Sapienza Università di Roma

Development and Evaluation of a Voice-based Conversational Agent for Prevention and Management of Kids with Obesity

Department of Computer, Control and Management Engineering
Corso di Laurea Magistrale in Engineering in Computer Science

Candidate
Andrea Vona
ID number 1923140

Thesis Advisor
Prof. Ioannis Chatzigiannakis

Academic Year 2020/2021

Thesis defended on January 14th, 2022
in front of a Board of Examiners composed by:

Prof. Alessandro De Luca (chairman)
Prof. Aristidis Anagnostopoulos
Prof. Roberto Navigli
Prof. Giuseppe Oriolo

Development and Evaluation of a Voice-based Conversational Agent for Prevention and Management of Kids with Obesity
Master's thesis. Sapienza – University of Rome

This thesis has been typeset by LaTeX and the Sapthesis class.

Version: January 8, 2022

Author's email: vona.1923140@studenti.uniroma1.it

Dedicated to
Alessandro Samaritani

Abstract

As underage obesity indicators grows around the world and Western Countries at an alarming pace, the urge for new strategies and technologies to solicit healthy behaviors is becoming more and more compelling. Under these circumstances, this work aims to present technical knowledge and usability's principles to design a voice-based conversational agent (VCA) able to promote healthy habits through engaging interactions. The technical part consists of a overview of state-of-art frameworks for designing chatbots as well as being comprehensive of a feasible architecture as starting point for future projects. Conversely, an analysis of the user experience (UX) and usability related to VCAs is illustrated by the means of the most recents studies of the field. Despite the fact that intelligent assistants may not provide good usability, these studies reveal that their acceptance is expanding attesting paradoxical circumstances. We then proceed to investigate the best practices regarding the modeling of human-agents interactions, how these differ from common human interactions and their sociological aspects as well as some considerations about accessibility of voice-based chatbots. After listing the heuristics regarding VCAs usability, we present how they have been employed for designing a robust conversation flow for our use case. Finally, we run user test and review the conversation flow in accordance with a Design Thinking Process-like approach in order to boost its usability and improve the overall user experience. This last step must be completed in order to reduce the consequences of developers' biases and develop higher empathy with the final users.

Acknowledgments

I would like to thank Prof. Chatzigiannkis for the chance he gave me and mostly for the incredible patience he had with me, as well as Michail Levitikos who preceded me on this work, for the selfless support he provided me.
I would like to thank the personnel of Sapienza and our Magnifico Rettore.
I thank my family for the support they gave me in these years.
I thank my friends Riccardo and Giuseppe for helping me, I wouldn't be here otherwise.
I thank Martina for making me the man I am today.

Contents

1 Introduction and Previous Work **1**
1.1 The Issue . . . 1
1.2 VCA as Determent to Underage Obesity . . . 3
1.3 Related Work and Requirements . . . 3

2 Technical Analysis **5**
2.1 Comparison between Available Technologies . . . 6
2.1.1 Dialogflow . . . 6
2.1.2 Microsoft Bot Framework . . . 7
2.1.3 Amazon Lex . . . 8
2.1.4 Rasa Stack . . . 8
2.1.5 IBM Watson . . . 10
2.2 An Example of Possible Architecture . . . 10

3 User Experience's Analysis **13**
3.1 Analysis of Usability in Intelligent Assistants . . . 14
3.1.1 Usability Studies' Outcomes . . . 14
3.1.2 The Social Dimension of the Interaction with IAs . . . 15
3.1.3 IAs' Adoption Study . . . 17
3.2 The User Experience of Chat Bots . . . 18
3.3 Accessibility and Other Considerations . . . 19

4 Conversation Flow Principles and Design **21**
4.1 Why a Rule-Based Conversational Agent . . . 21
4.2 Conversation Principles . . . 22
4.2.1 Human Types of Conversation . . . 22
4.2.2 Human-Agent Interactions (HAI) . . . 22
4.2.3 Heuristics for HAI Design . . . 23
4.3 Design of the Flow . . . 25

5 Testing and Prototyping **29**
5.1 Prototype 1 . . . 29
5.1.1 Welcome Intent . . . 30
5.1.2 First Intent: Starting the Cooking Conversation Flow . . . 30
5.1.3 Second Intent: Cooking for Family Members . . . 31
5.1.4 Third Intent: Choosing the Recipe . . . 31
5.1.5 Fourth Intent: Listing the Ingredients . . . 32

5.1.6 Fifth Intent: Missing Ingredients 33
5.1.7 Sixth Intent: Asking to Illustrate the Recipe 33
5.1.8 Seventh Intent: Repeating the Recipe 34
5.1.9 Last Intent: Concluding the Process 35
5.2 Peer Reviewing 35
5.3 Prototype 2 36
5.3.1 New Functionality: Weekly Plan 37
5.3.2 Corrections of Recipes and Ingredients' Listing Intents 37
5.3.3 New Intent: User Telling Ingredients 38
5.4 User Testing 39
5.5 Final Prototype 39

6 Conclusions **41**

Chapter 1

Introduction and Previous Work

This thesis is set to be considered the follow up of the work developed by Professor Chatzigiannakis and Dr. Levitikos. As it relies on their expertise as well as the ones' quoted, it is meant to take a step further on the implementation of a Voice-based Conversational Agent (VCA) to prevent underage obesity. Before discussing a possible practical approach from both the users' and the technical point of view of the development of the agent, in this chapter, it will be presented a brief analysis of the issue (in terms of statistics and social and health's costs to those affected by it) and the proposed mean (i.e. the VCA) to prevent and tackle overweight[1] in minors. Furthermore, it will be shown how the hindrances linked to weight gain can be prevented by a user friendly agent that promotes healthy eating habits.

1.1 The Issue

More than half of the population in 34 out of 36 OECD (Organisation for Economic Co-operation and Development) member countries is overweight and almost one in four people are obese.[1] Rates of severe obesity – known as morbid obesity – are now growing at the same pace as milder forms of obesity. Nonetheless, little over half the population in OECD countries for which data is available consumes a healthy diet and even fewer eat sufficient fruit and vegetables. People spend, on average, half of their waking hours in sedentary activities.[1] Finally, obesity and its related diseases reduce life expectancy by 0.9 years to 4.2 years, depending on the country. It is projected that by 2050 there will be around 92 million premature deaths from obesity-related diseases in OECD, G20 and EU28 countries.[1]

[1] Even if the terms "obesity" and "overweight" are linked to different states of the disease - as reported in the next footnote -, for sake of simplicity they will be used as synonyms to refer to people affected by the same disease unless stated otherwise.

As if that's not enough, it's necessary also to consider the economic burden national governments and public healthcare have to bear to treat patients affected by overweight. Indeed, OECD countries spend about 8.4% of their total health budget on treating obesity-related diseases. This is equivalent to about USD PPP 311 billion per year (or USD PPP 209 per capita per year).[1] In such an unpleasant context, the WHO European Childhood Obesity Surveillance Initiative (or COSI), reports that in the fourth round of data collection (2015 - 2017) 1 in 3 children aged 6 to 9 years is living with overweight or obesity[2].[2] The results indicate that the prevalence of severe obesity varied greatly among countries, and was highest in Southern Europe.[2] Obesity in teenagers provokes numerous social and economic issues since, for instance, those affected by it have less life satisfaction and are up to 3.8 times more likely to be bullied, which in turn may contribute to lower educational outcomes. Conversely, teenagers with a healthy weight are 13% more likely to perform well at school and are more likely to complete higher education.[1] The most common cause of obesity throughout childhood and adolescence is an inequity in energy balance; that is, excess caloric intake without appropriate caloric expenditure.[4] Other aspects involved in its development are physical activity, environment, sugary beverages, family and psychological factors.[5] As a result, obesity increases the risk of developing early puberty in children, menstrual irregularities in adolescent girls, sleep disorders such as obstructive sleep apnea (OSA), cardiovascular risk factors that include Prediabetes, Type 2 Diabetes, High Cholesterol levels, Hypertension, NAFLD, and Metabolic syndrome. Additionally, obese children and adolescents can suffer from psychological issues such as depression, anxiety, poor self-esteem, body image and peer relationships, and eating disorders.[4] Furthermore, there are numerous psychological studies that shows how the pandemic negatively impacted both physical activity and psychological well being which, as already stated, are among the factors that contribute to develop obesity.

[2]"Obesity" denotes the disease of the specimen with a Body Mass Index (BMI) classed as "overweight" or "obese" according to the WHO child growth curve standards, by gender. The BMI data are based on self-reported information from children on their weight (without clothes) and height (without shoes), with the BMI calculated as weight in kilograms divided by height in meters squared (kg/m2). These BMI data are then compared against the WHO's child growth standards – gender-specific empirically-based standards used to monitor child growth. Children with a BMI more than one standard deviation above the gender- and age- specific median (equivalent to a BMI of 25 at 19 years of age) are classified as "overweight", and those with a BMI more than two standard deviations above the gender- and age-specific median (equivalent to a BMI of 30 at 19 years of age) are classified as "obese". (See Inchley et al. (2016) and the WHO child growth standards website (http://www.who.int/childgrowth/standards/en/) for more information.)[3]

1.2 VCA as Determent to Underage Obesity

Today it is broadly accepted that overweight and obesity are largely preventable if better healthcare strategies are adopted and innovative interventions are introduced.[6] For this purpose, in the last decade numerous scientific researches have aimed to demonstrate the benefits of AI and rule-based chatbots in preventing and managing health issues through: helping the reshaping of negative behaviors by introducing healthy habits;[7] assisting health care professionals in decision making and daily operations;[8] supporting elderly people and reducing caregiver burden.[9] In addition, there has been an increasing number of studies to review the scientific literature on the matter.[10, 11, 12] The majority of them tend to agree that VCAs can positively impact the prevention of health conditions, either since the results of system accuracy and technology acceptance are encouraging [10], or due to the wide-spreading of VCAs[8], or, in this case, through a meta-analysis of the literature there are evidences linked to improved physical activity (which is a preventing factor for obesity)[13]. On the converse, all reinforce the fact that there is still a need to establish more conclusive evidence on the efficacy of VCAs for the prevention and management of health conditions, both in absolute terms and in comparison with standard health care.[10] Furthermore, critical factors to be found as hindrances to a definitive answer in the role of VCAs lay in the scarcity of experimental studies on the matter[12], the heterogeneity in the methods, the limited number of studies identified, and the high risk of bias showing that research on VCAs for chronic and mental health conditions is still in its infancy.[10]
Having highlighted how no secure answer are furnished by the current scientific studies and reviews, that either justify or reject the adoption of VCAs as prevention mechanism, providing a possible implementation of a VCA will be convenient not only as a plausible mean for the accomplishment of the aforementioned goal but also as base project for further investigations.

1.3 Related Work and Requirements

In Levitikos' thesis have emerged five macro categories of needs (related to obesity prevention and management) to be addressed by the VCA:

- Physical activity;
- Food supply;
- Information, intended as alimentary education;
- Daily nourishment;

- Treatment, as overweight therapy and prevention.[6]

The focus of the proposed VCA is on the daily nourishment as functional requirement to be met. In particular, the agent will be able to sustain a conversation flow (whose design and principles will be illustrated in chapter 4) based on the ultimate goal to act as meal planner, meal finder and cooking assistant for the user. From a high-level point of view, the main functionality of the meal planner is to schedule the meals for the user's week when asked for, in order to regulate the effort to prepare them and the calories intake. The meal finder's role is carried out by retrieving the recipes even in presence of guests. And lastly, as a cooking assistant it needs the ability to guide the user through the execution of the recipe.[6]

Chapter 2

Technical Analysis

Voice-based conversational agents are one of the trending topics as machine learning and artificial intelligence have taken over in most of computer systems. The key component of VCAs is the natural language processing (NLP) engines on which it depends the efficiency and responsiveness of the system to extrapolate the user requests and translating them into machine-comprehensible requests.

