.

Figure 21: Context configuration of an Intent

Events allow finding actions performed by the user that trigger specific flows. In Figure 22 for example, the welcome attempt reacts to the "Facebook Welcome" event that is activated when the user starts using the agent from Facebook messenger.

The screenshot shows the configuration interface for the 'Default Welcome Intent'. At the top, there is a header bar with a menu icon, the intent name 'Default Welcome Intent', a 'SAVE' button, and a vertical ellipsis. Below the header, the 'Events' section is expanded, showing two events: 'Welcome' (with an orange circle icon) and 'Facebook Welcome' (with a blue speech bubble icon), each with a close icon. To the right of these is a text input field labeled 'Add event'.

Figure 22: Events configuration of an Intent

The training phrases allow us to start the agent training with phrases that will help to understand when a user needs to activate a flow. These opening phrases will be the first line of training to understand the message. Although at the beginning they must contain the textual message, with given interaction the agent trains itself to understand phrases that are similar but not exact. For example, in cases that are written with misspellings or punctuation marks. Figure 22 show the first training words for the intent "Welcome".

Default Welcome Intent

SAVE

Training phrases ?

Search training p

” Add user expression

” Empezar

” empezar

” hola

” hey

” saludos

” hey ho

” chao

Figure 23: Training Phases of an Intent

6.1.3 Actions and parameters

Actions and parameters allow you to analyze the message and find previously established entities. For example, in the intent of "Provincia", it is expected to find an entity associated with "Provincias del Ecuador". We can also accept a list of entities and determine if they are required or not to continue the flow.

provincia

!

SAVE

Action and parameters

^

lugar_cobro.lugar_cobro-custom

REQUIRED ?	PARAMETER NAME ?	ENTITY ?	VALUE	IS LIST ?
☐	provincias_€	@provincia s_ecuador	\$provincias _ecuador	☐
☐	Enter name	Enter entity	Enter value	☐

+ New parameter

Figure 24: Action and parameters of an Intent

The answers of each attempt are delivered according to the medium to which it is associated. For example, when responding via Facebook, we can send a short list of answers to help people understand the options of the chatbot shown in Figure 25. This is not mandatory but helps the user to understand how it works. You can also send a customized structure if the input medium supports it. For example, we can send links of files, photos, carousels, etc. This is because the responses are issued in a Json structure as the one in the Figure 26.

Default Welcome Intent

SAVE

Responses ?

DEFAULT FACEBOOK +

Responses from this tab will be sent to the Facebook integration.

Use responses from the DEFAULT tab as the first responses. ☐

Text Response

1

Hola, me llamo Maria 🇵🇷 y soy un agente inteligente y autónomo 🤖 que el MIES ha desarrollado para ayudarte.

2

Enter a text response variant

Quick Replies

¿Como puedo ayudarte?

Soy Beneficiario? 🇵🇷

🏠 Lugares de cobro

Otros Bonos 🇵🇷 🇵🇷

🏠 Sintomas Covid

Oficinas 🇵🇷

📄 Otros Trámites

ADD RESPONSES

Figure 25: Response's of an Intent

Diagnostic info

Raw API response

Fulfillment request

Fulfillment status

```
1 {
2   "responseId": "dca57d89-8254-4462-bb52-5d91db50aaba-0820055c",
3   "queryResult": {
4     "queryText": "empezar",
5     "action": "input.welcome",
6     "parameters": {},
7     "allRequiredParamsPresent": true,
8     "fulfillmentText": "Hola, me llamo Maria 🙋, y soy un agente inteligente y autónomo 🤖 que el MIES ha desarrollado para ayudarte.\nPara acceder a la consulta, escribe la palabra en *negrita*.\n\n¿Como puedo ayudarte?\n1. Soy *Beneficiario*? 🙋\n2. 🏠 *Lugares* de cobro\n3. Otros *Bonos* 🙋\n4. 🤒 *Síntomas* Covid\n5. Las *Manuelas* 🙋",
9     "fulfillmentMessages": [
10      {
11        "text": {
12          "text": [
13            "Hola, me llamo Maria 🙋, y soy un agente inteligente y autónomo 🤖 que el MIES ha desarrollado para ayudarte."
14          ]
15        },
16        "platform": "FACEBOOK"
17      }
18    ]
19  }
```

CLOSE

COPY FULFILLMENT REQUEST AS CURL

COPY RAW RESPONSE

Figure 26: Structure of responses

6.2 WebHooks and Connections

The database of the Ministry of Economic and Social Inclusion of Ecuador is located in Quito, at the central offices. This is centralized and is managed by the IT department. Due to national security and confidentiality policies, data cannot be stored on any private server external to the ministry. This database is an Oracle Database 10g Enterprise Edition. The connection chain and access users and passwords were provided by the ministry for the use of this project. To facilitate the connection processes, we use the Oracle "oracledb" library. With this library we can implement RESTful services through Node.JS and access the information in the database. The URL of our Webhook must be configured in the main Dialogflow configuration screen for later calling. This services will be hosted on a private server of the Ministry of Social Inclusion and are implemented in a NodeJs server, since it is implemented in JavaScript. The structure of the service is organized as follows:

1. Config: It helps us to organize the connection to the Data Base and the processes according to the development environment in which it is selected in the .env configuration.
2. Controller: Each one of the intents is associated with a controller that allows keeping the logic for each selection classified. These controllers also allow us to track the personalized responses that we will create for each one.
3. DB Connector: A specialized layer to separate queries to the database. This intermediate layer separates the information that enters from the database, from the business logic to the webhook controller.
4. Models: To structure the information that must be sent to Dialogflow, in the form of JSON, we organize it into Javascript objects that help us to manipulate it in a better way. This layer also allows us to organize the structure of any response and input media.
5. Services: The main layer of the webhook is organized in Services. Here we organize the routing of requests and dependencies.
6. Utils: Utils are a group of utilities that allow us to organize our business logic in a more optimal way.

Although the static responses of the attempts are essential to start the communication process, Dialogflow enables a fulfillment that allows dynamiting the content of the responses. For this, we must enable the response option in each intent. The fulfillment configurations are found in the section of the same name.

Queries and Responses.

By using the Oraclejs library and making the connection using the user credentials and password provided by MIES, we can perform direct queries using a

simple SQL language. For example, to consult the information of a person and the vouchers in which he is enrolled, we made the following query:

```

1 function connExecute(err, connection)
2 {
3     if (err) {
4         console.error(err.message);
5         return;
6     }
7     sql = " select nombres, apellidos, email, cedula, celular, edad
            , autoriz_humana, turno_bloqueado, valor_copago,
            usuario_crea, observaciones from cci_turnos where cedula =
            :cedula";
8     connection.execute(sql, {}, { outFormat: oracledb.OBJECT }, //
            or oracledb.ARRAY
9
10 function(err, result)
11 {
12     if (err) {
13         console.error(err.message);
14         connRelease(connection);
15         return;
16     }
17     console.log(result.metaData);
18     console.log(result.rows);
19     connRelease(connection);
20     });
21 }

```

Once we have the reply array, we can use functions to create a reply message:

```

1 async function createAnswerVerificarDigito(array) {
2     try {
3         let respuesta = new Array;
4         respuesta.push("Ya hemos validado la informacion")
5         if (array.length>0) {
6             array.forEach(element =>
7             {
8                 respuesta.push("Tu bono " + element.nombreBono + " de "
9                     + element.bono + " esta listo para ser retirado en
10                     nuestras agencias. \n")
11             });
12             return respuesta;
13         } else {
14             respuesta.push("No tienes ningun bono activo")
15             return respuesta
16         }
17     } catch (err) {
18         next(err);
19     }
20 }

```

To make the necessary queries to the database we have implemented the following controllers:

- Producto (Product): In charge of listing all the benefits registered by user;
- Seleccionar_ciudad (choose city): In charge of listing the collection locations that the ministry has provided by city;
- Seleccionar_documento (choose document): In charge of listing the procedures and descriptions that the MIES performs online;
- Seleccionar_programa (choose program): In charge of listing the programs and descriptions in which the MIES currently works;
- Verificar_Digito (validate ID): In charge of validating the user by using their ID and the last digit of their validator number.

Each of these functions also has a function in charge of assembling the response that will be delivered through the webhook to Facebook for its presentation to the user.

6.3 Integration with Facebook Developers

Developers who create applications can install the Facebook SDK in their application to increase customer interaction and obtain a great advance of functionalities. In Figure 27, we can see the application created on the Facebook Developers platform, prior to the initial configuration. First, we must access the Facebook Developers environment at <https://developers.facebook.com/>. For that, we must have an account created and access it. It is not important to create a corporate account.

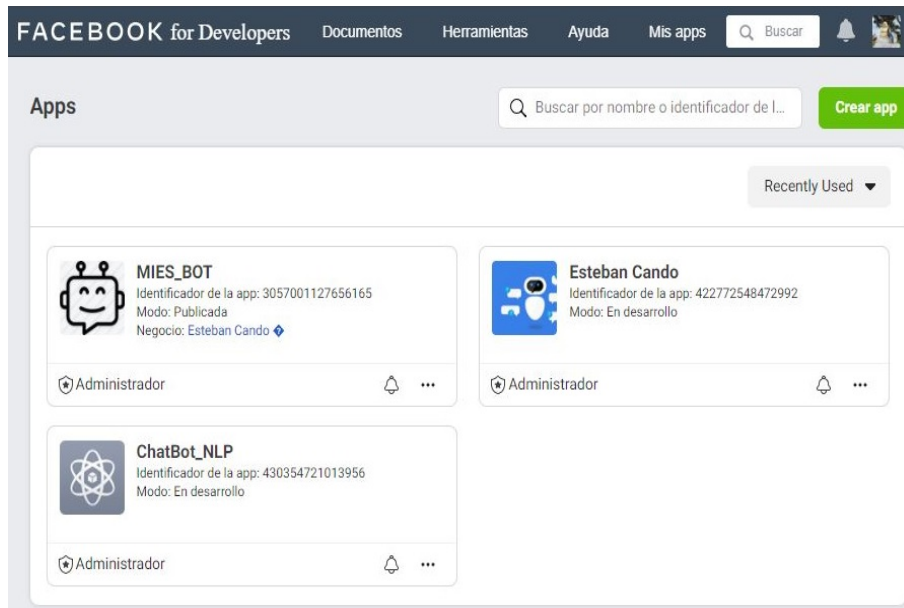


Figure 27: Facebook Apps configuration

6.3.1 Access Tokens

Once we have created an application, and after giving a name to the Facebook Application, a contact email and once verified that it is on the development stage, we continue with the Messenger configuration. In the Access Tokens section, we need to select the Facebook page in which the agent will be working. This will generate the token that we need to add into Dialogflow configuration.

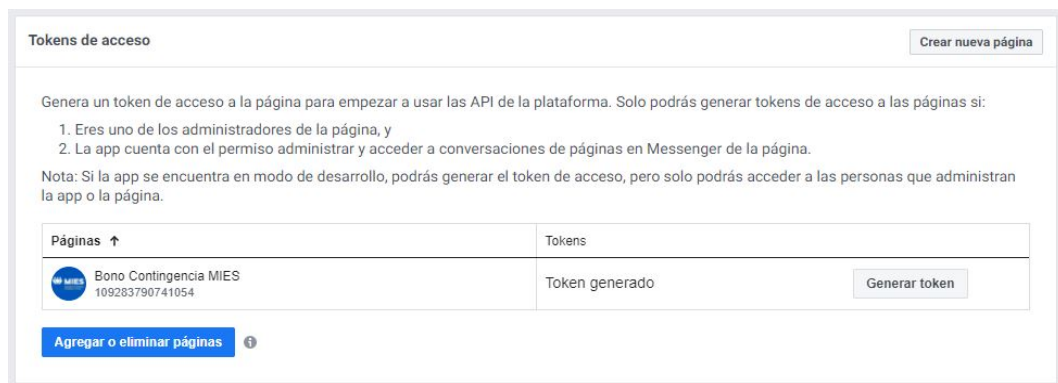


Figure 28: Basic Configuration on Facebook App

In the webhook section of the Messenger, as it is shown in Figure 29, we add

request access to some important events:

- **messages:** The general content of the message entered through Facebook Messenger.
- **messaging_postback:** The response sent by the agent using the webhook.
- **message_deliveries:** Flag of messages sent from Facebook Messenger.
- **message_reactions:** Reactions sent in the messages from Facebook Messenger.
- **messaging_optins:** Event that marks the acceptance of a message request by the user.

Editar suscripciones de la página ✕

 **Bono Contingencia MIES**
109283790741054

Campos de suscripción

☒ messages	☒ messaging_postbacks	☒ messaging_optins
☒ message_deliveries	☐ message_reads	☐ messaging_payments
☐ messaging_pre_checkouts	☐ messaging_checkout_updates	☐ messaging_account_linking
☐ messaging_referrals	☐ message_echoes	☐ messaging_game_plays
☐ standby	☐ messaging_handovers	☐ messaging_policy_enforcement
☒ message_reactions	☐ inbox_labels	

[Más información](#) Cancelar Guardar

Figure 29: Subscriptions services from Facebook Messenger

Once we have created the application, we can configure it to connect it to our agent in Dialogflow. In Dialogflow we only have to activate the integration with Facebook and keep the URL response information of the webhook and the confirmation token. This will serve us later.

 **Facebook Messenger**
An easier way to message.

Create and teach a conversational bot for Facebook Messenger.

After you design and test your Dialogflow agent, you can launch your Messenger bot

1. Get your Facebook Page Access Token and insert it in the field below.
2. Create your own Verify Token (can be any string).
3. Click 'START' below.
4. Use the Callback URL and Verify Token to create an event in the Facebook Messenger Webhook Setup.

[More in documentation.](#)

☒ Show old callback URL (more details [here](#))

Callback URL	https://bots.dialogflow.com/facebook/c5abb35f-9d4f-487c-b8b5-1ff54327d283/webhook
Verify Token	BOTMIESEC2020
Page Access Token	EAArcU4pKuuUBAIZErZAcVMsDFerSACvuoYpowf21EsFKn58i4G5CgKAH4dOXV0FSWZA

Active environment: 

STOP

Figure 30: Basic Configuration on Facebook App

In the webhook configuration in Facebook developers we must place the response URL and the security token. After verification, our agent is able to receive responses from Facebook, but only from the development team that owns the application. To be active for all users we must validate the application with the Facebook team.

It is important that the request access to the messaging service will be active. For that, we need to select the Facebook page on which the bot will be tested, as well as include instructions for use and additional comments for the Facebook review team to test our assistant. This is part of the process to activate the agent. Normally, it takes one or two days to the review and we need to add all the information that Facebook asks to review. If the Facebook team has any comments or needs to review the information in greater detail, they contact through the email added when the application was created.



Figure 31: Basic Configuration on Facebook App

To verify the connection, as it is shown in Figure 32, we open the Facebook page which the agent is associated to. In the message box the text input will be blocked, but we must have a button called: "Start" or "Empezar" in Spanish. Then we can start the conversation with our Agent, as shown in Figure 33, and all the interaction will be passed to Dialogflow.



Figure 32: Start call from Facebook Messenger



Figure 33: Response test from Facebook Messenger

6.3.2 Analytics

To verify response times, errors, number of active users or messages sent, we have two types of tools available. Facebook has a group of tools called Facebook Analytics. Although together they have endless variables and analytics, we can focus on the Statistics API. This will help us to know the number of calls that the agent has received from Facebook Messenger, the number of errors he has received back and the average request time. To validate the response in the agent, we use the Dialogflow Analyst.

The Dialogflow console provides a statistics page that displays various agent response and request data request statistics. This data helps us to evaluate how the agent is used in production. Our main statistics will be two: Interactions and sessions. These statistics will be reviewed in Chapter 7 "Agent evaluation".



Figure 34: Statistics from Facebook App

6.4 Agent Training

To begin the agent training in Dialogflow, we use the input data to create a machine learning model. Typically supplying training data by directly entering training phrases to intents starts the learning process. We can also use the training tool to analyze, import and export real conversation data and improve our training data. The training runs automatically every time we save or update the agent. Once the process is complete, the training status is displayed as a screen notification. we should always wait until the training is finished before testing our agent. If an agent has more than 780 interactions, or if we have disabled the automatic training setting, we must run the training manually.

The training tool is used to review the conversations that the agent has made with messages coming from Facebook to improve our training data of our chatbot. With the tool we can review the interactions made by users and the intentions that matched each turn of the conversation. We can validate or change them if the agent did not interpret correctly. This will help for future conversations. The tool uses data from the agent's conversation history to load the information and detected attempts. The registration of this information must be enabled to use the tool. Only end user expressions can be presented. In order to see all the data of the conversations of the agent and the end user, we must use the agent history.

When you edit the displayed data or click an action button, they create training data update tasks. This information is in a queue waiting to be saved. Once all the conversations have been reviewed and the tasks finished, the changes must be accepted and saved. We can see in the training page a conversation in the training view. and you can also modify the matching entities as highlighted annotations. This will help the model to understand two entities that may be similar. For example, a 10-digit DNI from a 10-digit phone.

As a good practice, Dialogflow recommends using the training tool at various stages of development. Before our agent goes into production, it can be tested with a small group of users. After entering production, the actual conversations must be analyzed and followed to verify the results. Whenever significant changes are made to the agent, we must verify that the new changes behave as expected. The implementation process is simple and fast compared to other technological projects such as the development of mobile applications or webs, since both development frameworks have easy-to-use tools and quick start-up. In addition, Dialogflow and Facebook developers have a wide community of developers that feed databases with their knowledge in forums and free documentation. In our project we have found that training, validating the information and responses that are delivered from users to the chatbot represents a much more detailed and complex process and take the most of the time at the beginning. This process can take a long time and must be thorough if our chatbot is to function properly.

7 Agent evaluation

It is difficult to find an optimal methodology to evaluate an intelligent agent. Much of the information must be interpreted and evaluated according to the parameters presented by the client. To make this evaluation quantitative, we decided to use usability testers that will rate the agent's performance using tasks and actions in the chatbot. Dialogflow and Facebook Analytic also allow us to issue a rating based on parameters. To optimize these evaluations, we will focus on two main points: the technical and technological performance of the agent, and a survey carried out with a group of users to know their evaluation of the chatbot responses and ease of use. The example of the form that we request to answer the users is in Appendix B.

7.1 Analytic Testing

To test the response generated by the agent we have two options: access from the Facebook page where the agent has linked, as we can see on Figure 35, or from the main Dialogflow page as in Figure 36. All the information on actions and transactions generate a statistical report in Facebook Developer, and in Dialogflow Statistics. This information is relevant to analyze the agent quantitatively, and corroborate compliance with the objectives set.



Figure 35: Test from Facebook Messenger

We recollect the information and process it to analyze the chatbot responses. Our principal parameters are:

- Calls;

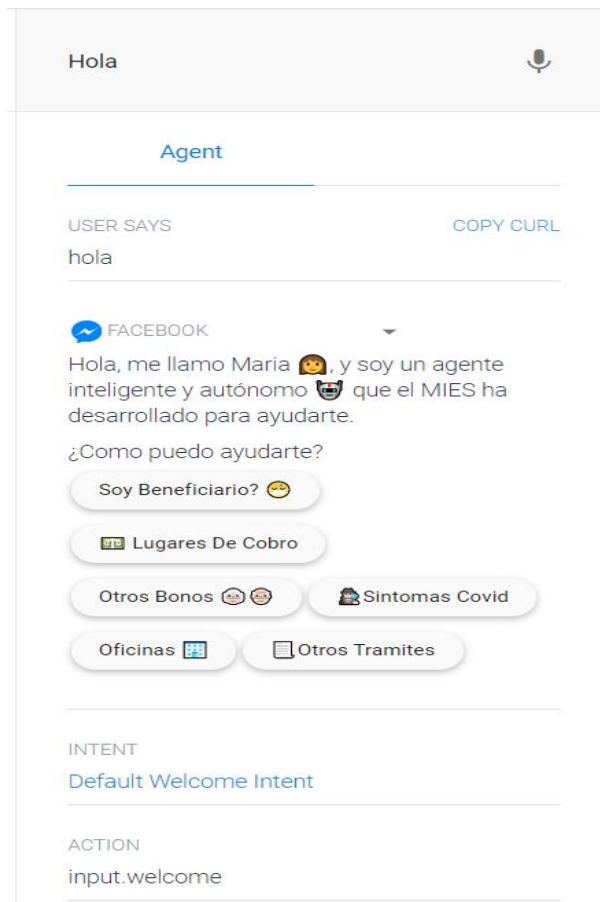


Figure 36: Test from Dialogflow console

- Mistakes;
- Average request time;
- Queries for each Intent;
- Percentage of calls per intent.

To retrieve the values of the calls to the agent, we consult the information of statistics generated in the Facebook developers. This information is filtered by dates and graphs that help us to better understand its meaning and understand it. We have also exported it to study it in detail.

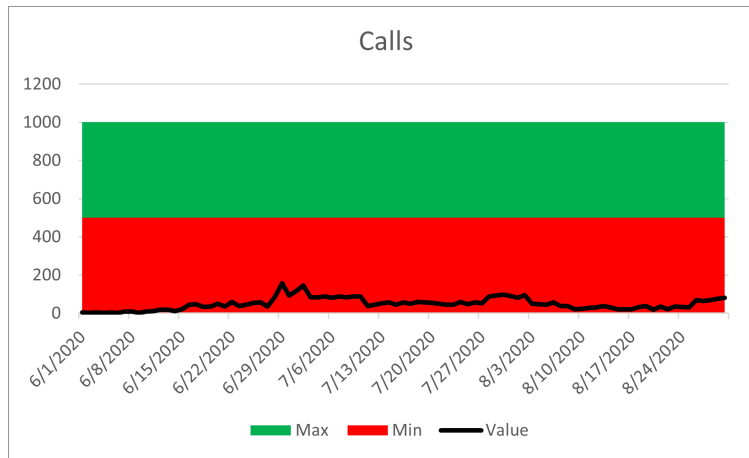


Figure 37: Calls from Facebook Messenger

Calls from Facebook allow us to know the volume of requests made from the main page. The client has requested to have parameters greater than 500 interactions per day to consider a high volume of calls. On average, the customer expects to receive from 500 to 1000 calls a day to consider it an acceptable level. An average of less than 499 calls is considered a low level of calls. Nowadays, we have an average of 50 calls per day, with maximum peaks of 160, as can be seen in Figure 37.

Regarding errors, the customer has considered that these should not exceed 10% of the maximum call quota. Considering 1000 interactions per day, we have a maximum of 100 errors. For our daily average of 50 calls, we have a limit of 5 errors per day, as shown in Figure 38. We must understand that the errors recorded by Facebook do not correspond to errors with the chatbot agent. Facebook considers error to: incorrectly structured responses, responses not delivered, high latency in timeouts, or responses that do not resolve or with code 500.