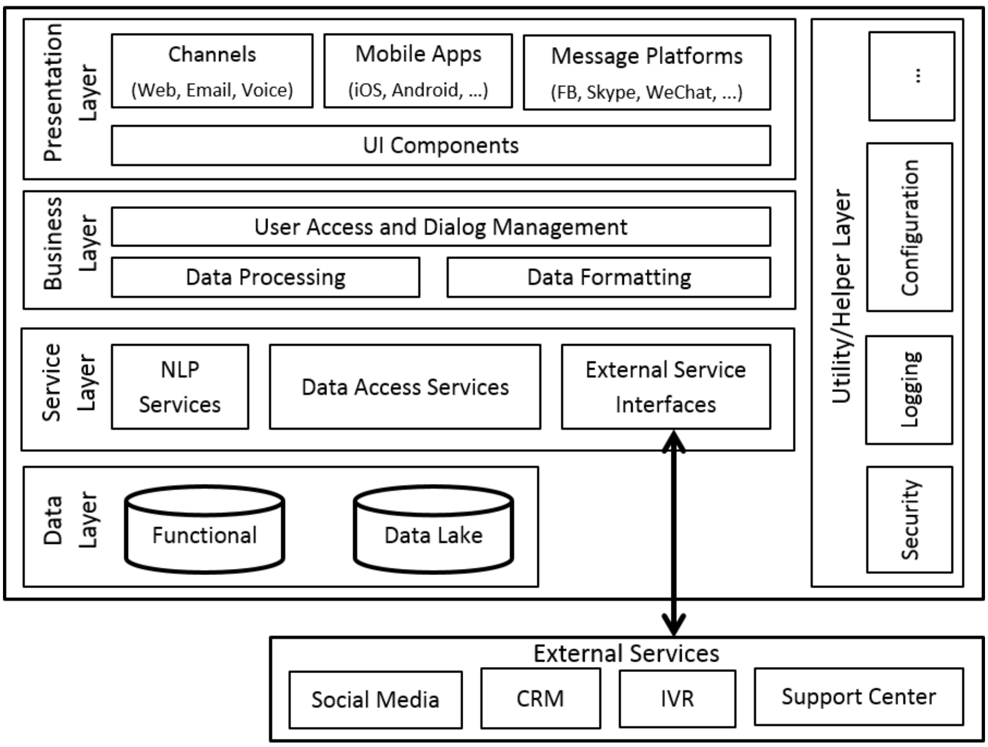

Figure 2.1. Conversational agents' standard architecture.[14]

2.1 Comparison between Available Technologies

As the market of conversational agents (in particular, voice assistants) grows and the latter are becoming more and more part of everyday life, a plethora of standardized frameworks have been developed by tech industries such as Amazon, Microsoft and Google. An analysis of the state-of-art frameworks is presented next, underlining strengths and weaknesses of each framework in terms of functionalities, costs and constraints as well as reporting their respective base architecture. Taking into account that there exists a large amount of frameworks, for sake of brevity the focus of this section will be on the most used and known ones.

2.1.1 Dialogflow

DialogFlow is developed by Google for building custom chatbot solutions. It supports both voice- and text- based assistants, as well as more than 20 languages and Internet of Things (IoT) integration for home automation. Furthermore, it provides:

- sentiment analysis for every query raised by user;
- live analytics reports once your chatbot is deployed;
- pre-built agents already trained on various knowledge domains for basic activities;
- one of the most intuitive developers' interfaces.

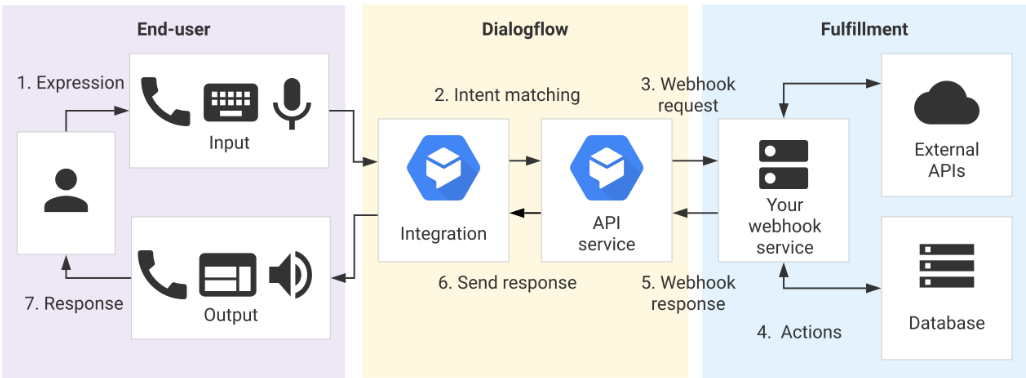

Figure 2.2. Dialogflow's base architecture and communication flow of internal modules with the fulfillment. [15]

Conversely, if the amount of intents[1] increases and the conversation flow grows in complexity, the entropy of the project increments as well, thus resulting that the

[1]From the documentation, "An intent categorizes an end-user's intention for one conversation turn. For each agent, you define many intents, where your combined intents can handle a complete

agent will misunderstand the user requests. Also, the developer has to train the bot inserting manually all the frequent synonyms which may result time consuming. Additionally, only one webhook is supported per project meaning that is not possible to assign to each intent a different webhook[2]. Lastly, as many other frameworks, its standard edition is free to use under a certain amount of queries.

2.1.2 Microsoft Bot Framework

Microsoft Bot Framework is part of Microsoft services and is integrated with vast majority, including Cortana, Skype, MS Teams, and so on as shown in figure 2.3. Unfortunately, there are major drawbacks in the framework. It obliges developers to pick between NodeJS and C# for creation or customization of their chatbot. It also needs a significant coding effort in order to implement simple functionalities. Moreover, most of the powerful capabilities are accessible only within the Microsoft toolkit. And lastly, the Azure documentation for the bot framework needs improvements.

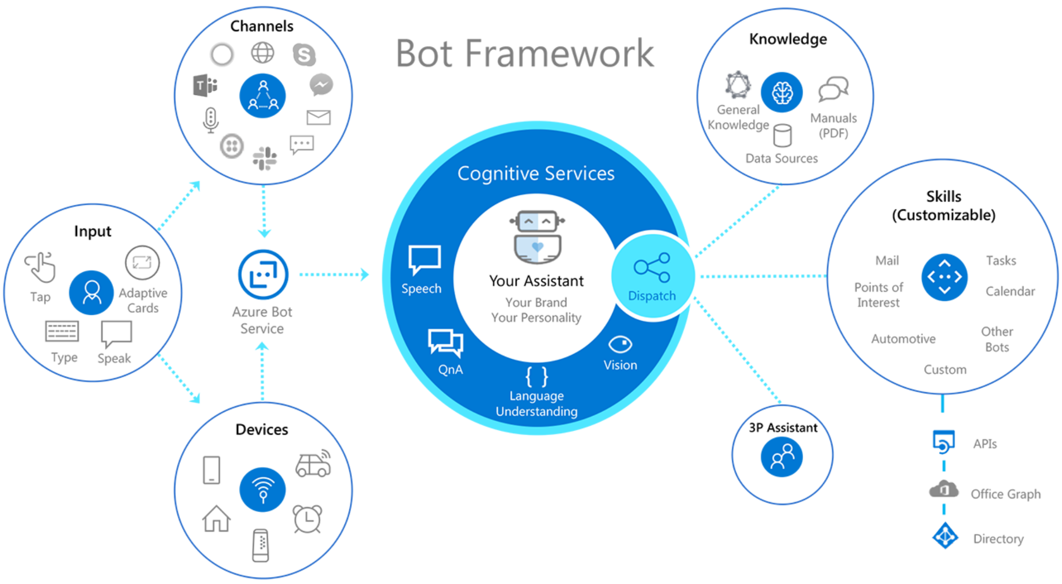

Figure 2.3. High-level Microsoft Bot Framework component's architecture. [16]

conversation. When an end-user writes or says something, referred to as an end-user expression, Dialogflow matches the end-user expression to the best intent in your agent. Matching an intent is also known as intent classification."[15]

[2]Webhooks are integrated in Dialogflow-designed chatbots through the fulfillment feature, whose architecture representation is reported in figure 2.2. This functionality will be better illustrated in section 2.2

2.1.3 Amazon Lex

Amazon Lex is a service for building conversational interfaces into any application using voice and text. It provides the advanced deep learning functionalities of automatic speech recognition (ASR) for converting speech to text, and natural language understanding (NLU) to recognize the intent of the text. Indeed, it is powered by the same technology and tools of widely-known VCA Alexa. Amazon Lex integrates with AWS Lambda which is an event-driven, serverless computing platform. It is a computing service that runs code in response to events and automatically manages the computing resources required by the indicated code.[18] This accelerates the deployment of the project and promotes higher scalability. Once implemented, an Amazon Lex chatbot can be deployed directly to chat platforms, mobile clients, and IoT devices. Moreover, the bot framework, complying to regular AWS functionalities, provides reports to track metrics for the agent.

One of the drawbacks of Amazon Lex is that it only supports English language. Apart from that it has a critical process flow for web integration, which comports a raise in complexity compared to other bot frameworks.[17] From a costs' point of view, there are no upfront commitments or minimum fees and the pricing behaves similarly to the previous cited frameworks (free within a certain threshold of messages, although discriminating between texts and voice messages).

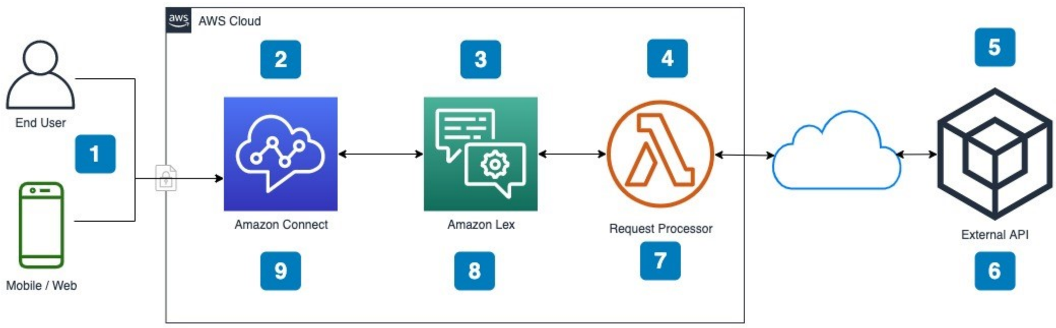

Figure 2.4. Similarly to Dialogflow's fullfilment, the image reports Amazon Lex's integration's architecture and communication flow with third parties' systems. [19]

2.1.4 Rasa Stack

Rasa Stack is the leader in the open-source machine learning toolkits that help developers create better AI chatbots with minimal training data. Compared to other chatbot frameworks, Rasa detains the advantage of not being constrained to any web services' platform (as for instance Amazon Lex to AWS). Also, similarly to IBM Watson Plus and Enterprise pricing plans' autolearning, Rasa is designed with interactive learning functionalities, particularly useful in early stages of the

projects where sufficient data may not be available for training the artificial intelligent chatbot.
Rasa chatbot framework holds a competitive profile also when it comes to costs. Indeed, as previously stated, Rasa is open-source and can be employed for commercial purposes. Additionally, exists an enterprise edition of Rasa as well, that provides accessorial services such as Single Sign-on system, analytics, customer support and so on. Nonetheless, the most important components are those offered in both editions. A minor drawback is represented by an unclear pricing system, similarly to IBM Watson: indeed, instead using the amount of messages as pricing factor it employs the size of the use case[20], although not specifying how without contacting the company.
From a technical point of view, the diagram in figure 2.5 provides an overview of the Rasa Open Source architecture. The two primary components are Natural Language Understanding (NLU) and dialogue management. NLU is the part that handles intent classification, entity extraction, and response retrieval. It's shown below as the NLU Pipeline because it processes user utterances using an NLU model that is generated by the trained pipeline. The dialogue management component decides the next action in a conversation based on the context. This is displayed as the Dialogue Policies in the diagram.[20]