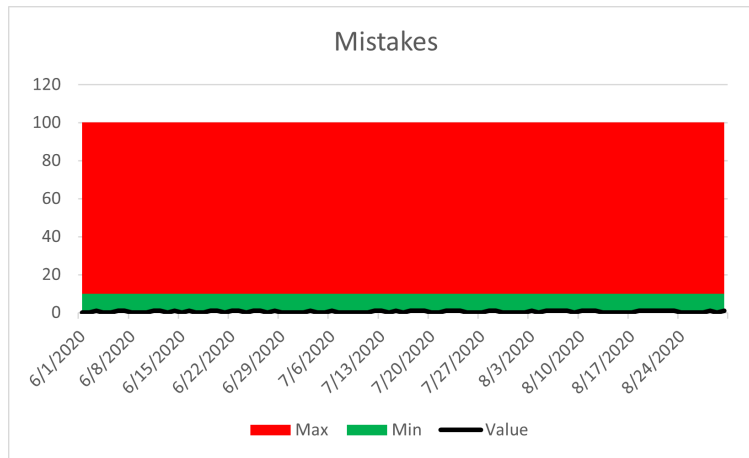


Figure 38: Mistakes from Facebook Messenger

On average, we have 0.4 errors per day, most of them due to connectivity problems. Response times are expected not to exceed the range of 1000 to 2000 milliseconds. Response levels less than 999 milliseconds are considered fast and timely responses. The acceptable tolerance range is 0 to 999 milliseconds. On average, the agent's response was sent in approximately 139 milliseconds after the agent's call, as we can see on Figure 39.

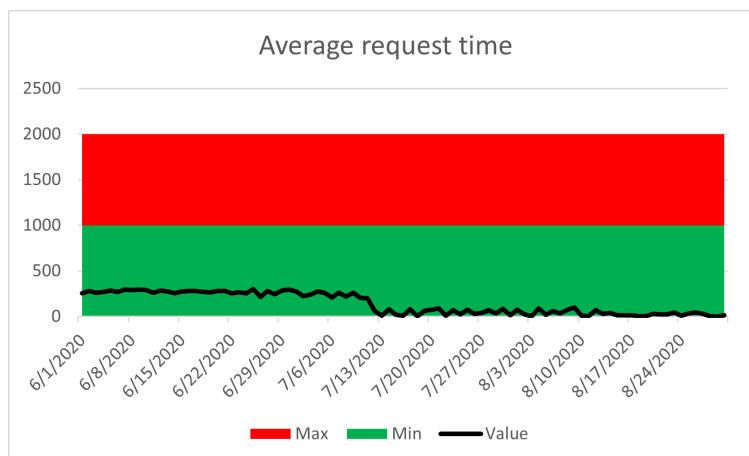


Figure 39: Average request time from Facebook Messenger

To evaluate the agent from the Dialogflow platform, we determine the most used intents, to find out how important they are for users. Then, we can determine which attempts are really useful for them as we can see on Figure 40. We observe that the most consulted intentions correspond to the "Consultation of

beneficiaries" and "Welcome Intent". The two intents represent 45.95 % of the total interactions. The least requested intents are the consultation of "Other bonuses", "Symptoms of covid 19" and "Exit". This allows us to realize that most users do not use the main menu to know all the functions of the chatbot agent, since the user tries to communicate as if it were a person.

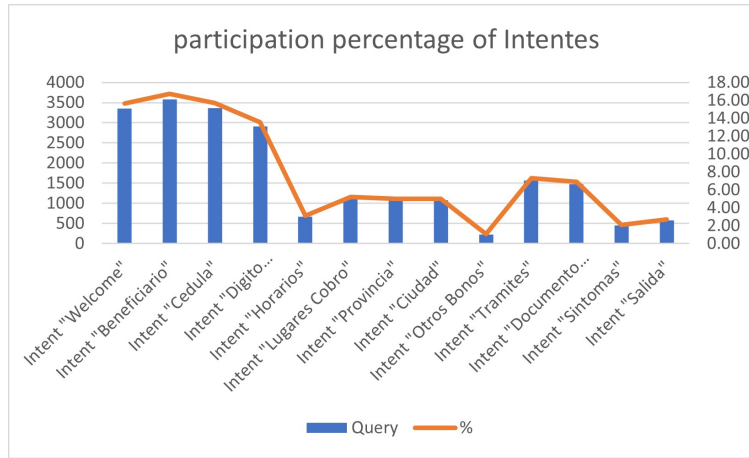


Figure 40: Dialogflow Statistics

7.2 Usability Testing

The usability tests are, according to Usability Professionals Association, a direct feedback to developers by a special group of users, which allow them to find problems in development cycles and reduce production times and costs. Among its main benefits, we find reducing development times and costs of maintaining and modifying a product that is already in production environments and increasing the satisfaction of the first users who use the software in the early stages.[9] For our usability test, we focus on needs, tasks and objectives. This test model includes a moderator who imposes a list of tasks to be carried out, monitoring each action carried out by the user, without intervening in their actions. Based on the methodology proposed by Jakob Nielsen[13] we have selected 20 users of different gender, age, socioeconomic level, and locations of Ecuador. The entire form is anonymous and allows users to express themselves naturally. Before the test, a 20-minute explanatory talk was held, explaining and delivering basic guidelines for use. The evaluation is focused on three aspects:

1. Preparation prior to use;
2. Guided tasks and requests;
3. Recommendations and feedback.

Preparation prior to use These questions are focused on preparing the user and knowing a little more about the users. We want to know the characteristics of the users who are going to be interviewed and be able to determine if these affect the performance when using it. The basic questions to determine their characteristics have been developed based on the target group of the client and how this can influence the agent's behavior. Each user was assigned specific tasks to be carried out using the chatbot for their resolution. These tasks corresponded to finding information about, collection places, information about other MIES programs, recommendations for Covid 19, information about online procedures and hours of the main offices. These tasks are the basic functionalities of the agent and they allowed us to know the capacity of this in solving doubts. In the end, the additional recommendations that the users could suggest were also collected, as well as the observations that we could obtain by being viewers of the processes while the users interacted with the chatbot. An example of the test that was carried out is shown in Figure 41. The questions asked were:

1. Age
2. Gender
3. Last academic level reached
4. Place of residence
5. Rating from 1 to 5. How much do you use Facebook to consult information that you require?
6. Rating from 1 to 5. How familiar are you with Chatbots?
7. Rating from 1 to 5. How pleasant was your experience with Chatbots been on other pages?

Test de usabilidad Chatbot MIES

Esta encuesta esta desarrollada para evaluar múltiples aspectos de uso y rendimiento del Agente Inteligente integrado a la página de Facebook del Ministerio de Inclusión Económica y Social

1. Valorando del 1 al 5. ¿Cuánto usa Facebook para consultar información que usted requiera?

1

2

3

4

5

No busco informacion en Facebook

☐

☐

☐

☐

☐

Busco todo a traves de Facebook

2. Valorando del 1 al 5. ¿Qué tan familiarizado se encuentra con los Chatbots?

1

2

3

4

5

No posea ningun conocimiento

☐

☐

☐

☐

☐

Los utilizo todo el tiempo

Figure 41: Usability test form example

Guided tasks and requests

Each user has been given a random person from the database to interact and obtain information. The main objective of these questions is focused on quantitatively determining the agent's efficiency in response times, the ability to obtain information and the ease of use of the agent's three main functionalities. The questions asked were:

- Considering the previous information provided of personal information of the test user, an identity card. Identify if he has a Human Development economic bonus and answer the following questions.
 - Rating from 1 to 5. Did you get the requested information?
 - Rating from 1 to 5. How long did it take you to obtain the requested information in the task?.
 - Rating from 1 to 5. Was it easy to access the requested information in the task?.
 - Rating from 1 to 5. Did the agent understand your questions and answers provided?
- Considering the previous information provided of personal information of the test user and the place of residence. Identify the closest place to his residence to collect the economic bonus and answer the following questions.
 - Rating from 1 to 5. Did you get the requested information in the task?.

- Rating from 1 to 5. How long did it take you to obtain the requested information in the task?.
 - Rating from 1 to 5. Was it easy to access the requested information?
 - Rating from 1 to 5. Did the agent understand your questions and answers provided?
3. Considering the previous information provided of programs of the Ministry of Social and Economic Inclusion of Ecuador. Request information about the programs and answer the following questions.
- Rating from 1 to 5. Did you get the requested information in the task?.
 - Rating from 1 to 5. How long did it take you to obtain the requested information in the task?.
 - Rating from 1 to 5. Was it easy to access the requested information?
 - Rating from 1 to 5. Did the agent understand your questions and answers provided?
4. Considering the previous information provided of the projects in which the Ministry of Social and Economic Inclusion of Ecuador works. Request information about all programs and answer the following questions.
- Rating from 1 to 5. Did you get the requested information in the task?.
 - Rating from 1 to 5. How long did it take you to obtain the requested information in the task?.
 - Rating from 1 to 5. Was it easy to access the requested information?
 - Rating from 1 to 5. Did the agent understand your questions and answers provided?
5. Considering the previous information provided of symptoms and recommendation about Covid 19. Request information about Covid and answer the following questions.
- Rating from 1 to 5. Did you get the requested information in the task?.
 - Rating from 1 to 5. How long did it take you to obtain the requested information in the task?.
 - Rating from 1 to 5. Was it easy to access the requested information?
 - Rating from 1 to 5. Did the agent understand your questions and answers provided?
6. Considering the previous information provided of offices and customer service hours of the Ministry of Social and Economic Inclusion of Ecuador and its main offices. Find information about business hours answer the following questions.

- Rating from 1 to 5. Did you get the requested information in the task?.
- Rating from 1 to 5. How long did it take you to obtain the requested information in the task?.
- Rating from 1 to 5. Was it easy to access the requested information in the task?.
- Rating from 1 to 5. Did the agent understand your questions and answers provided?

Recommendations

This section of the form helps us to determine problems, conclusions and recommendations that users have encountered when using the agent. This helps us as feedback to improve and implement the suggestions in future versions. The questions asked were: Considering the interaction you received from the chatbot agent, answer the following questions:

- Rating from 1 to 5. Do you think you need a guide prior to use the agent?
- Rating from 1 to 5. Do you think that the agent presents problems when using it?
- What types of problems did you encounter?
- Rating from 1 to 5. Did you find the agent's functionalities useful?
- Rating from 1 to 5. Do you think you need the help of a human agent after using the chatbot?
- Rating from 1 to 5. Would you recommend the agent to your acquaintances or relatives?
- Rating from 1 to 5. What overall rating would you give the agent?
- What additional features would you add to the agent?
- Do you want to submit an additional recommendation?

7.3 Results

After tabulating the responses obtained (All of them are located in appendix C), as we can see on Figure 42, we can see that the group consulted is in an age range of 18 to 65 years, with an average age of 36 years. In Figure 43 we collect the information of the province and city in which our test subjects live. And in Figure 44 the education levels of our test users were tabulated. They all have basic education studies and live in Ecuador. This allows us to validate that they are valid users and within the client's target group.

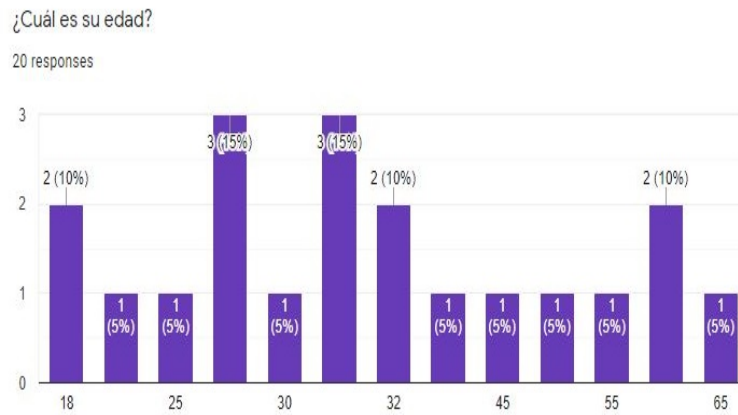


Figure 42: Survey statistic: Age

¿En qué provincia / ciudad reside?