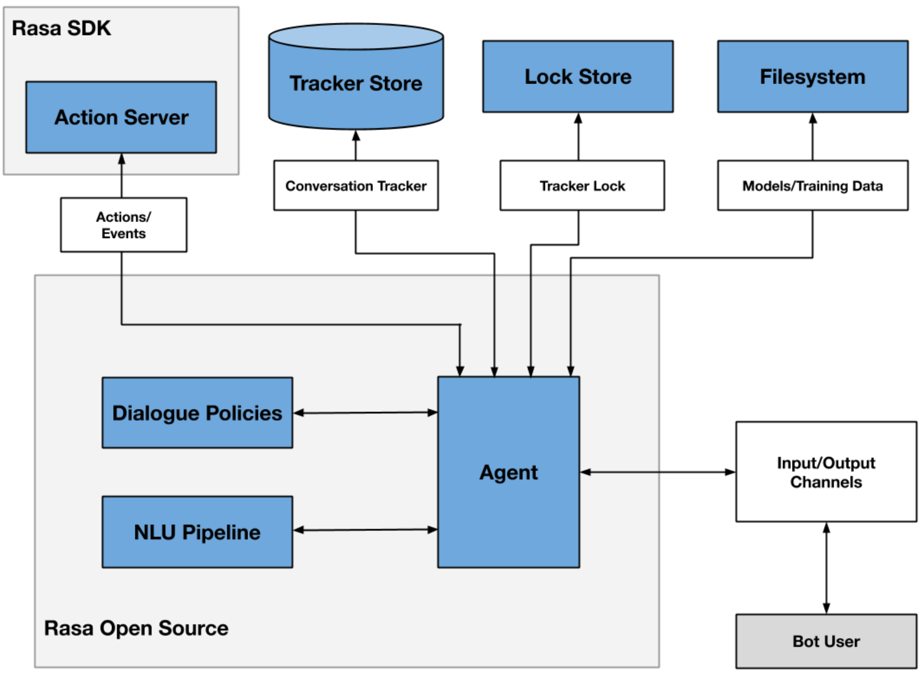

Figure 2.5. Rasa architecture diagram. [20]

2.1.5 IBM Watson

IBM Watson offers three possible pricing options: Lite, Plus and Enterprise.[21] The former is free and, without upgrading the subscription, has numerous missing features respect to the competitors' free alternatives. Indeed, it doesn't provide voice-based integration as channels for user interactions. To put it in other words, it won't be possible to develop VCAs in this pricing plan. Also, even by choosing a payed subscription, it is difficult to make projections about the costs since the latter are not publicly available without contacting the support. Conversely, it is illustrated that the amount of monthly active users is employed as a expenditure measure, dissimilarly to the aforementioned frameworks, which take in account the number of interactions and, in particular, of messages exchanged between end users and the chatbots. This could benefit the costs under the plausible assumption that the quantity of messages dealt is greater than the users' total.
Other services provided by the Plus and Enterprises subscription are versioning, enhanced AI services (such as autolearning, intent conflict resolutions, ecc), phone and voice features (that are obviously not included in the Lite option due to the exclusion of voice-based integration) and other deployment and security functionalities (mostly for Enterprise accounts).
From a technical point of view, an AI architecture representation is reported in figure 2.6, not only limited to the use case of the conversational agent but also showing how it is integrated inside a complete project.

2.2 An Example of Possible Architecture

In this section, it will be presented a practical architecture of the use case stated in the requirements.
As previously stated, the chatbot will have the role of meal planner, meal finder and cooking assistant. The framework chosen to carry out the goal is Google's Dialogflow due to the lightness of its architecture, the rapidity needed to realize a prototype and the intuitive interface.[3] The figure 2.2 illustrates a base architecture of a Dialogflow chatbot's main components:

- The communication channel (such as Google Assistant or Telegram). It works as interface between the middle layer and the end user.
- The middle layer, i.e. Dialogflow chatbot. This component is in charge to translate the user requests coming from the communication channel into APIs

[3] For the same reasons, Amazon Lex is also a good candidate for the proposed task, the discriminant factor has been Dialogflow's ease of use.

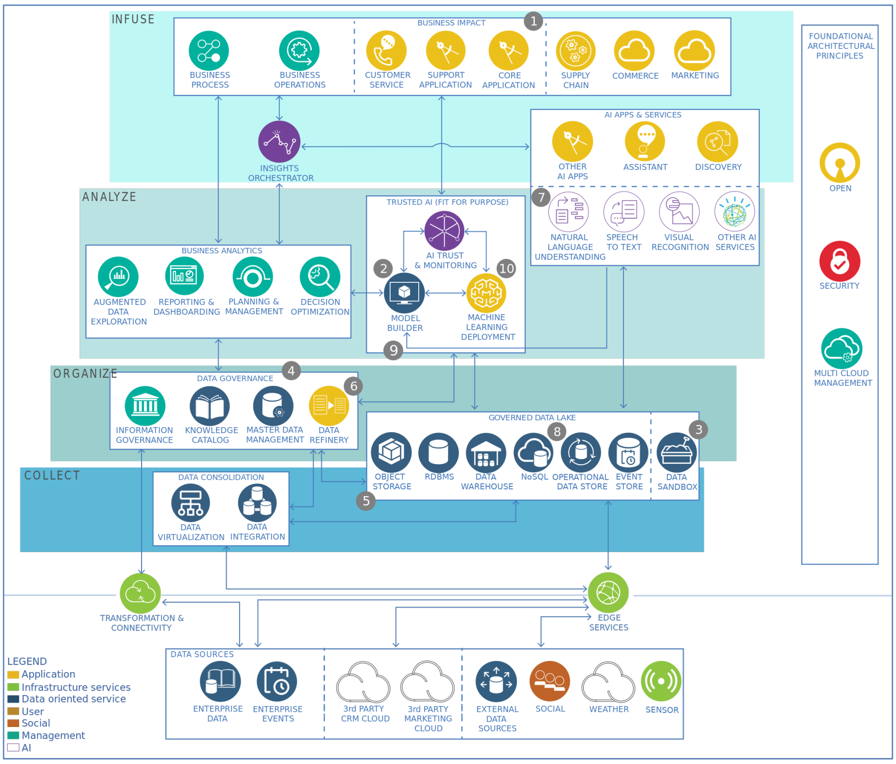

Figure 2.6. A complete AI project architecture realized by IBM's means. In the section AI APPS & SERVICES IBM Watson is represented by the ASSISTANT icon.[22]

calls. Here resides the integration for the communication channel, the NLU unit and the APIs service interface.

- The fulfillment is the last tier. It's in charge to execute the APIs calls and the Data requests through the webhook implemented by the developers.

The critical part is designing and hosting the webhook. Numerous technologies can be employed to reach this goal, although it is recommended to exploit Google Firebase for a variety of reasons: firstly, it will be necessary to meet the requirements for external webhooks; secondly, there is a discreet risk of performances' decrease; and lastly, it might be convenient to the developer the use of the internal console provided by Dialogflow, which is not employable to implement external webhooks. For these reasons, in the proposed architecture Firebase will host the webhook and Firestore (one of Firebase components) will store the database. The voice-based communication channel for Dialoglow is Google Assistant for obvious reasons, but it still needs to be selected as integration inside the project settings.
In conclusion, the flow of the communication between the internal components of the architecture will behave exactly as in the figure, taking in account the chosen communication channel and Firebase and Firestore in the fulfillment layer.

Chapter 3

User Experience's Analysis

In the introductory chapters, the project has been introduced in terms of the issue intended to be tackled (obesity) and the technical means to implement the proposed solution (i.e. the VCA). Although naively it may seem sufficient to carry out the purpose, it is not possible to achieve such task without users' adoption and acceptance. For this reason, it's intended to present an analysis[1] of the user experience (UX). As a matter of fact, whilst usability benchmarking studies and empirical standards are rare and domain-specific[23], it is broadly accepted that high usability and, more widely, positive user experience are key requirements for the adoption of any technological artifact. Indeed, usability is a quality attribute that assesses how easy user interfaces are to use.[24]
Usability is inversely correlated to interaction cost[2]: as the interaction cost with the system grows, the chances of the system to be rejected by the end users increase. Additionally, one of marketing and branding's goals is increasing the user motivation and the expected benefits for engaging with a particular site, brand or service[25], i.e. promoting its adoption. Moreover, the expected benefits and interaction cost are used to calculate the expected utility[3] of a system from the end user's perspective. Generalizing this statement, it is possible to consider as an adoption factor the expected utility. To put it more simply, higher expected utility is correlated with higher probabilities of a technology to be accepted and adopted by the end users. Thus the aim of any UX analysis is to maximize the system's expected utility by maximizing the expected benefits and minimizing the interaction costs.
Lastly, usability is a key component of the UX but the latter is a broader concept that encompasses all aspects of the end-user's interaction with a company, its services,

[1]This analysis will heavily rely on the works and articles produced by the Nielsen-Norman Group's experts, widely recognized as world leaders in research-based user experience.

[2]The interaction cost is the sum of efforts — mental and physical — that the users must deploy in interacting with a site in order to reach their goals.[25]

[3]The expected utility is the difference between expected benefits and expected interaction cost[25]

and its products.[26] For obvious reasons, this UX's analysis will be oriented only on the product, i.e. the VCA.
Under these premises, in this chapter it will be introduced an evaluation of the usability in Intelligent Assistants[4], which are the technological evolution of rule-based chatbots, whose UX will be assessed as well. The choice to discuss chatbots' UX is due to the fact that most of the frameworks presented in section 2.1, allow to design chatbots and VCAs inside the same projects. In the final part, there will be reported accessibility considerations and miscellaneous.

3.1 Analysis of Usability in Intelligent Assistants

Intelligent assistants (IAs) are an increasingly popular way of interacting with technology. Users repeatedly engage with them to perform relatively simple tasks such as getting the weather forecast or navigating to a destination. Many articles have been published to analyze the UX related to IA. Among this expanding literature, two non-experimental usability studies tried to assess the question whether users' needs can be addressed with state-of-art assistants. In the first research[27], 211 daily users of Alexa, Siri, or Google Assistant reported how they last used their assistant. In the second one[28], the users where asked to compile a week's diary showing how an ideal IA could have solved daily needs. Furthermore, the authors of the latter study had used the results of the previous work to interpret their results. Along with these inquiries, it has also been published an article to further investigate the social dimension of the interaction with IAs.[29] And lastly, it's reported a study on the IAs' adoption.[30]
A discussion regarding the outcomes of the aforementioned research is presented in the following sections.

3.1.1 Usability Studies' Outcomes

The "diary" article highlighted that existing assistants could have addressed 41% (177 out of 428) of the ideal needs logged in the study. Another 21% of these needs could be partially addressed by existing assistants. Nonetheless, diary-study participants attempted to use an existing digital assistant for only 7% of the unique needs they would want a digital assistant to address. Not using an assistant was eight times more common than using it for those needs that could have been fully or partially solved with the assistant. Some expectations were based on explicit data,

[4]Intelligent Assistants such as Amazon Alexa, Google Assistants and so on behave as communication channels in the architectures presented in the previous chapters since a VCA can be for instance added to Alexa as a new skill. For this reason the considerations formulated for Intelligent Assistants are extended to the proposed VCA of this work, as the latter is shipped by these technologies.

while others were more nuanced and involved subtle human-like cues. Participants logged a variety of needs ranging in complexity from simple, one-step actions to complicated, multi-step interactions.
Some people wanted the assistant to access others' personal data and alert them when they mess up or miss an appointment. Some of the most common needs recorded by diary participants were simple one-step tasks, such as ordering coffee or finding a sweet pie recipe. The least common needs were multitask needs, which required the assistant to program an action into the future for each person. The complexity of an activity is a major factor in whether today's assistants can successfully complete it. Users have great difficulty accomplishing advanced tasks with traditional computer systems.
Only 31% of the adult population in rich countries are capable of performing tasks similar to the ones addressed by Alexa, Siri and Google Assistant, when using traditional user interfaces. Researchers identified twelve different types of tasks that users logged on their smartphones and smart speakers. The most common tasks were reminders (26% of all) and creating new digital artifacts. Some needs were rated in multiple categories. For example, "Look up a recipe for zoodles and print it" was rated as both information retrieval (find a recipe) and IOT (sending it to the printer).
Figure 3.1 shows that the realized usefulness of current intelligent assistants (the green area) is fairly low, especially in the range of more complex tasks. The potential usefulness is much higher, as indicated by the full set of needs mentioned by the users: potential usefulness is represented by the full area below the top line in the chart. However, the usability gap (blue area) and the utility gap (orange area) lay inside most of that potential usefulness. The usability gap is caused by features that exist but are too difficult to use, whereas the utility gap is caused by missing features. Both gaps must be closed (or at least narrowed substantially) for intelligent assistants to be truly useful.[28]

3.1.2 The Social Dimension of the Interaction with IAs

The study conducted observed that customers are well aware that IAs are not fully intelligent. While users are not often conscious of the assistants' limits, their AIs' perceptions vary from frightening or to childish to just considering it as another computer tool. In addition, the authors report that the technology of IAs is still far from the future condition in which consumers would trust a computerized intelligent assistant in the same way they would a smart human administrative assistant.[29]
From an anthropology point of view, the social nature of language made people project anthropomorphic qualities onto the computer. Most of the participants

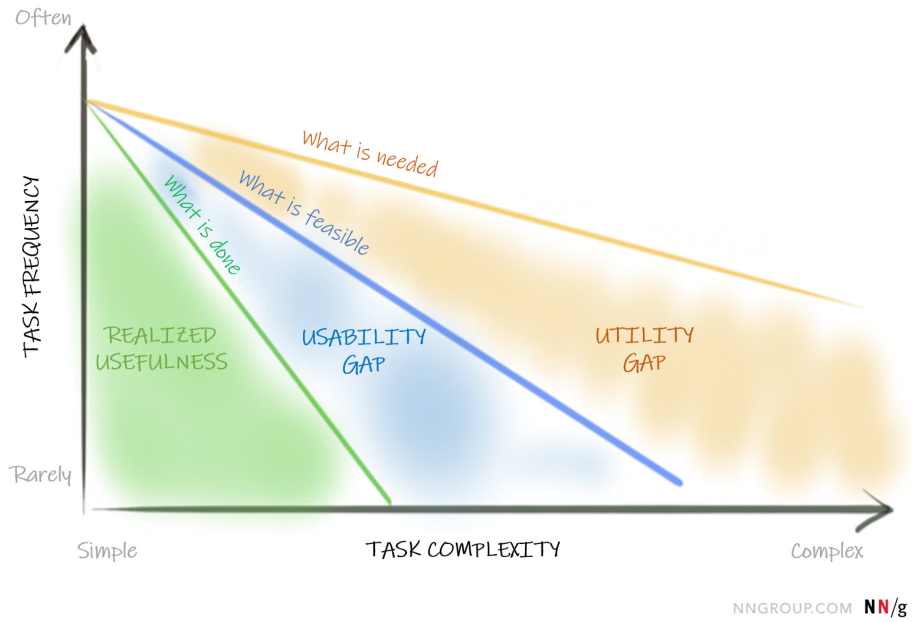

Figure 3.1. The gaps between what's done today with current intelligent assistants, what's feasible, and what is needed.[28]

referred to the assistant using a gendered pronoun. Some felt that slang might not be understood by the assistant and purposefully avoided it. Users did not expect agents to pick fine meaning distinctions. For example, a user who asked "How much is a one-bedroom apartment in Mountain View?" commented that her question was really too vague for Alexa.

The study also investigated scenarios that would promote or discourage the employment of IAs by the users. It emerged that there is no compunction carrying out a conversation with a real person in a public space although this behavior did not apply to intelligent assistants. Users in the study reported a strong preference for using voice assistants only when they were at home or by themselves[5]. Most people said they would not interact with a phone-based agent like Siri or Google Now in a public setting. Some, however, where willing to ask for directions while walking. A sentiment of awkwardness was common between users while describing a possible public interaction with an IA. Furthermore, whereas people usually reported using the assistants for simple, black-and-white queries, often in situations when their hands were busy, another common use was entertainment: many participants stated that at least occasionally they or someone in their family (usually their child) enjoyed hearing a joke or playing a game with the assistant. Several parents reported

[5]This is particularly important for the VCA proposed to prevent and manage obesity, indeed the opposite – where users wouldn't use IAs in private contexts – would have been an important issue which needed to be addressed.

using Alexa as a way of entertaining their children and keeping them away from a screen-based device such as a tablet or a smartphone.
Other concerns the users had in order to trust an IA are:

- Privacy and social awkwardness
- Always recording and transmitting audio to the cloud
- Consequences from misunderstanding what the user said
- Contacting other people in an unauthorized way
- Bugs that would cause smart-home features to work improperly
- Using excessive mobile data

In summary, the study concludes that even as intelligent conversational assistants rapidly improve in their ability to correctly understand user's speech, there are still some major social and mental-model challenges that prevent users from interacting with these systems naturally. Trust issues and user expectations for these systems drive the adoption of the agents for tasks beyond simple dictation and fact-lookup requests.[29]

3.1.3 IAs' Adoption Study

Despite the numerous usability problems, voice-based assistants are becoming increasingly popular with 46% of U.S. adults reported using voice-controlled digital assistants in 2017, according to the Pew Research Center.[30]
This paradox can be explained in terms of expected utility derived from the users' type of IAs' employment. Most frequent users do not use intelligent assistants to perform all tasks. Instead, they selectively assign certain types of tasks to their assistant. Indeed, voice assistants are good at performing a series of highly predictable tasks, such as getting directions or checking the weather and people mostly ask their agents to do tasks with only one step. Moreover, 26% of the participants used a voice assistant for tasks with multiple steps, but these were primarily getting directions. In addition to the perks of a smart use of IAs made by users, hands-free interaction is by far the most frequently mentioned benefit of using a voice assistant. The study therefore concludes that the benefit of being able to use a device "hands-free" outweighs the annoyance of poor usability.

3.2 The User Experience of Chat Bots

A chatbot is a text-based conversational interface that supports users with a limited set of tasks. A chatbot needs to possess two attributes: natural language processing and intelligent interpretation. There are two types of chatbots available today: customer service bots and interaction bots.
A study surveyed 8 US participants and asked them to perform chat-related tasks on mobile and desktop. Some of the tasks involved chatting for customer-service purposes with either humans or bots.[31] From this study, it has emerged that:

- Customers' attitudes toward bots ranged from neutral to slightly positive, according to the participants' responses.
- Most users are not aware of the existence of chat as a separate interaction channel in Messenger, Twitter, or Slack.
- People appreciated having both text and links used for inputting information into interaction chatbots and expected them for common inputs. [6]
- With simple linear processes that tackle complex tasks, users fear omissions. They doubt that the best answer can be gotten through the bot.
- The language used with customer-service bots was fairly complex[7] — most of the time people were focused on their problem and attempted to describe it; they did not think about whether the bot would be able to understand it or not.

In its conclusions, the study argues that in their current embodiment, chatbots' only advantage is less information overload. Additionally, the authors state that an intelligent chatbot that could answer any user question would have a huge interaction-cost advantage over any web- or app-based interface but that "unfortunately we're nowhere close."[31]
In summary, chatbot usability is a very incipient field of research, where the published studies are mainly surveys, usability tests, and rather informal experimental studies. Hence, it is necessary to wait for more accurate studies that will perform more formal experiments to measure user experience, and exploit these results to provide usability-aware design guidelines.[32]

[6] Comparatively to voice-based only AIs and VCAs, this is a major advantage since it respects Nielsen's usability principle "Recognition rather than recall" which states that the memory load for user should be minimized and instructions to continue should be easily retrievable.[33]

[7] The study compares the users' language used with chatbots to the one used with IAs. The study noted that since chatbots are less human-alike, the users would be more direct in making the requests and would drop the politeness markers such as "Please", "thank you" and "would it be possible to... ?"

3.3 Accessibility and Other Considerations

Accessibility should always be considered as a critical factor while developing a new product. In the case of web services, the cornerstones to implement an accessible utility are provided by the Web Content Accessibility Guidelines (WCAG2).[34] Although it's not the goal of this work to cover exhaustively the best practices in this field, it's presented a report about the main system qualities for achieving accessibility through a scientific review of the WCAG2.[35] The VCA should be:

- Perceivable. All elements of VCA dialogue must be available to a user according to their available senses.
- Operable. The operation of a VCA is dependent on the available communication modalities and (if applicable) the channel through which it is presented.
- Understandable. Information and the operation of the user interface must be understandable. It is based on the concepts of readability, input assistance and predictability.
- Robust. The content exposed by the VCA must be robust enough that it can be interpreted by a wide variety of user agents, including assistive technologies.

In conclusion, we have presented in this chapter the critical points which must be taken into consideration when designing usable VCAs and chatbots. In the next chapter, it will be shown how the conversation flow has been implemented to overcome the aforestated issues.

Chapter 4

Conversation Flow Principles and Design

4.1 Why a Rule-Based Conversational Agent

The two most common conversational agents are completely AI-based and rule-based chatbots[1]. It's crucial to note that even in rule-based chatbots, artificial intelligence and machine learning are present, and they serve to fuel the NLU components of the system, as discussed for each framework in chapter 2. However, as a result, the machine learning model built within rule-based chatbots only covers natural language understanding (NLU), and the chatbot cannot learn from prior encounters. As a result, the developer will have to create a conversational flow. The bot will obey regulations (also known as "rules") designed a priori by the developer in response to user requests collected via the exchange. More formally, a rule-based chatbot is modeled by a finite-state automata, in which the conversation always is in one definite state of the conversation at a time, each state having a fixed number of transitions to other states.[36]

We chose to build upon Levitikos and Chatzigiannakis' work and create a rule-based chatbot prototype hence extending their work. The rationale behind this decision is to gain valuable information insight directly from users, rather than putting the workload on the implementation of an artificial intelligence chatbot, which would have required significantly more effort for development and thus diverted attention away from the user experience (UX). As a result of giving a holistic study of the user experience, the scope of this work to provide the groundwork for a useable VCA

[1]A more accurate chatbots' classification takes in account four attributes: interact mode (voice- or text- based), open or closed knowledge domain, task- or non-task oriented, design approach (rule-based, retrieval-based and generative based).[37] Under this categorization, the chatbot proposed is voice-based, task-oriented, rule-based and belongs to a closed knowledge domain.

that is independent of the technology used to deliver it.