20 responses

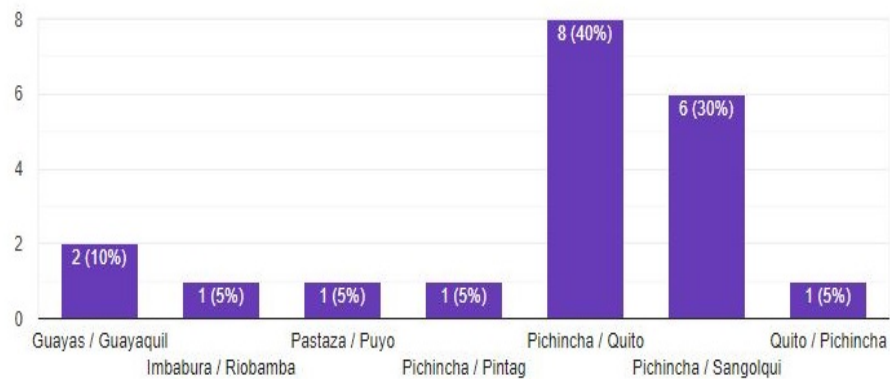


Figure 43: Survey statistic: Residence

¿Cuál es su último nivel de estudios alcanzado?

20 responses

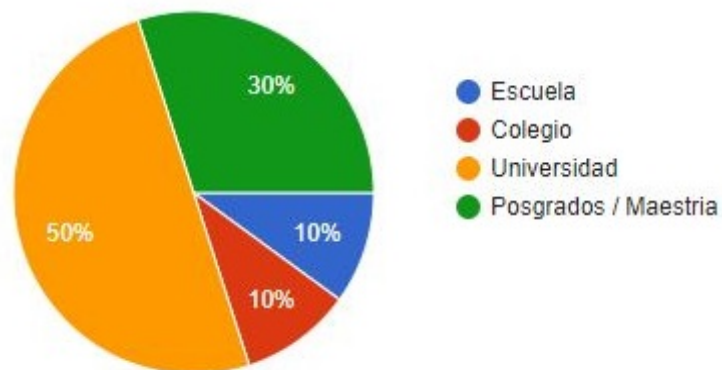


Figure 44: Survey statistic: Education level

Valorando del 1 al 5. ¿Cuánto usa usted Facebook para consultar información que usted requiera.?
20 responses

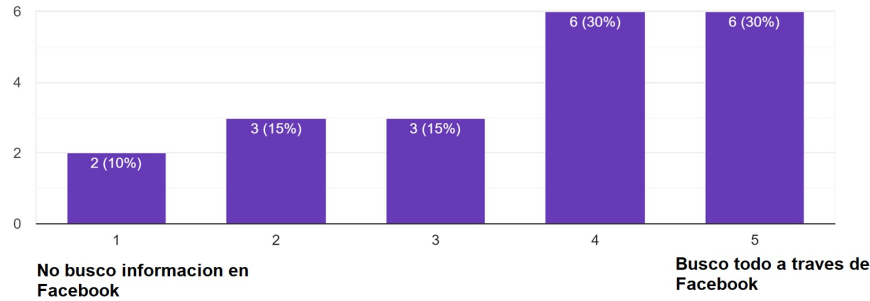


Figure 45: Survey statistic: Facebook as a query tool

Figure 45 shows us that most of them use Facebook as an information query tool. In the range 1 to 5, where 1 means that users do not use Facebook as a search tool and 5 means that users always use Facebook as a search tool. We can conclude that 60% of our test users use Facebook constantly and daily. And they have a basic knowledge of smart agents or chatbots in other tools or web pages. They are also satisfied with the use of these tools.

In the main task requested to identify the economic bonuses that the test user has, as shown in the answers obtained in Figure 46, all the interviewees were able to find the assigned economic bonus information. 90% of users consider that they obtained all the requested information, and the remaining 10% found the information, but are not completely sure if it is complete or requires more detail. When inquiring about response times, 65% of users entered the information in less than 1 minute. This is due to the familiarity that the user had when asking the agent about the required information. It was observed that many at first tried to interact with the options to know about everything they could do before starting with the assigned tasks. These times are in an optimal range and no user showed discomfort with them. 10% of the users consulted state that it was difficult to find the requested information, due to language problems. It was observed that users who had problems with the agent, it was due to incorrectly entering the ID numbers, or confusing the verification number. When asked if the agent understood all their messages, 90% found the chatbot agent capable of fully understanding them.

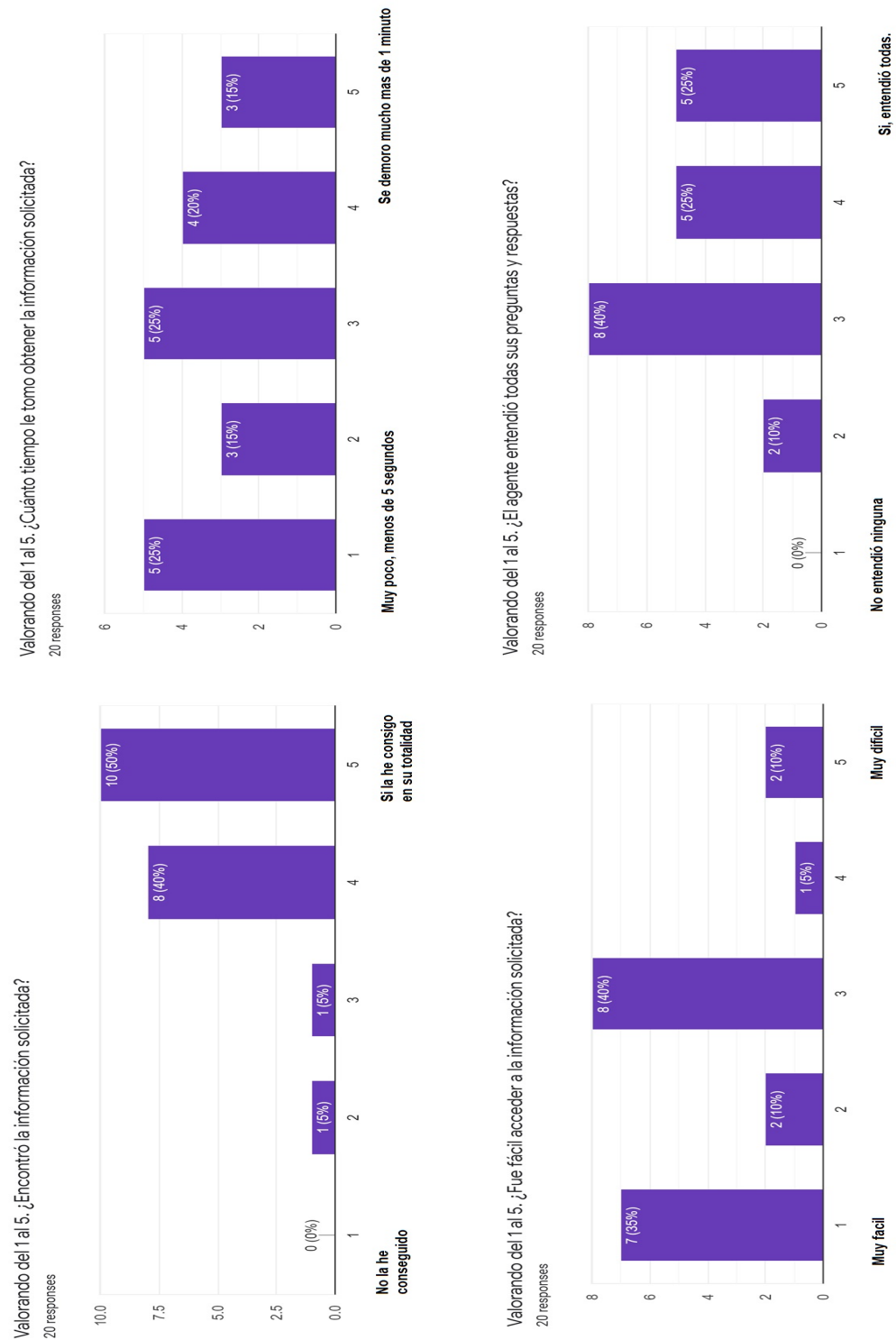
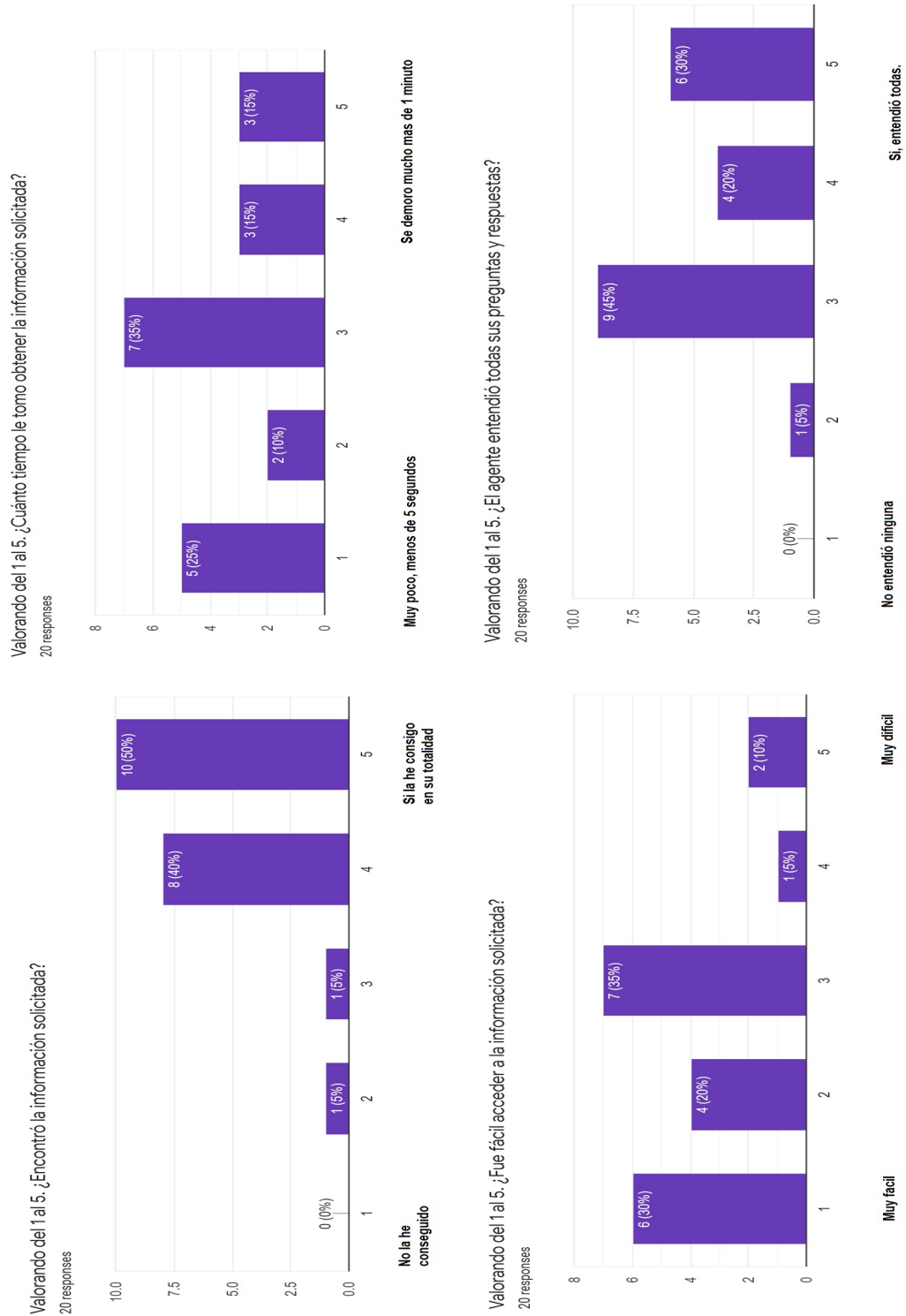


Figure 46: Survey statistic: Human development bonus

In the second section of questions, as shown in Figure 47, when asking participants to find places near the test user's home to collect their bonus money, 100% of respondents were able to find the information and 85% determined that they obtained the information in less than 1 minute. It is observed that there is a perception of difficulty in 15% of the participants when consulting nearby locations. This is due to the use of short names or synonyms to search for a city or province. This will be reduced as the chatbot continues to train and receive city names and associate them with the knowledge base. Only 5% showed difficulties communicating with the chatbot, due to the same issues detailed above.