4.2 Conversation Principles

Before enunciating the conversation principles, it is necessary to provide a clear explanation of conversation. Spoken conversation is defined as “any interactive spoken exchange between two or more people”.[38]

4.2.1 Human Types of Conversation

Human spoken conversation serves many purposes. These are broadly classified as transactional (task-based) or social (interactional). Transactional conversation pursues a practical goal, often fulfilled during the course of one interaction. In these types of exchanges, both interlocutors know what the goal of the dialog is. They have different clearly defined roles, and success is measured by the achievement of the transaction’s purpose. The aim of more social conversation is not to complete a task as such, but to build, maintain and strengthen positive relations with one or more interlocutors. Social conversation ranges from small talk and social greetings to longer interactions. Examples include talk between friends, office chat, or brief discussions between strangers. This type of social conversation can help develop common ground, trust and rapport between interlocutors. Although transactional and social talk serve different purposes, they often overlap in natural conversation.[38] Critical to a conversation is the opening of a channel by an interlocutor, with a commitment from the other to engage, each then using the dialogue to co-construct meaning and converge on agreement. Particularly in task oriented dialogue, this may lead to the proposition of an action to be completed or a transaction to take place.[38]

4.2.2 Human-Agent Interactions (HAI)

The research literature regarding the conversation between humans and artificial agents unanimously agrees on categorizing the latter as almost exclusively task-oriented and transactional for its users.[38, 39, 40] However, according to a study[38], there are settings in which conversational agents may need orientation toward interpersonal and social objectives to establish a long-term and intimate human-agent relationship. The research indicates that the conversational content and structure of encounters with strangers or acquaintances may serve as an excellent starting point for social agent dialogues. Nevertheless, technology does not have to replicate the form or consequences of human discourse. Rather than imitating human conversational skills in the goal of engaging in effective social interactions with users,

the study's results imply that we should approach human-agent interaction as a distinct genre of conversation with its own set of rules, conventions, and expectations. Furthermore, the results show that chatbots and VCAs may be more functional and intrinsicly valuable, focusing less on the relational growth or emotional outcomes associated with human communication.[38]
Furthermore, the study confirms the considerations reported in section 3.1.2 regarding the importance of context in shaping the content and norms of agent conversation. For instance, what people deem appropriate to divulge or discuss conversationally with agents may differ markedly in private and public settings. Current users are unlikely to engage with IPAs in public, noting social embarrassment and awkwardness. Users have fewer concerns divulging private information when using VUIs in private compared to social spaces. The study identifies a clear distinction between human and agent based conversation in terms of its perceived norms, rules and expectations. "The context of interaction is no doubt likely to impact these and further work should look to explore this impact" the study concludes.[38]

4.2.3 Heuristics for HAI Design

In order to present the best practices for HAI design, it is necessary to introduce the concepts of "Gulfs of Execution and Evaluation": the Gulf of Execution, where users try to figure out how something operates, and the Gulf of Evaluation, where they try to figure out what happened (Figure 4.1). The role of the designer is to help people bridge the two gulfs.[41]

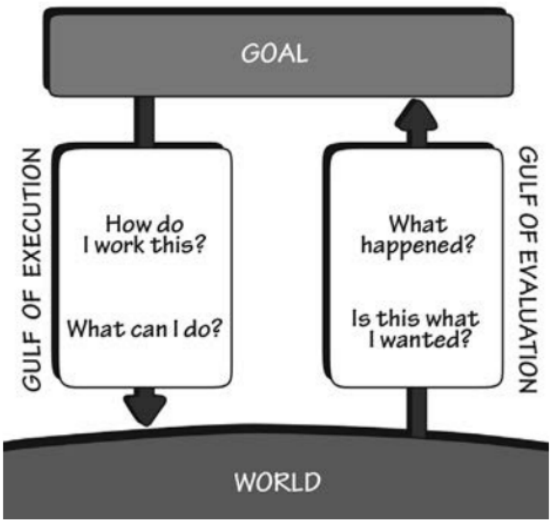

Figure 4.1. The Gulfs of Execution and Evaluation. When people encounter a device, they face two gulfs: the Gulf of Execution, where they try to figure out how to use it, and the Gulf of Evaluation, where they try to figure out what state it is in and whether their actions got them to their goal.[41]

The Gulf of Evaluation reflects the amount of effort that the person must make to interpret the physical state of the device and to determine how well the expectations and intentions have been met. The gulf is small when the device provides information about its state in a form that is easy to get, is easy to interpret, and matches the way the person thinks about the system. The major design elements that help “bridge” the Gulf of Evaluation are, indeed, feedback and a good conceptual model. On the other hand, it is possible to bridge the Gulf of Execution through the use of signifiers, constraints, mappings, and a conceptual model.[41]
Under these assumptions, we present the heuristics’ principles which aim to minimize the two gulfs. They have been obtained through a research published in 2016.[42]

- **CA design should reveal system intelligence.** For the study’s participants, expectations of how to interact with the CA were out of step with reality. In the majority of cases users were unable to make accurate judgments about system capabilities. Anthropomorphism set unrealistic expectations that framed user perceptions of what constituted system failure.[42] Indeed, the specimen with lower levels of knowledge described little alteration in their expectations and greater levels of frustration, leading them to question the ‘intelligence’ of the system, indicating that user expectations of CAs should be scaffold through more considered revelation of system intelligence through design.[42]

- **Reconsidering the interactional promise made by humour.** Despite users actively engaging in the process of ’learning’ to speak more simply to their CA, this did not seem to affect their expectations of system intelligence. Even where users perceived CA failure, they continued to attribute episodic social intelligence to the system, such as sarcasm or humour. One reason might be the expectation set by the act of conversation and use of humour as a form of interaction.[42]

- **Consider new ways of conveying CA capability though interaction.** In order to reflect the expectations set by ‘conversation’ as interface, some thought should be given to how to convey system limitations and capabilities in a variety of ways. Currently, the reality of CAs is such that the system response presents only task-related information to the user. In some cases this has the consequence of conveying capability, for example reverting to visual web-search as an indicator that the system is struggling, or through polite trigger responses that might signal the user about limited connectivity.[42]

- **Rethink system feedback and design goals in light of the dominant use case.** The majority of the study’s participants used a CA where their primary task required a high level of attention (e.g., driving, cycling, child-minding). In each use case described by participants, the activity was not only hands-free but required a level of visual attention. Where participants felt they had to resort to 'old school' techniques, or where the CA reverted to screen-based response, the resulting stress and extra effort were seen as a failure from their perspective. Most of the users engage with the system only until it ceases to provide utility. This begs the question, what is the design goal of the current CA system and how might these be rethought to deliver a more compelling user experience.[42]

4.3 Design of the Flow

So far, we have presented the reasons why a rule-based chatbot as been chosen as underneath technology for the upcoming prototypes and the conversation principles that will guide its design. In this section, we will provide the ideal conversational flow that will be implemented by the user and the VCA. In order to achieve this goal, we will start from a brief list of functional requirements for the VCA.
The chatbot is requested to:

- Be activated when the user talks to him and greet the user.
- Interacting with the user in order to learn what meal the user intends to cook, the date, who is making it for (guests, the cooker only, the family members and so on).
- Present the recipe's name and ingredients and wait for feedback from the user whether he would like to use the proposed recipe or not.
- Ask to present the recipe and, in case, present it.

These are the requirements proper to our study case. Along them, we will make the following assumptions:

- The user is already logged and has previously carried out a profiling phase, by indicating the name, the age, weight, allergies and so on.
- The user's family members are profiled as well. The chatbot knows who they are and holds the same information of the logged user.
- The system keeps track of the past meals in order to monitor the calories intakes and will unlikely repeat suggesting the same recipes he has already suggested.

In figure 4.2, are presented the building blocks used to model the interaction between the user and the system (i.e., the VCA).

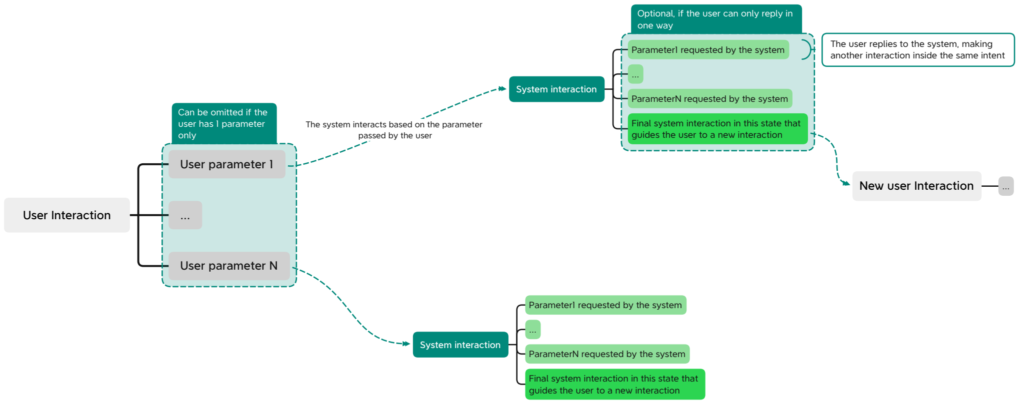

Figure 4.2. The interaction building blocks legend.

The conversation flow designed for our VCA is reported in figure 4.3 and is based, as much as possible, on the principles anticipated in this and the previous chapter. For instance, from the very beginning of the interaction, when the user greets the system, the latter clearly states what it can do for the user (i.e., assist him with cooking). This choice, along with the more general Nielsen's usability heuristics, is compliant with the first best practice listed in subsection 4.2.3. Nonetheless, it's imperative to notice that designing a good conversation flow alone is not enough to reach the wider goal of high usability. Indeed, in order to achieve this scope, it is necessary to have a broader point of view on the UX, which can be obtained only by user-testing the delivered product. For example, the aim of designing a positive conversation flow in a rule-based chatbot is to model the interactions that modify the state of the system in a usable way like minimizing redundant states or implementing interactions that guide the user from a safe system state to a risky one in a not-risky perceived way (in our case, asking a double confirmation before proceeding to the next state is meant to feel the user in control of the system and not vice versa). But beyond this, there are a series of factors that can be developed in order to increment a VCA's usability, for instance, taking into account humor (as reported in the second principle stated in subsection 4.2.3) to raise the engagement of the interactions. Indeed, the remaining elements will be covered in the next chapter.

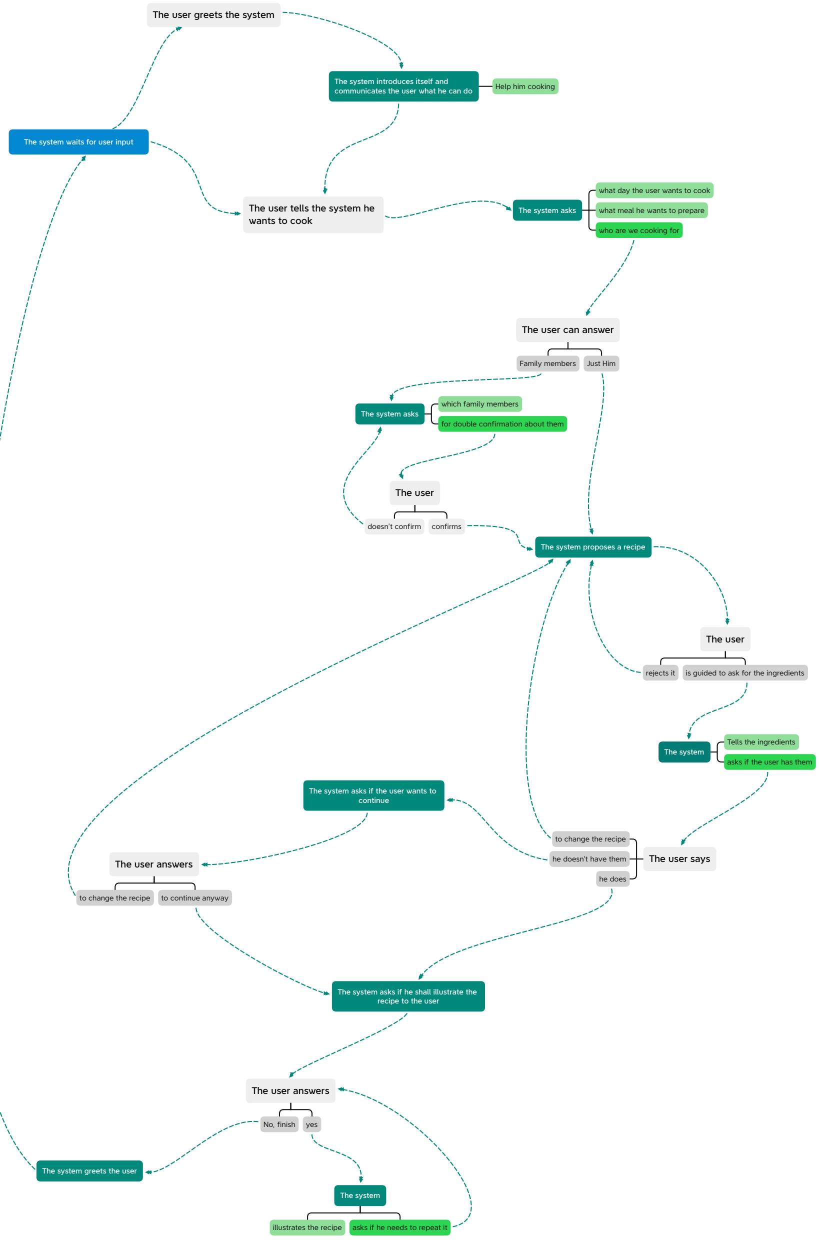

Figure 4.3. Conversation flow designed

Chapter 5

Testing and Prototyping

In this chapter, we will analyze the usability of the prototypes produced in order to achieve a more satisfactory user experience of the final prototype of the chapter. Although we haven't followed all the steps of the design thinking process methodology, the latter clearly inspired the conduct of this work as prototyping and user-testing are some of its key components. We will use a peer review made by Dr. Levitikos who did not actively participate in any stage of the implementation of this work. Moreover, after having added the usability discoveries to produce a second prototype we will test it with real users.
Lastly, we will add to the assumptions on the system that we made when discussing the conversation flow development (in section 4.3) also a fallback option, whenever the user will fail in any part of the interactions with the bot: its discussion has been omitted for simplicity but the user will be guided to answer correctly to the instances of the system, thus making a cycle in the conversation flow that will be broken when the flow will be restored.

5.1 Prototype 1

We used (i) Dialogflow in order to implement a static prototype and (ii) its standard interface for the testing, although, in an enterprise setting, it is possible to integrate Dialogflow with emphActions on Google for the testing and *Actions Builder* for the developing as the Google Assistant integration is being deprecated.
Let us take into account the conversation flow designed in section 4.3. We will break it down into intents[1] which will compose our first prototype. An intent is formed by an initial user interaction and a reply by the system, that will lead to a new user interaction that will start a new intent. It's interesting to notice that, on one hand, the user is the leader of the conversation as he is always the one to initiate a new

[1]The intents are the components of an interaction developed with Dialogflow framework.

intent but, on the other, the system guides the user through the conversation via a series of questions and suggestions at the end of every intent.
We will maintain the same diagram notation from section 4.3 in order to make easier for the reader to understand what part of the flow we are implementing, although in the former section we were modeling the interactions from a theoretical point of view whilst here the interactions which form the intents are actually implemented inside the prototype.

5.1.1 Welcome Intent

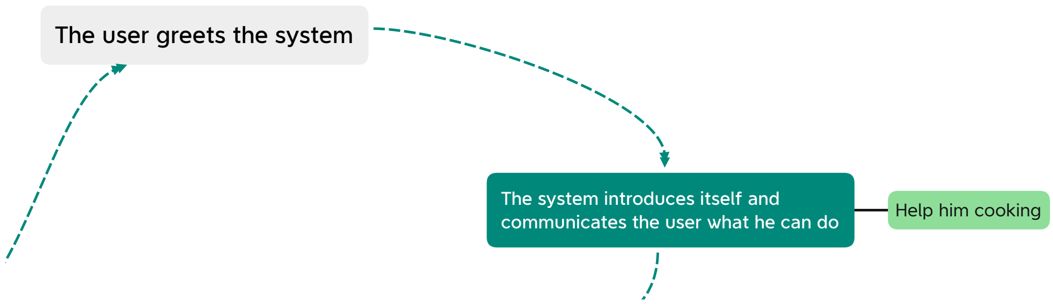

Figure 5.1. Welcome Intent

The purpose of this intent is to facilitate the user in the learning process of the prototype. Indeed, as stated in the tenth principle of Nielsen's heuristics, the user should be provided with help and documentation. Furthermore, Nielsen also stressed that ideally the system should be usable to the extent that can be used without documentation.[33] And this is what we seek to achieve. Although, this is a difficult requirement to be met if it's the first time the user is employing the system and therefore a "Welcome" intent has been developed.
Even if this prototype provides only one service, we want to point out that as the project will be enriched with functionalities, this intent is easily scalable thus adding no interaction costs to further use cases.

5.1.2 First Intent: Starting the Cooking Conversation Flow

Now that the user knows how to make use of the prototype, he will be able to start a conversation with the VCA. The user can directly feed the system with the parameters it needs or otherwise, he will be asked to provide them. The darker green parameter requested by the system will trigger the beginning user interaction of the next intent.

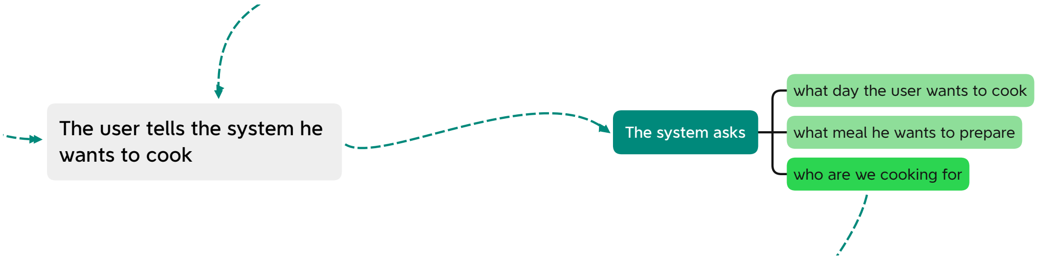

Figure 5.2. First Intent: Starting the Cooking Conversation Flow

5.1.3 Second Intent: Cooking for Family Members

The first distinction requested by the system is to identify who is the logged user cooking for, whether it's just him or the family members as well. Remember the assumption that the system knows who composes the family. This filter is necessary to propose the correct recipe, as, for instance, other relatives might have allergies or share different preferences on food.
We also included a double confirmation request by the system. This step is imperative since informing the user of the state of a system (especially in a Voice-Based setting where it is harder for the user to retrieve the system state) is one of the most important strategies when it come to enhance usability and minimize the gulf of evaluation presented in the previous chapter.[33, 41]

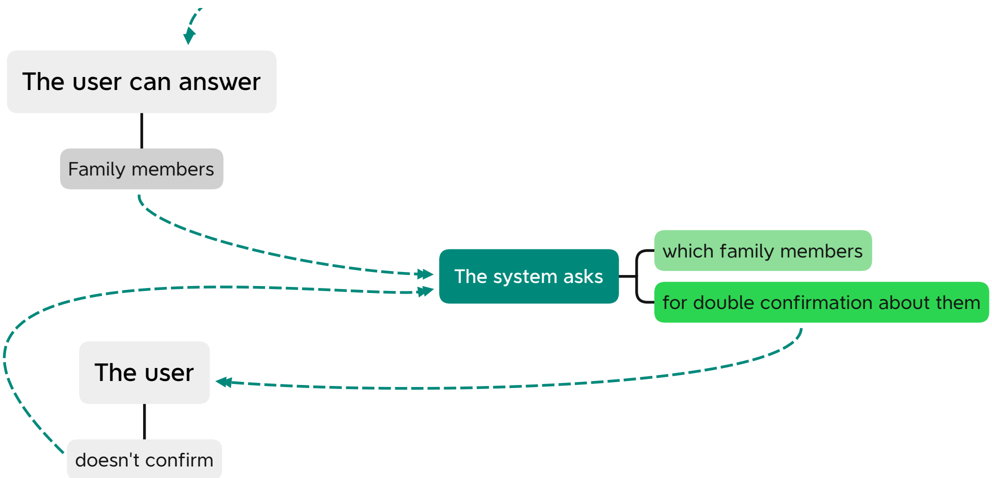

Figure 5.3. Second Intent: Cooking for Family Members

5.1.4 Third Intent: Choosing the Recipe

As complex as it might seem, it's actually fairly easy to carry out this intent. There is a plethora of user interactions that trigger this intent, where the user:

- has declared who he is cooking for (whether it's just him or it has confirmed the family members from the second intent);
- is not satisfied with the proposed the recipe and cycles until has found one is pleased by;
- doesn't own the ingredients needed for the recipe (see next intents);

Once any of these interaction occurs, the VCA proposes a recipe, that might be approved or rejected by the user.
One could argue that there are redundant states of the system[2], that trigger this intent. This is, indeed, true but it must not to be considered as an obstacle to enhance usability, we argue it holds the converse. Moreover, since the task of choosing a recipe – we hypothesize – might be the most frustrating one for the user, we provided many states of the interaction[3] with the user to reach this one in order to mitigate the risk of regret in choosing a recipe, which would lead to worsen the UX.

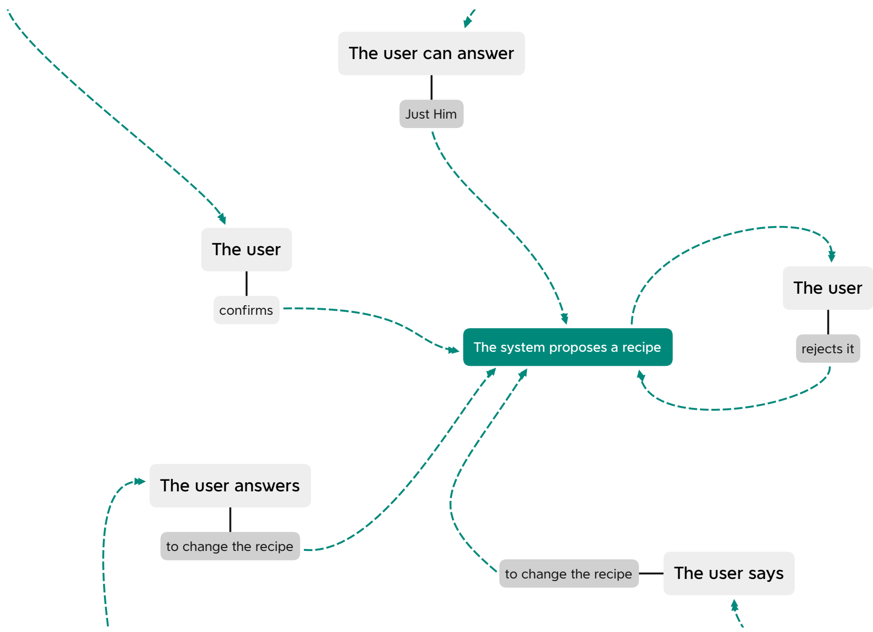

Figure 5.4. Third Intent: Choosing the Recipe. *We consider this task the most relevant for the implemented prototype.*

5.1.5 Fourth Intent: Listing the Ingredients

[2]in particular case, we refer to the "system" as the entire conversation flow and not the VCA.
[3]in this case, by "interaction" is meant it as a whole and not a particular one that, for instance, sparks an intent.

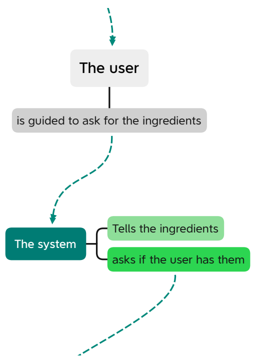

Figure 5.5. Fourth Intent: Listing the Ingredients

Once the recipe has been approved by the user, the latter is prompted to ask for the ingredients. As a result, the system lists the ingredients and demands whether the user has them as shown in figure 5.5. Based on the answers which are indicated by the system (the user is guided to indicate if "he does [have the ingredients]", "he doesn't have them" or "to change the recipe"), the user will trigger the next intent.

5.1.6 Fifth Intent: Missing Ingredients

This intent deals with the problematic point of the conversation with the user, where the latter is not provided with all the ingredients. As a response to the user stating that he doesn't own an element of the recipe, the system asks the user whether he would like to continue anyway or if he wants to change recipe.