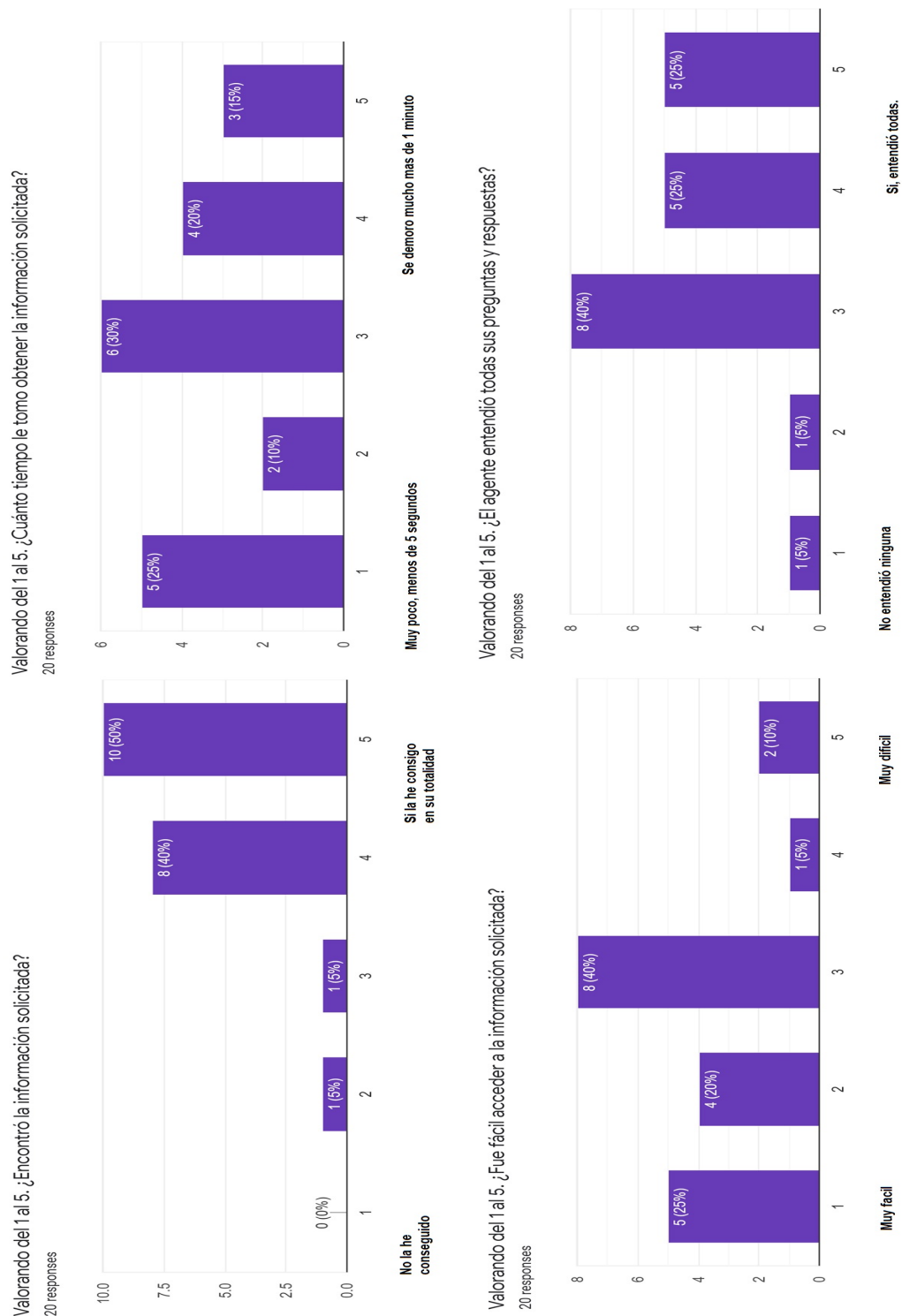
The next task assigned to users was to search for information on the programs carried out by the Ministry of Economic and Social Inclusion of Ecuador. The responses were tabulated in Figure 48. The response was also positive. 100% of users found the requested information, but 5% faced slight problems when searching. It was observed that most of them tried to investigate more in depth as requirements or try to register for such projects. These functionalities are not enabled, but I mention them to the client to take them into account in future versions. Due to being undeveloped functions, the chatbot agent sometimes could not understand all your questions. 35% of those surveyed experienced delays when searching for information and it took them on average more than 1 minute to find it. This is due to the previously mentioned problem.



Test users were asked to perform exploration tasks with minor intentions such as: "MIES project information", "Covid 19 symptoms", "Hours of attention". The first task was to search for information on all the projects that the ministry is currently carrying out. The answer was positive but we found a training problem. All users managed to find the information in less than 1 minute, but showing that at times the agent could not understand all their questions. This difficulty radiates from the fact that the agent does not receive many interactions and therefore his training is at lower levels compared to the attempts of "Welcome" "Economic Bonds" or "Places of collection of economic bonds". If an agent cannot be trained by users, it will always present interpretation problems. It has been decided to give more time and validation to this intention.

Regarding the tasks requested for the "Covid Symptoms" intent, many users experienced functionality confusion. Users when interacting with the chatbot sent a list of symptoms and hoped to receive a classification to help them determine if COVID cases were possible. This was not part of the intent functions. The goal was to provide basic information and simple recommendations. This intent will be modified and adapted to make it a simple triage, but notifying that this information does not replace the use of validation tests to detect COVID 19.

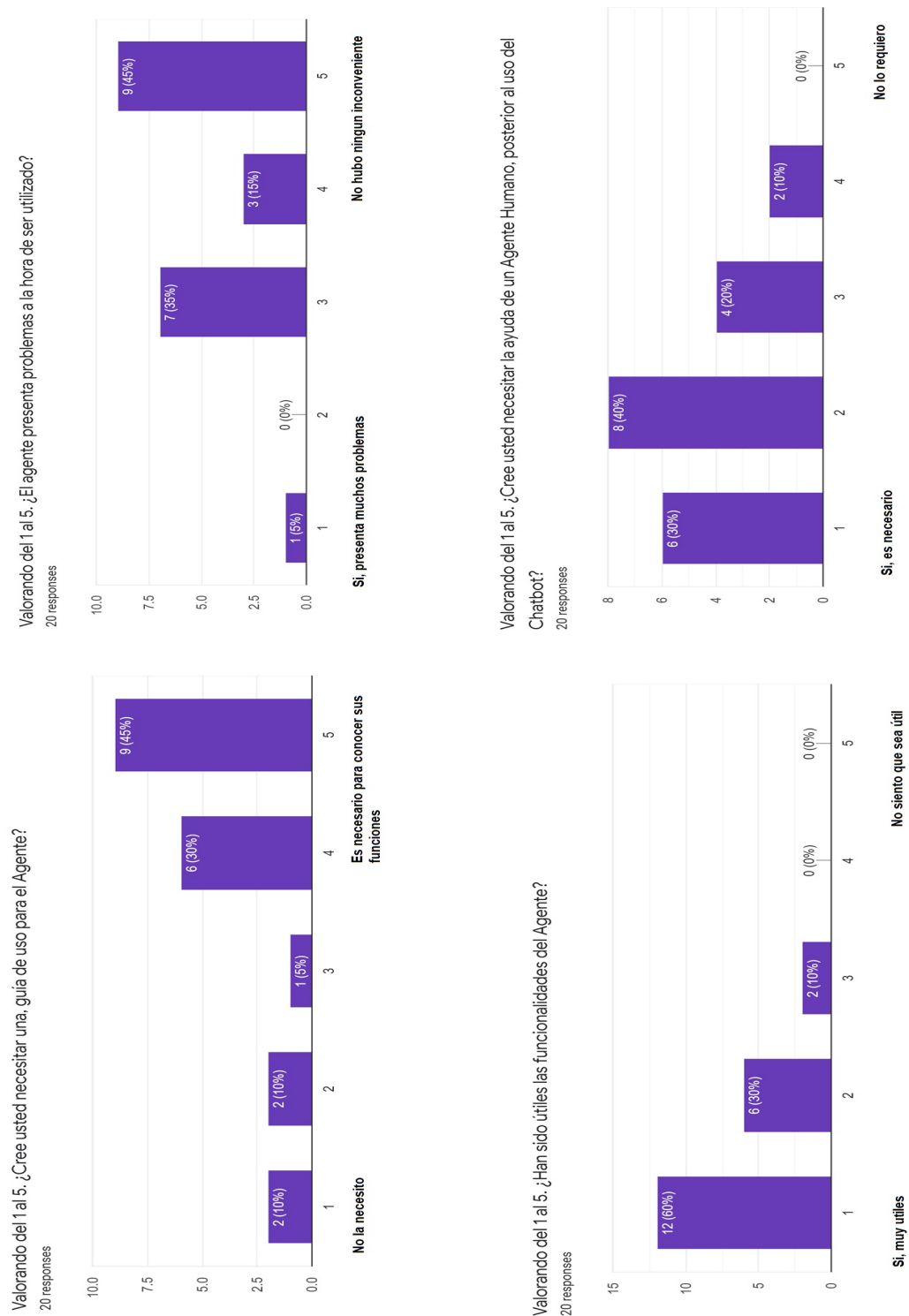
The last of the tasks requested the users was to consult information on hours and locations of offices of the Ministry of Social and Economic Inclusion of Ecuador. This attempt was well received by users, but in the feedback the vast majority also asked to be able to carry out scheduling processes or consult information on specific dates such as Christmas, Labor Day, among other holidays in which we do not have a normal schedule.



81
Figure 48: Survey statistic: Other programs

The last section allowed users to validate the general usefulness and ease of use of the chatbot agent, as well as recommendations and comments that were compiled in Figure 49. Most users recommend creating a guide or tutorial to learn about all the functions of the chatbot agent. The first question was focused on knowing if the user perceived a need for a user manual or tutorial in order to understand the functions of the chatbot. 80% of users believe they need additional help to understand how it works. This is because users have a certain level of resistance to automatic systems. When asked if they had difficulty in the activities, many responded negatively, but still preferred to request a manual to better understand the chatbot. In general, 95% of the test users affirm that they had no major problem when using the chatbot, and 100% of the users responded that they found it useful or very useful.

To our surprise, 100% of the users responded that they would need to interact with a human person after using the chatbot. 30% of test users say that they will always need the help of a human agent no matter how good their experience with the chatbot was. To change this trend, the ministry must implement socialization and training programs that help citizens to understand the functioning of intelligent agents and allow technology leaps.



83

Figure 49: Survey statistic: recommendations

The final rating they gave the chatbot agent is very positive. As seen in Figure 50, all test users feel a high level of acceptance. The average grade is very good, since they were able to get the most responses in optimal times and in the right way. We can conclude that the usefulness of the chatbot agent and its functionalities are ideal for this first version and will help to minimize the work times of the information and marketing area, but do not demarcate from the need to consult or interact with human personnel. All surveyed users agree that they need to or would require to speak with a human being to resolve their doubts at some point. This evaluation has allowed us to know the strengths and weaknesses of a chatbot agent in production. Our chatbot is currently in production and all these feedbacks and suggestions are being analyzed in order to develop new functionalities for future versions.

Valorando del 1 al 5. ¿Qué calificación entregaría al Agente Chatbot?