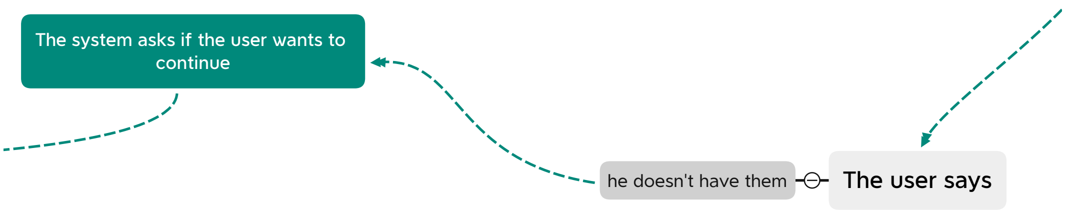

Figure 5.6. Fifth Intent: Missing Ingredients

5.1.7 Sixth Intent: Asking to Illustrate the Recipe

Whether the user has all the ingredients or, does not mind not having one, the resulting interaction leads to this intent. The system in order to complete this part of the conversation asks the user if it shall illustrate the recipe. The answer of this question might seem obvious, but it could be the case that a user already knows the recipe or the reason the user engaged with the conversation is to schedule a recipe for another day, which is also permitted otherwise the VCA wouldn't ask what day the user intends to cook in the first intent.

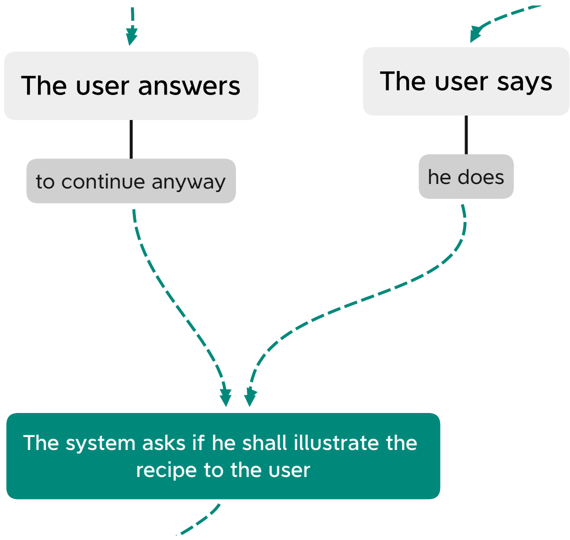

Figure 5.7. Sixth Intent: Illustrating the Recipe

5.1.8 Seventh Intent: Repeating the Recipe

As stated in chapter 3, pure VCAs (opposed to voice- and text-based conversational agents) have to deal with diminished usability due to voice only interactions. The only purpose of this intent is to mitigate this issue simply by repeating the recipe. Whenever a user reaches this intent, it is possible to conclude the interactions were successful. As shown in figure 5.8, this intent is initiated by the user telling the VCA to reproduce one more time the recipe.

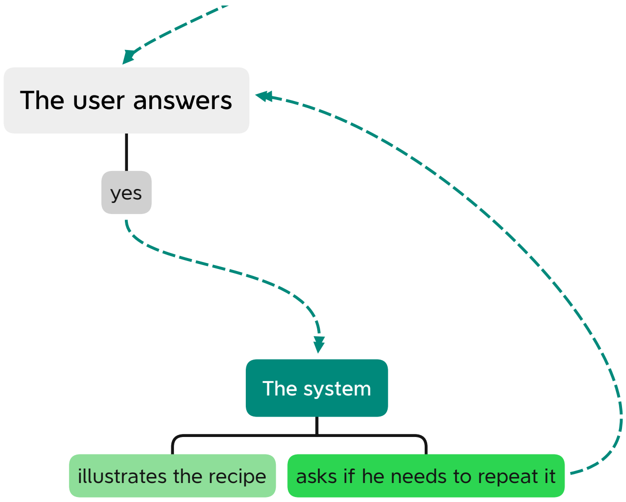

Figure 5.8. Seventh Intent: Repeating the Recipe

5.1.9 Last Intent: Concluding the Process

Being welcomed at the beginning of the interaction is as important for a good UX to being greeted once the conversation is over. In human-human interactions, there might be less need of an explicit goodbye since the 50% of communication is nonverbal and people may simply nod each others, but of course this doesn't apply to a human-VCA interaction as the interchange is verbal-only. Indeed, users need feedbacks when concluding the conversation with an agent for two reason: (i) to avoid reinforcing misconceptions and trust issues towards VCAs (as stated in chapter 3) and (ii) for usability reasons, because, as stated many times, the users has to be informed of the state of the system. For all these reasons, it is necessary to implement a "dummy" intent that allows the user to end the chat. In this last intent the users signals the system that the conversation is over and the system acknowledge its end by greeting the user.

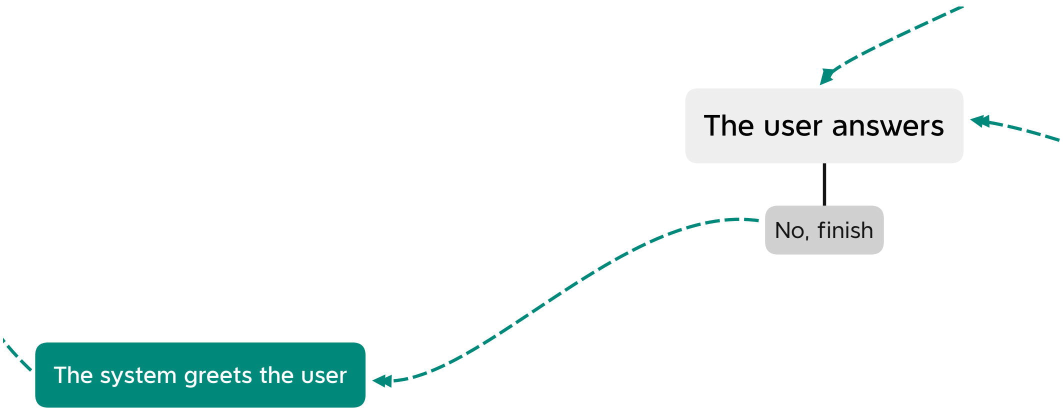

Figure 5.9. Last Intent

5.2 Peer Reviewing

The peer review will consist of a usability analysis of the first prototype. In particular, the reviewer will interact with the system in order to find possible flaws in the implemented conversation sequence.

The outcome of the review is the following. The overall usability of the system was positive, the VCA was more than enough compliant with the best practices for enhancing usability. Nielsen's heuristics were well-implemented along the conversational flow. The author of the review expressed encouraging words regarding the usability of the project although noticing minor inconveniences that will be addressed in the following section of the chapter. In particular, he stated that in order to achieve an even better outcome, the user should be more prepared when

suggested with a recipe. To accomplish this, the solution proposed was to permit to have a picture of the dish in some way to give a better insight of the recipe's final result.
Other than that, the reviewer suggests to add a following system functionality: since its research in obesity prevention indicated that an effective way to limit overweight is to know what to eat every day (and as side effect optimizing the use of time for grocery shopping), the bot should let the user know – when asked to do so – what is a healthy weekly meal plan the user could follow. He also noted that this approach follows Nielsen's Heuristics by making the user more aware of the internal state of the system. Indeed, we worked under the assumption that the VCA is able to remember past and future calories intakes, thus having the capability to generate a weekly eating plan to follow.
According to the reviewer the major flaws were how the ingredients or the recipe's steps were listed. In particular, the system would only state the ingredients and the entire recipe. If this seems sufficient from a requirements' point of view, it is a poorly designed way to present a list of elements through a VCA. Indeed, these can be considered actual usability defects, since in this way the users is forced to recall long complex lists, heavily damaging the UX with negative emotions such as stress and frustration.
Along with these issues, the reviewer pointed out the it is not a good approach to change a recipe every time the user doesn't have an ingredients. Indeed, the risks is to keep cycling with the third, the fourth and the fifth intent[4] for a long time. Instead, the reviewer assessed that it would be preferred to directly ask what ingredients the user have after the first missing ingredient.

5.3 Prototype 2

In this section, we will discuss how we have applied the suggestions given during the peer review in the project. Moreover, we have created a new prototype obtained by introducing or improving the intents implemented in the first one.
As it was not possible to include a visual content that presents the recipe in a pure VCA, we overcome this issue in two ways: the agent would show the picture if the device it was deployed has a screen, otherwise for users who requested it, we added a Telegram Bot linked to the agent that could display the picture of the dish (as shown in figure 5.10).

[4]where the user first agrees on a recipe (third intent), then hears the missing ingredients (fourth intent) and asks to change the recipe (fifth intent).

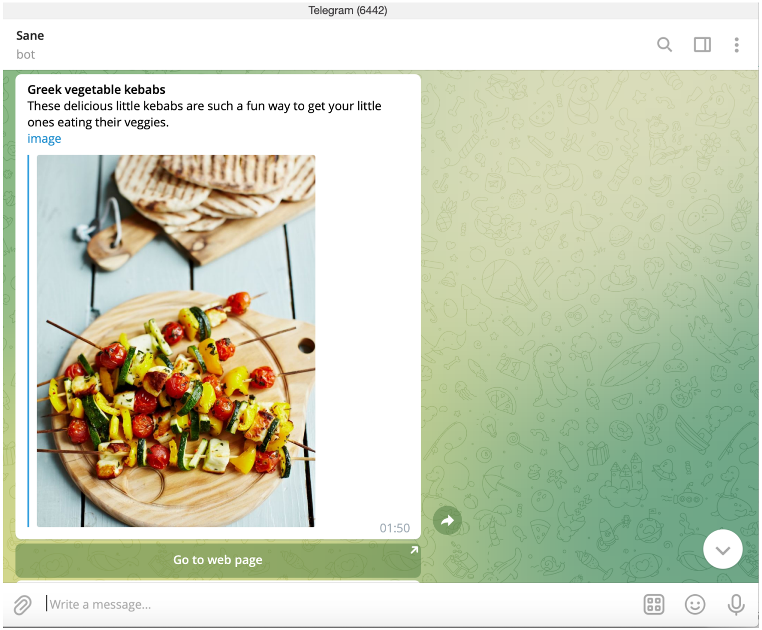

Figure 5.10. Prototype 2, Telegram bot for displaying the recipes

5.3.1 New Functionality: Weekly Plan

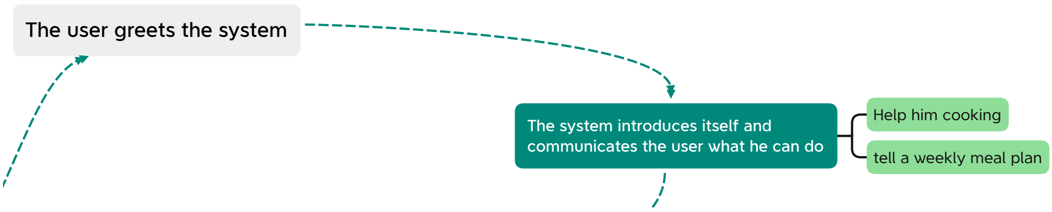

Figure 5.11. Prototype 2, new “Welcome” Intent

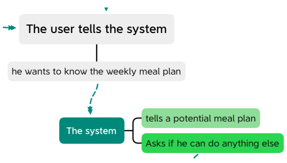

Figure 5.12. Prototype 2, new “Show Weekly Meal Plan” intent

The system needs to be able to present a weekly meal plan to the user. This leads to modifying the Welcome intent that now has to include the new functionality.

Along with that, we need to create a new intent with new user and system interactions. These intents are shown in figures 5.11 and 5.12.

5.3.2 Corrections of Recipes and Ingredients’ Listing Intents

In order to improve usability in listing the recipes’ steps or ingredients, we need to adjust their respective intents which are reported in the subsections 5.1.5 and 5.1.8. Instead of creating a new intent for each step or ingredient, the system considers them as parameters and waits for the user's feedback before enunciate the next item on the list, as we can see from figure 5.13 .