20 responses

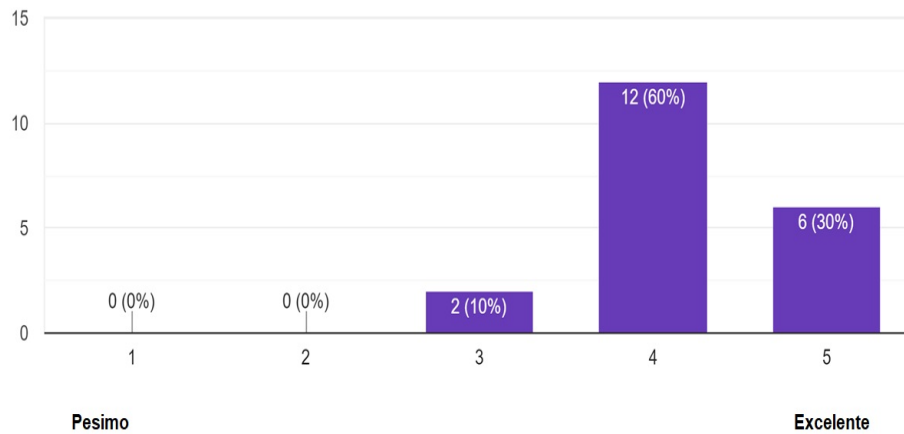


Figure 50: Survey statistic: Agent Qualification

8 Conclusion and future work

The project to create an intelligent conversational agent to assist users in public institutions of Ecuador has allowed us to know more in depth the true strengths and weaknesses that we will go through when developing, training and analyzing an intelligent agent chatbot. Validating and analyzing each requirement to know the interaction flows that we want to obtain will allow us to adapt a correct and efficient architecture that will allow the chatbot to grow in the future. Correct training will be reflected at the time of testing with users and it is convenient to take usability tests, even if the product is already in production for validate that the chatbot is performing its functions correctly.

The frameworks used in this project, Dialogflow and Facebook Developers, are robust and provide all the necessary tools to create an enterprise-level chatbot agent. Even in its most limited level of resources, it does not have any type of limitation in characteristics or functions. The free tier used for both environments has had no major limitation and can be scalable as the number of users who use our chatbot continue to increase.

The three-layer structure used when implementing the solution is complete and scalable for the capacity of users who have used it. Nowadays, it has endured an average of 1000 weekly users interactions and growing. The purpose of using a scalable architecture was to sustain an average of 100,000 weekly interactions, and it is estimated that by the end of the year 2020 this figure will be reached without much difficulty. It has also been very helpful to use the NLP technology provided by Facebook and Dialogflow to facilitate the agent's processes of identifying user intentions. The processes of training and feeding its knowledge base has been the biggest obstacle to face since Spanish is a language very rich in variants and synonyms.

Publishing a chatbot agent, using Facebook Developers is a simple process, but not as fast as we would expect. Facebook has many verification channels that prevent and filter creating malicious applications. This helps the final product delivered to the user to be of quality and to build confidence towards its use. This quality and validation of applications by Facebook generates confidence in the use and delivery of information. No user was suspicious of handing over her information or answering identification questions since they trust on Facebook.

The analytics generated by Facebook and Dialogflow reveal optimal response times to maintain a fluid and smooth conversation. But users do not neglect human interaction. After using the chatbot agent, most of the users who carried out the tests consider it appropriate to have a dialogue with a human agent. This is linked to the need to interact with people.

We can conclude that the use of a chatbot agent significantly helps users to find information and carry out personal validation processes, avoiding travel times, reducing the operational load of the marketing and information depart-

ment of the Ministry of Social and Economic Inclusion of Ecuador, but it does not replace the need of users to interact with human personnel. This can be helped to improve by including information plans and inviting users to use these new technological tools. The Marketing department of the Ministry of Social and Economic Inclusion of Ecuador must implement and promote the use of these new technologies in users, however, the need for users to interact with support and help personnel can never be avoided.

8.1 Future Work

After talking with the Marketing Department and presenting the observations that have been made in this first version of the chatbot to help their users, future projects have been proposed that they want to include.

It is desired to include improvements in phrases and idioms of the Spanish language to facilitate the use of people of economic resources and low levels of education. It is also proposed to adapt the chatbot to understand other national languages such as Quichua, Quecha and Shuar. Although these languages are not supported in the Dialogflow platform, it is possible to take a base language, for example the same Spanish, and train the chatbot Agent from scratch. That the agent understands Spanish means that it has been previously entered based on this Language, but training it from zero would mean the same effort we have made to understand synonyms and cultural idioms of Ecuador. The effort required to start a chatbot Agent from scratch, in cases of unsupported languages, is high and the impact on training could extend to months, but the final product would maintain the same quality as a language included in the platform.

Other functionality that is desired to include is processes for requesting documents and digital certificates. For this, we must include in our webhook functions that connect to multiple bases which are outside the Ministry of Social and Economic Inclusion, and for example obtain the information from the Civil Registry of Ecuador to validate and generate certificates. We should also set up Digital Signature and Digital Certificate functions. This effort is a joint task between ministries and relationships between them. This process of interaction has begun and it is expected to obtain a positive response since at the government level it is desired to create an integrated model that does not exist at present. The biggest problem in this functionality lies in the capacity for cooperation between ministries, and unfortunately we are close to ending the government period and if it is not concluded before the end, they must start from the beginning when new leaders are elected next February 2021.

And we have already started work on a Covid 19 identifier and diagnostician based on the symptoms presented by the users. A basic process of symptom identification and triage is being created using the information shared by the Ministry of Public Health of Ecuador and the World Health Organization. This is not intended to replace the use of physical tests to identify patients with COVID 19. Its function is to help citizens better understand and identify diseases such as the common flu that could be confused with COVID 19.

References

- [1] Mohamed Ally. *Mobile learning: Transforming the delivery of education and training*. Athabasca University Press, 2009.
- [2] Liesl Jane Beilby, John Zakos, and Glenys Anne McLaughlin. *Chatbots*. US Patent 8,630,961. Jan. 2014.
- [3] Steven Bird, Ewan Klein, and Edward Loper. *Natural language processing with Python: analyzing text with the natural language toolkit*. ” O’Reilly Media, Inc.”, 2009.
- [4] Héctor Alberto Botello-Peñaloza. “Determinantes del acceso al internet: Evidencia de los hogares del Ecuador”. In: *Entramado* 11.2 (2015), pp. 12–19.
- [5] Jack Cahn. “CHATBOT: Architecture, design, and development”. In: *University of Pennsylvania School of Engineering and Applied Science Department of Computer and Information Science* (2017).
- [6] Menal Dahiya. “A tool of conversation: Chatbot”. In: *International Journal of Computer Sciences and Engineering* 5.5 (2017), pp. 158–161.
- [7] Marie-Claire Jenkins et al. “Analysis of user interaction with service oriented chatbot systems”. In: *International Conference on Human-Computer Interaction*. Springer. 2007, pp. 76–83.
- [8] Pan Juin Yang Jonathan, Chun Che Fung, and Kok Wai Wong. “Devious chatbots-interactive malware with a plot”. In: *FIRA RoboWorld Congress*. Springer. 2009, pp. 110–118.
- [9] Titti Kallio, Anne Kaikkonen, et al. “Usability testing of mobile applications: A comparison between laboratory and field testing”. In: *Journal of Usability studies* 1.4-16 (2005), pp. 23–28.
- [10] Mari-Carmen Marcos. “HCI (human computer interaction): concepto y desarrollo”. In: *El profesional de la información* 10.6 (2001), pp. 4–16.
- [11] Bernard Marr. “Is Artificial Intelligence dangerous? 6 AI risks everyone should know about”. In: *Forbes* (2018).
- [12] David Mimno. *Machine Learning with MALLET*. 2004.
- [13] Jordi Grau Moracho. “Pensando en el usuario: la usabilidad”. In: *anuario thinkEPI* 1.1 (2007), pp. 172–177.
- [14] Patricio Javier Pérez Vallejo. “Implementación de un agente conversacional para negocio de repuestos automotrices integrado a plataformas de mensajería instantánea”. B.S. thesis. PUCE-Quito, 2019.
- [15] Gualtiero Piccinini. “Turing’s rules for the imitation game”. In: *Minds and Machines* 10.4 (2000), pp. 573–582.
- [16] Bayan Abu Shawar and Eric Atwell. “Chatbots: are they really useful?” In: *Ldv forum*. Vol. 22. 1. 2007, pp. 29–49.

- [17] Heung-Yeung Shum, Xiao-dong He, and Di Li. “From Eliza to XiaoIce: challenges and opportunities with social chatbots”. In: *Frontiers of Information Technology & Electronic Engineering* 19.1 (2018), pp. 10–26.
- [18] Fundación Telefónica. *Sociedad digital en España 2018*. Fundación Telefónica, 2019.
- [19] Sandra Vázquez Toledo et al. “Design Thinking como metodología auténtica y transformadora de la educación universitaria”. In: *Book of abstracts CIVINEDU 2018: 2nd International Virtual Conference on Educational Research and Innovation*. Adaya Press. 2018, p. 115.
- [20] Darius Zumstein and Sophie Hundertmark. “Chatbots –An interactive technology for personalized communication, transactions and services.” In: *IADIS International Journal on WWW/Internet* 15.1 (2017).

A Tables of Requirements

Scene Name	As a user I want to know if I am a beneficiary of the "Bono de Desarrollo"
Background	The user is interacting with the bot
Scenario	User requests information about "Bono de Desarrollo"
When the user asks for "Bono de Desarrollo". Then the bot should ask for the ID of user. Then the bot should ask for the validate number of ID. Finally, if ID its valid, give information of "Bono de Desarrollo".	

Table 2: Requirements: Feature 1, scene 1.1

Scene Name	User requests information about "Bono de Desarrollo" but gives wrong validate number of the ID
Background	The user is interacting with the bot
Scenario	User requests information about "Bono de Desarrollo"
When the user asks for "Bono de Desarrollo". Then the bot should ask for the ID of user. Then the bot should ask for the validate number of ID If validator ID is wrong, the bot informs the user And ask again for the validator ID.	

Table 3: Requirements: Feature 1, scene 1.2

Scene Name	User requests information about "Bono de Desarrollo" but give wrong number of the ID
Background	The user is interacting with the bot
Scenario	User requests information about "Bono de Desarrollo"
When the user asks for "Bono de Desarrollo". Then the bot should ask for the ID of user. If ID number is wrong, the bot informs the user and ask again for the ID number.	

Table 4: Requirements: Feature 1, scene 1.3

Scene Name	As a user I want to know the available offices
Background	The user is interacting with the bot
Scenario	User requests information about places to collect money
When the user asks for "Lugares de Cobro". Then the bot should ask for the state closest to user place of residence. Then the bot show a list of cities on chosen state and ask for one of them. Finally show the Office and the daily available hours on the chosen city.	

Table 5: Requirements: Feature 2, scene 2.1

Scene Name	As a user I want to know office on my city, but its not listing
Background	The user is interacting with the bot
Scenario	User requests information about places to collect money
When the user asks for "Lugares de Cobro" Then the bot should ask for the state closest to user place of residence Then the bot show a list of cities on chosen state and ask for one of them If User give a not listed city, said its not on list. And ask again for one listed before.	

Table 6: Requirements: Feature 2, scene 2.2

Scene Name	As a user I want to know office on my state, but its not listing
Background	The user is interacting with the bot
Scenario	User requests information about places to collect money
When the user asks for "Lugares de Cobro" Then the bot should ask for the state closest to user place of residence If User give a not listed state, said its not on list. And ask again for one nearest state on the list.	

Table 7: Requirements: Feature 2, scene 2.3

Scene Name	As a user I need information about programs that MIES is working.
Background	The user is interacting with the bot
Scenario	User requests information about programs
When the user asks for "Otros programas" Then the bot show a list of all programs and ask for one Then the bot show a list of requirements if user need more information give the MIES contacts	

Table 8: Requirements: Feature 3, scene 3.1

Scene Name	As a user I choose one program that its not on the list
Background	The user is interacting with the bot
Scenario	User requests information about programs
When the user asks for "Otros programas" Then the bot show a list of all programs and ask for one if user choose other program said that its not listen and ask again	

Table 9: Requirements: Feature 3, scene 3.2

Scene Name	As a user I choose one program that its not on the list
Background	The user is interacting with the bot
Scenario	User requests information about programs
When the user asks for "Otros programas" Then the bot show a list of all programs and ask for one if user choose other program said that its not listen and ask again	

Table 10: Requirements: Feature 3, scene 3.2

Scene Name	As a user I ask for information about Covid
Background	The user is interacting with the bot
Scenario	User requests information about Covid
When the user asks for "Informacion Covid" Then the bot show for list of sintoms or recommendation if user choose sintoms, show the list of sintoms if user choose one sintom, show recommendation	

Table 11: Requirements: Feature 4, scene 4.1

Scene Name	As a user i ask for recommendation about Covid
Background	The user is interacting with the bot
Scenario	User requests recommendation about Covid
When the user asks for "Informacion Covid" Then the bot show for list of sintoms or recommendation if user choose recommendation, show the list of recommendation if user said more information, show information of contacts	

Table 12: Requirements: Feature 4, scene 4.2

Scene Name	As a user I ask for information about Covid
Background	The user is interacting with the bot
Scenario	User requests information about Covid
When the user asks for "Informacion Covid" Then the bot show for list of sintoms or recommendation if user choose sintoms that its not listed, tell that is not sintom but show information of contacts.	

Table 13: Requirements: Feature 4, scene 4.3

Scene Name	As a user I ask for information about Offices
Background	The user is interacting with the bot
Scenario	User requests information about Offices
When the user asks for "Horarios de Atencion" Then the bot show the list of all oficce, and ask for choose one if user choose one office, tell him the available office hours	

Table 14: Requirements: Feature 5, scene 5.1

Scene Name	As a user I ask for information about Offices
Background	The user is interacting with the bot
Scenario	User requests information about Offices
When the user asks for "Horarios de Atencion" Then the bot show the list of all oficce, and ask for choose one if user choose one office that its not on the list Tell him that its wrong and choose again.	

Table 15: Requirements: Feature 5, scene 5.2

Scene Name	As a user I ask for information about procedure
Background	The user is interacting with the bot
Scenario	User requests information about procedure
When the user asks for "Otros tramites" Then the bot show the list of all procedure, and ask for choose one if user choose one procedure , tell the details of that procedure	

Table 16: Requirements: Feature 6, scene 6.1

Scene Name	As a user I ask for information about procedure whom its not on list
Background	The user is interacting with the bot
Scenario	User requests information about procedure
When the user asks for "Otros tramites" Then the bot show the list of all procedure, and ask for choose one if user choose one procedure that is not listed Tell that procedure is not listed, and ask for other one.	

Table 17: Requirements: Feature 6, scene 6.2

Scene Name	When user start the conversation
Background	The user is interacting with the bot
Scenario	User start the conversation
When the user said "Empezar"	
Then the bot start with a greeting, and explain his functionalities	

Table 18: Requirements: Feature 7, scene 7.1

Scene Name	When user said something that is not a funcion on Agent
Background	The user is interacting with the bot
Scenario	Bot don't understand the question
When the user said something that is not on his funtions	
Then the bot explain his funcionalities like a menu, and ask again	

Table 19: Requirements: Feature 8, scene 8.1

Scene Name	When user said something that is not a funcion on Agent
Background	The user is interacting with the bot
Scenario	Bot don't understand the question
When the user said something that is not on his funtions	
Then the bot explain his funcionalities like a menu, and ask again	

Table 20: Requirements: Feature 8, scene 8.1

B Usability Test Form

18/11/2020

Test de usabilidad Chatbot MIES

Test de usabilidad Chatbot MIES

Esta encuesta esta desarrollada para evaluar múltiples aspectos de uso y rendimiento del Agente Inteligente integrado a la página de Facebook del Ministerio de Inclusión Económica y Social

*** Required**

1. ¿En qué provincia / ciudad reside? *

2. ¿Cuál es su edad? *

3. ¿Cuál es su género? *

Mark only one oval.

- ☐ Mujer
☐ Hombre
☐ Prefiero no decirlo
☐ Other: _____

4. ¿Cuál es su último nivel de estudios alcanzado? *

Mark only one oval.

- ☐ Escuela
☐ Colegio
☐ Universidad
☐ Posgrados / Maestria
☐ Other: _____

https://docs.google.com/forms/d/1dr1JAXN0UNrDBeFGossk0nyJhFB_rkI3bYC5jPg7rig/edit

1/7

5. Valorando del 1 al 5. ¿Cuánto usa usted Facebook para consultar información que usted requiera.?

Mark only one oval.

	1	2	3	4	5	
No busco informacion en Facebook	☐	☐	☐	☐	☐	Busco todo a traves de Facebook

6. Valorando del 1 al 5. ¿Qué tan familiarizado se encuentra con los Chatbots?

Mark only one oval.

	1	2	3	4	5	
No posea ningun conocimiento	☐	☐	☐	☐	☐	Los utilizo todo el tiempo

7. Valorando del 1 al 5. ¿Qué tan agradable ha sido su experiencia con Chatbots en otras páginas?

Mark only one oval.

	1	2	3	4	5	
desagradable totalmente.	☐	☐	☐	☐	☐	Excelente, los recomiendo.

Bono de
Desarrollo
Humano

Utilizando la información entregada de usuario y el número de identificación: Consultar si dicho usuario posea un bono económico y responder las siguientes preguntas:

8. Valorando del 1 al 5. ¿Encontró la información solicitada?

Mark only one oval.

	1	2	3	4	5	
No la he conseguido	☐	☐	☐	☐	☐	Si la he consigo en su totalidad

9. Valorando del 1 al 5. ¿Cuánto tiempo le tomo obtener la información solicitada?

Mark only one oval.

	1	2	3	4	5	
Muy poco, menos de 5 segundos	☐	☐	☐	☐	☐	Se demora mucho mas de 1 minuto

10. Valorando del 1 al 5. ¿Fue fácil acceder a la información solicitada?

Mark only one oval.

	1	2	3	4	5	
Muy facil	☐	☐	☐	☐	☐	Muy dificil

11. Valorando del 1 al 5. ¿El agente entendió todas sus preguntas y respuestas?

Mark only one oval.

	1	2	3	4	5	
No entendió ninguna	☐	☐	☐	☐	☐	Si, entendió todas.

Lugares
de
Cobro

Utilizando la información entregada de usuario y lugar de residencia: Consultar si dicho usuario pose un lugar cercano a su domicilio y responder las siguientes preguntas:

12. Valorando del 1 al 5. ¿Encontró la información solicitada?

Mark only one oval.

	1	2	3	4	5	
No la he conseguido	☐	☐	☐	☐	☐	Si la he consigo en su totalidad

13. Valorando del 1 al 5. ¿Cuánto tiempo le tomo obtener la información solicitada?

Mark only one oval.

	1	2	3	4	5	
Muy poco, menos de 5 segundos	☐	☐	☐	☐	☐	Se demora mucho mas de 1 minuto

14. Valorando del 1 al 5. ¿Fue fácil acceder a la información solicitada?

Mark only one oval.

	1	2	3	4	5	
Muy facil	☐	☐	☐	☐	☐	Muy dificil

15. Valorando del 1 al 5. ¿El agente entendió todas sus preguntas y respuestas?

Mark only one oval.

	1	2	3	4	5	
No entendió ninguna	☐	☐	☐	☐	☐	Si, entendió todas.

Programas
e
información
adicional

Utilizando la información entregada de los programas del ministerio de inclusión económica y social: Consultar más detalles sobre dichos programas y responder las siguientes preguntas:

16. Valorando del 1 al 5. ¿Encontró la información solicitada?

Mark only one oval.

	1	2	3	4	5	
No la he conseguido	☐	☐	☐	☐	☐	Si la he consigo en su totalidad

17. Valorando del 1 al 5. ¿Cuánto tiempo le tomo obtener la información solicitada?

Mark only one oval.

	1	2	3	4	5	
Muy poco, menos de 5 segundos	☐	☐	☐	☐	☐	Se demora mucho mas de 1 minuto

18. Valorando del 1 al 5. ¿Fue fácil acceder a la información solicitada?

Mark only one oval.

	1	2	3	4	5	
Muy facil	☐	☐	☐	☐	☐	Muy dificil

19. Valorando del 1 al 5. ¿El agente entendió todas sus preguntas y respuestas?

Mark only one oval.

	1	2	3	4	5	
No entendió ninguna	☐	☐	☐	☐	☐	Si, entendió todas.

Recomendaciones

20. Valorando del 1 al 5. ¿Cree usted necesitar una, guía de uso para el Agente?

Mark only one oval.

	1	2	3	4	5	
No la necesito	☐	☐	☐	☐	☐	Es necesario para conocer sus funciones

21. Valorando del 1 al 5. ¿El agente presenta problemas a la hora de ser utilizado?

Mark only one oval.

	1	2	3	4	5	
Si, presenta muchos problemas	☐	☐	☐	☐	☐	No hubo ningun inconveniente

22. Si tuvo problemas. ¿Qué problemas presentó el agente?

23. Valorando del 1 al 5. ¿Han sido útiles las funcionalidades del Agente?

Mark only one oval.

	1	2	3	4	5	
Si, muy utiles	☐	☐	☐	☐	☐	No siento que sea útil

24. Valorando del 1 al 5. ¿Cree usted necesitar la ayuda de un Agente Humano, posterior al uso del Chatbot?

Mark only one oval.

	1	2	3	4	5	
Si, es necesario	☐	☐	☐	☐	☐	No lo requiero

25. Valorando del 1 al 5. ¿Qué calificación entregaría al Agente Chatbot?

Mark only one oval.

	1	2	3	4	5	
Pesimo	☐	☐	☐	☐	☐	Excelente

26. ¿Qué funcionalidades agregaría al Chatbot, aparte de las que actualmente posee?

27. ¿Desea enviar alguna recomendación adicional?

Fin de la encuesta

Muchas gracias por su ayuda.

This content is neither created nor endorsed by Google.

Google Forms

C Usability Test Results

18/11/2020

Test de usabilidad Chatbot MIES

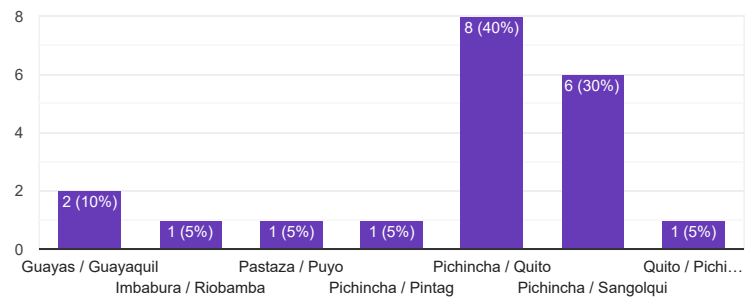
Test de usabilidad Chatbot MIES

20 responses

[Publish analytics](#)

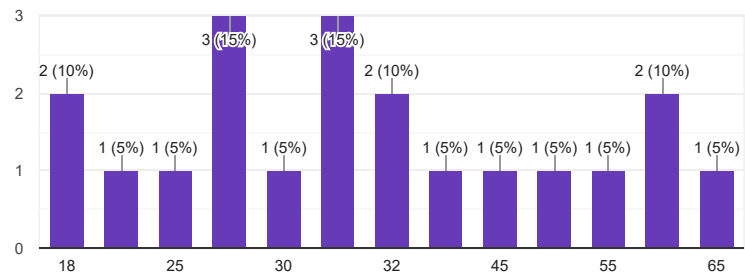
¿En qué provincia / ciudad reside?

20 responses



¿Cuál es su edad?

20 responses

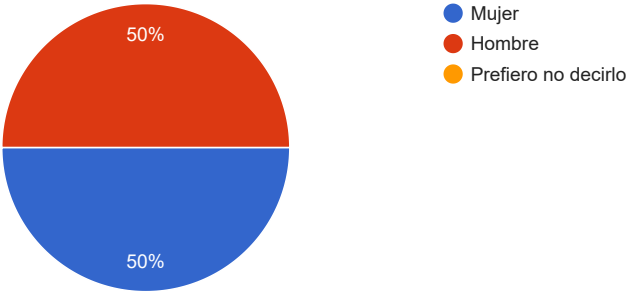


https://docs.google.com/forms/d/1dr1JAXN0UNrDBeFGosk0nyJhFB_rkI3bYC5jPg7rig/viewanalytics

1/14

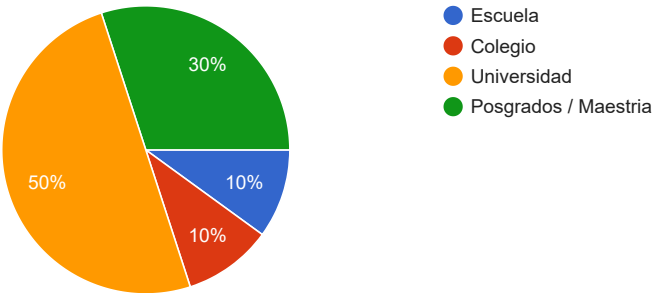
¿Cuál es su género?

20 responses



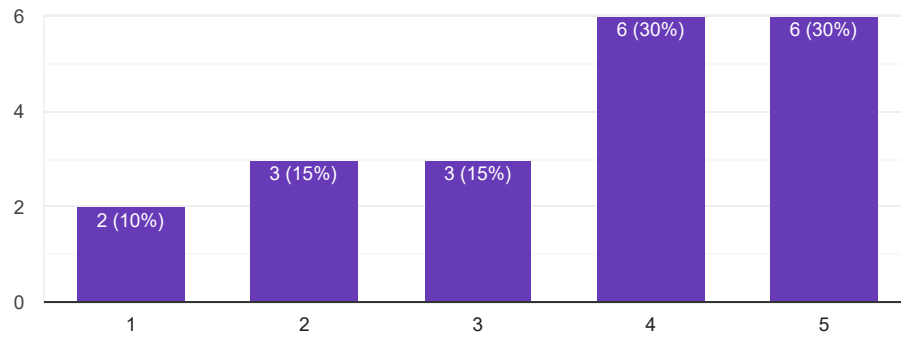
¿Cuál es su último nivel de estudios alcanzado?

20 responses



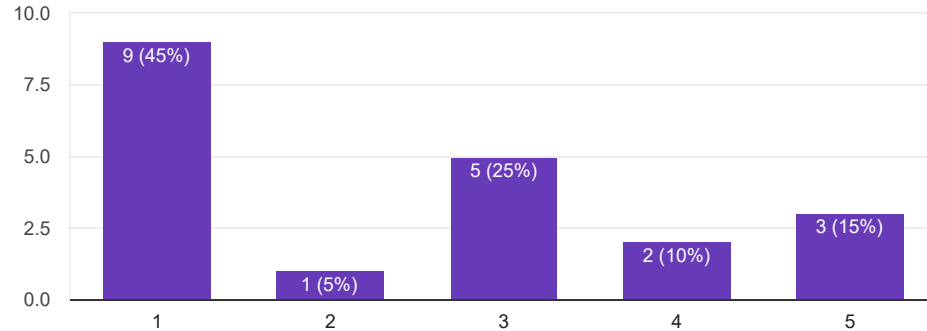
Valorando del 1 al 5. ¿Cuánto usa usted Facebook para consultar información que usted requiera.?

20 responses



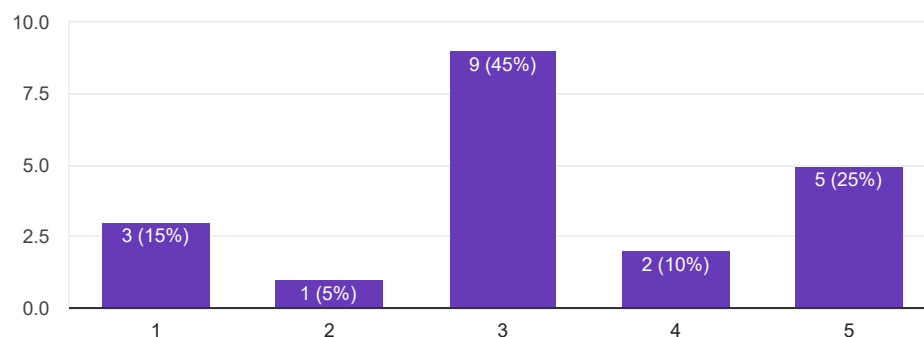
Valorando del 1 al 5. ¿Qué tan familiarizado se encuentra con los Chatbots?

20 responses



Valorando del 1 al 5. ¿Qué tan agradable ha sido su experiencia con Chatbots en otras páginas?

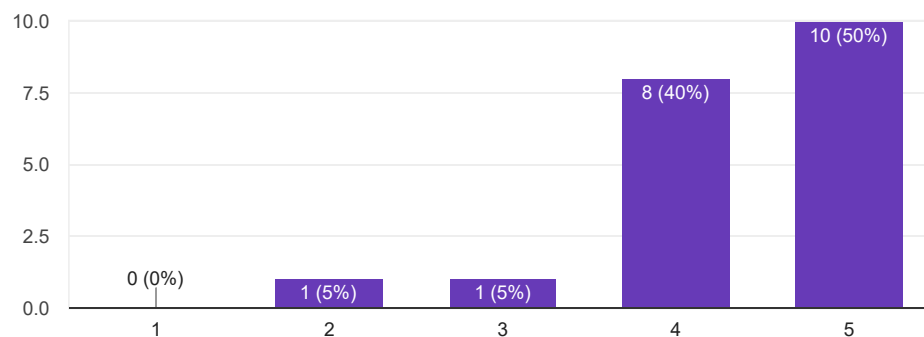
20 responses



Bono de Desarrollo Humano

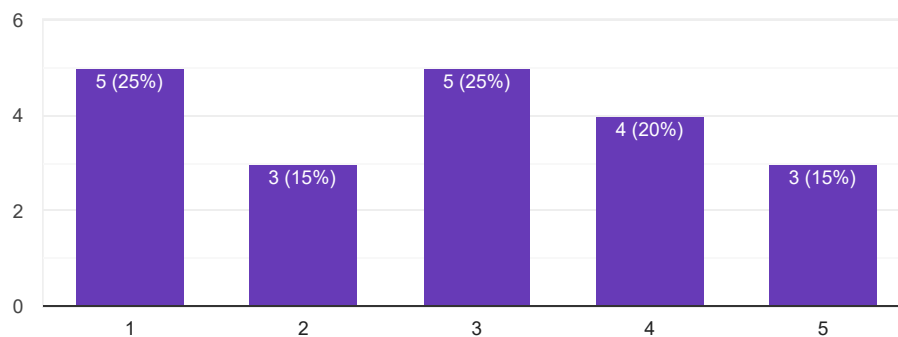
Valorando del 1 al 5. ¿Encontró la información solicitada?

20 responses



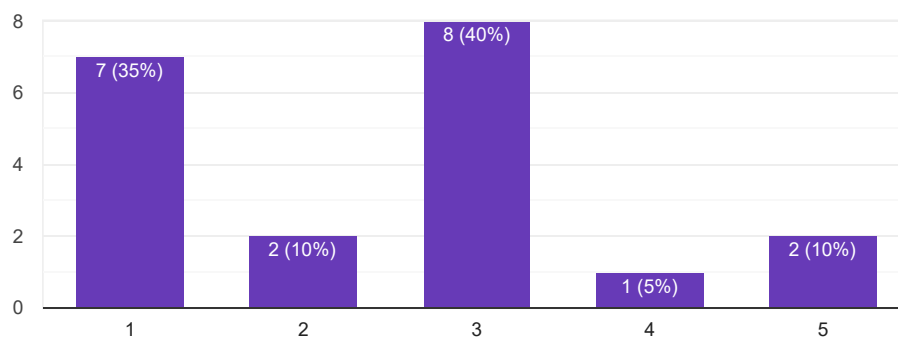
Valorando del 1 al 5. ¿Cuánto tiempo le tomo obtener la información solicitada?

20 responses



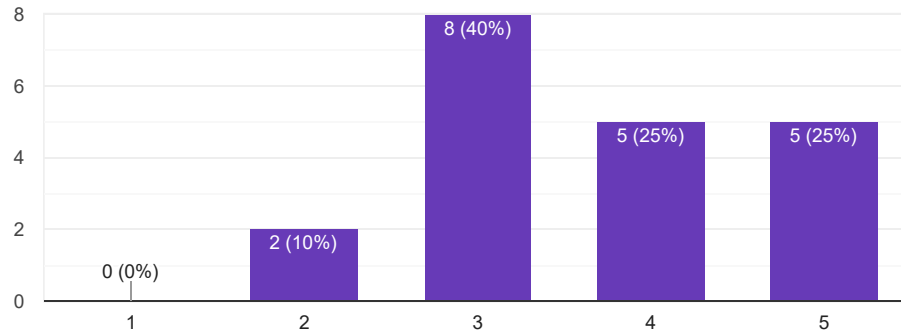
Valorando del 1 al 5. ¿Fue fácil acceder a la información solicitada?

20 responses



Valorando del 1 al 5. ¿El agente entendió todas sus preguntas y respuestas?

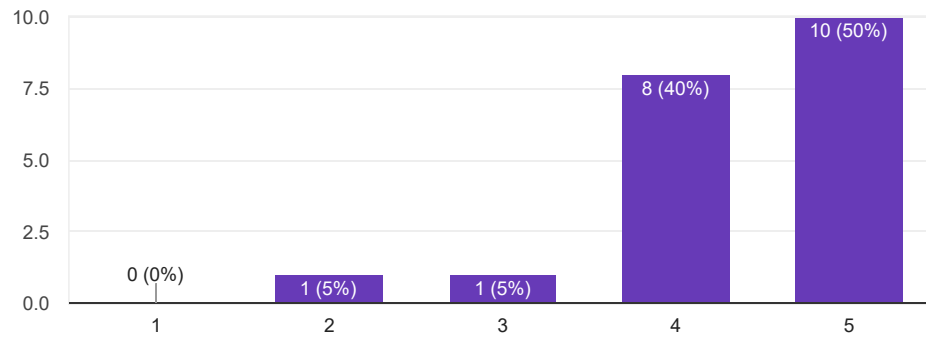
20 responses



Lugares de Cobro

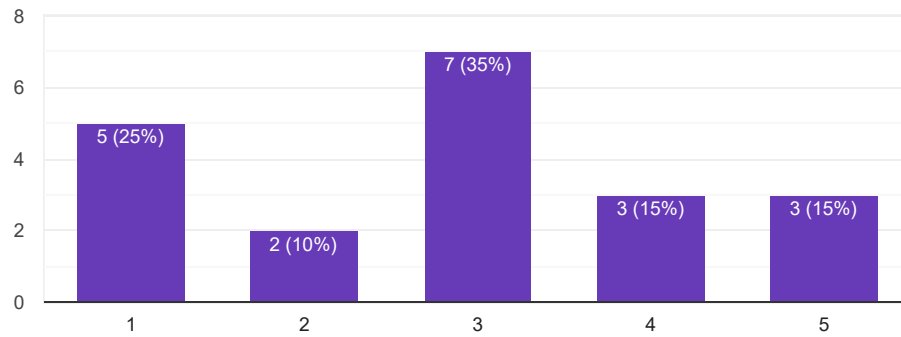
Valorando del 1 al 5. ¿Encontró la información solicitada?

20 responses



Valorando del 1 al 5. ¿Cuánto tiempo le tomo obtener la información solicitada?

20 responses



Valorando del 1 al 5. ¿Fue fácil acceder a la información solicitada?

20 responses