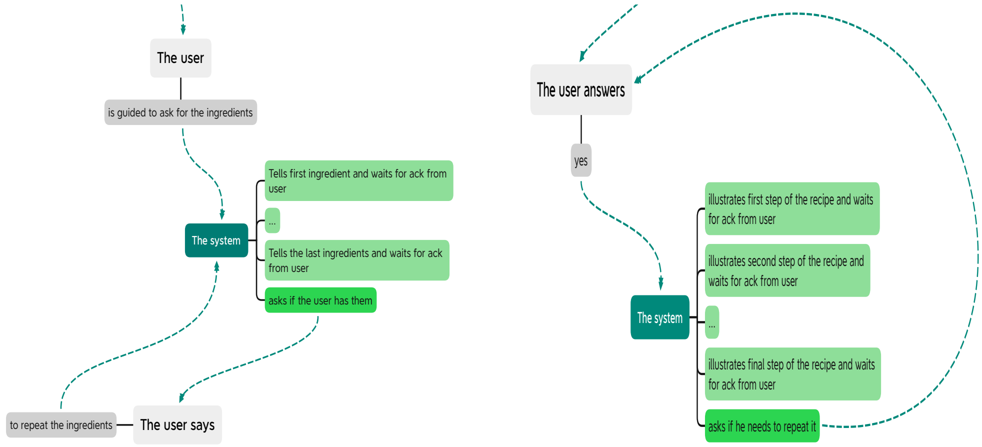

Figure 5.13. Prototype 2, (a) new "System Listing Ingredients" and (b) "Illustrating Recipe" intents

Having introduced all the requested features and improvements which have been discovered during the peer review, we have a new prototype to be user-tested before delivering the final prototype.

5.3.3 New Intent: User Telling Ingredients

The last correction we need to apply is avoiding to cycle for changing recipes due to a missing ingredient. We obtain this by introducing a new intent, where the user communicates the system what ingredients he has, as system parameters to filter recipes. Notably, this intent can be triggered both by the user (i) communicating the will to tell the ingredients or (ii) not confirming when prompted so by the system, as depicted in figure 5.14.

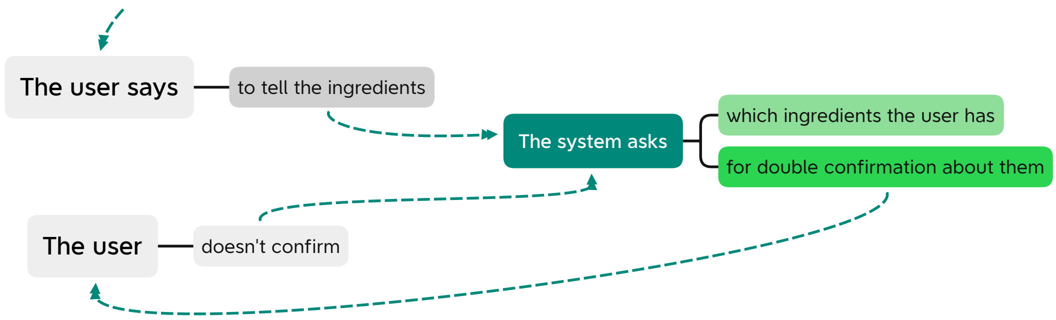

Figure 5.14. Prototype 2, new "User Listing Ingredients" intent

5.4 User Testing

The user testing has been conducted on a sample of three users. Other than simply measuring the usability and the quality of the UX, the purpose of this testing is to understand if there are small use cases belonging to our broad use case that we haven't highlighted.
The first part of the testing was focused on empirically confirming the results and the predictions we obtained during the peer review. We asked the participants to test both prototypes and tell us the strengths and weaknesses of both. The consensus was unanimous in ranking the second prototype's UX more enjoyable than the other's. Indeed, they declared that it was truly a difficult task to cook when the bot would simply state the entire recipe without repeating any step or waiting for the user to achieve the task. Obviously, the same motivations applied for when the chatbot listed the ingredients just by reading the full list.
Still concerning the ingredients, the participants approved the idea of adding the possibility to directly tell the system what ingredients they had at home instead of waiting until it would suggest them a feasible recipe with the groceries they had.
In addition, users evaluated the new dishes' image feature (whether it was with the Telegram bot or with the screen of the device) better than simply hearing the title of the recipe. Moreover, they asserted that having the possibility to judge the image before, gave them a better perspective on either approving the suggested recipe or not.
The last considerations about the comparison between the two prototype's UX are related to the feature of simulating a weekly meal plan. Although they added that it could work fairly well when cooking for a single person, when doing it for more people it could be hard to follow.
They did however spotlight some minor features that could be added to make the conversation flow less constraining. For instance, one suggested to add an interaction when people are cooking for guests. Even better, a user suggested to make the VCA to ask to read the complete recipe before actually starting to cooking thus giving the chance to anticipating a step of the recipe or simply preparing the user to the complexity of the recipe.

5.5 Final Prototype

Since the user testing confirmed the value of the work, there remains very few improvements to make for our use case. We will implement the suggestion made by the user in order to produce the final prototype. Despite of showing the singular intents modified, we will provide at the end of this section the final conversation flow, adjusted after the review and the testing.

Adding the guests' option to our use case is very straight forward. We will add the possibility for the user to tell the system that he wants to cook for guests. Although, since a study of the profiling stage lies outside of the scope of this work, the system will behave like it would if the user said "just me" (i.e., the system will directly propose a recipe).

The second improvement advanced during the testing is to read the entire recipe before actually starting to cook.

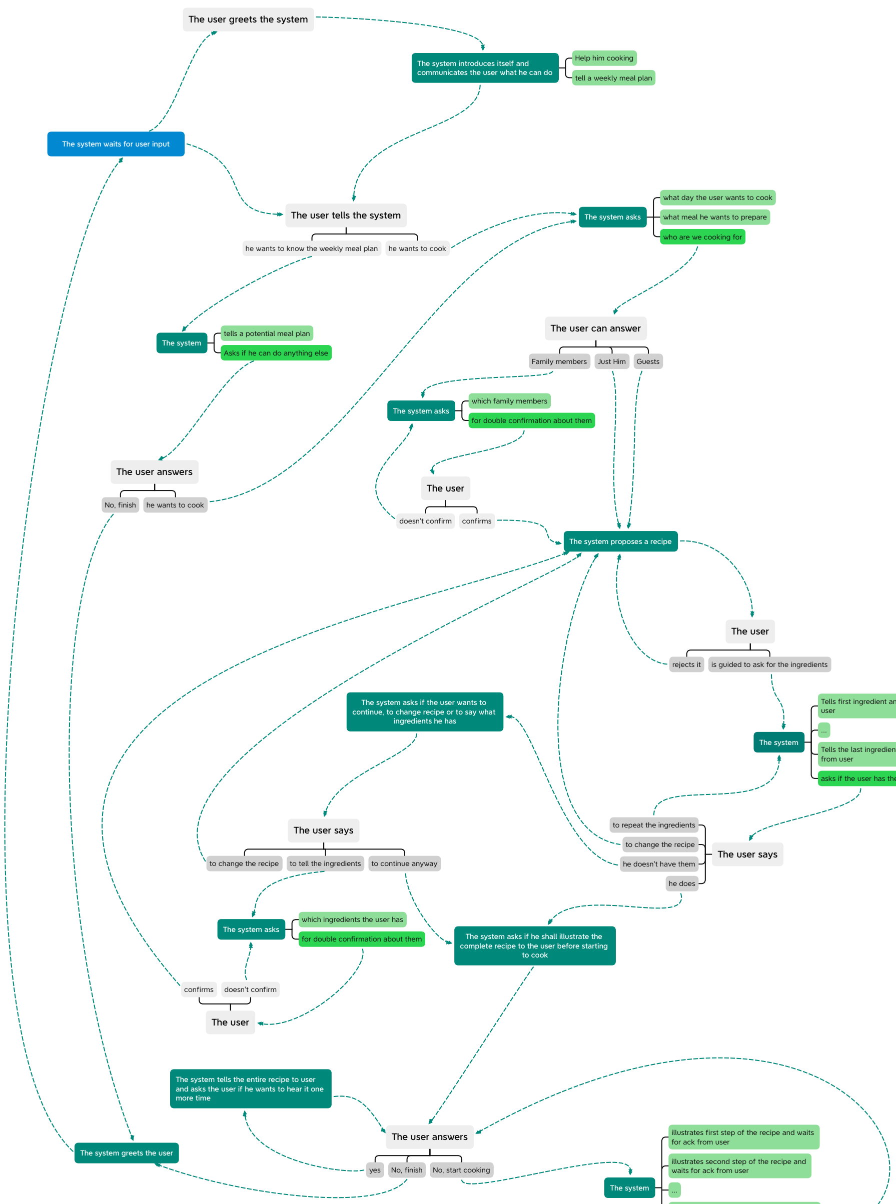

Figure 5.15. Final Prototype's Conversation Flow

Chapter 6

Conclusions

The aim of this work was to investigate the usability of a voice-based conversational agent as well as proposing a practical methodology to implement it.
After introducing the issue we aimed to solve, it has been shown a comparison between commercial frameworks and how they can be employed to achieve this goal. We further inspected their underneath architectures, the projected costs and downsizes along with the overall benefits each one provides. Moreover, it has been presented an implementable architecture to design the VCA.
If from a technical point of view there is clarity in defining factors which can be measured in a way or another, it is far more ambiguous getting a hold of principles and heuristics of usability-related issues. Nonetheless, we presented state-of-art studies in order to delineate the foundations based on which implement a usable VCA, taking into account the complex nature of the system concerning this issue. In fact, as we proceeded reporting the adoption of this technology, it was possible to show the numerous hindrances that keep it apart from a complete integration in every day life. Misconceptions and trust issues as well as intrinsic difficulties linked to the interaction with a non-human voice are just some of the obstacles intelligent assistants' developers need to face when designing such services.
In this work we exposed the delivery of a conversation flow, trying to be as coherent as possible with the modern best practices regarding VCAs. In particular, we detailed how we meant to prevent (underage) obesity through promoting healthy eating habits via a VCA. We then reviewed and tested an easily implementable conversation flow, under particular assumptions such as earlier profiling stages and memory of past interactions. The former, in particular, is critical and any developer who intend to use this work should take in consideration to implement it in the initial phases of the project. We also emphasize that when delivering a voice-based service, it's required greater effort in modeling the human-agent interaction due to its vague and dynamic nature in comparison to traditional computer systems.

Hence, methodologies such as design thinking process are recommended since they allow to cut the ambiguity at every iteration.
Considering the scope's magnitude of preventing obesity through a VCA, it was likely that numerous points remained open. We although consider our aim to increase usability as well as improving the user experience of the agent achieved through the employment of contemporary standard means of development related to this field.

Bibliography

[1] OECD, *The Heavy Burden of Obesity*, OECD Publishing, Paris, 2019.

[2] Spinelli, Angela et al., *Prevalence of Severe Obesity among Primary School Children in 21 European Countries.*, Obesity facts vol. 12,2 (2019): 244-258.

[3] OECD, *CO1.7 Overweight and obesity at ages 11, 13 and 15, by gender*, OECD Family Database, 2016.

[4] Kansra, Lakkunarajah, Jay, *Childhood and Adolescent Obesity: A Review*, Frontiers in Pediatrics, vol. 8, 2021.

[5] Sahoo, Krushnapriya et al. *Childhood obesity: causes and consequences.*, Journal of family medicine and primary care, vol. 4,2 187-92, (2015).

[6] Levitikos, Chatzigiannakis, Giambattista, *A virtual assistant for the prevention and management of obesity and overweight condition in school aged kids and teenagers*, Master's Thesis, "La Sapienza" University of Rome, 2021.

[7] Zhang, Jingwen et al., *Artificial Intelligence Chatbot Behavior Change Model for Designing Artificial Intelligence Chatbots to Promote Physical Activity and a Healthy Diet: Viewpoint.*, Journal of medical Internet research vol. 22,9 e22845. 30 Sep. 2020.

[8] Ermolina, Alena, Tiberius, *Voice-Controlled Intelligent Personal Assistants in Health Care: International Delphi Study.*, Journal of medical Internet research vol. 23,4 e25312. 9 Apr. 2021.

[9] O'Brien, Katherine et al. *Voice-Controlled Intelligent Personal Assistants to Support Aging in Place.* Journal of the American Geriatrics Society vol. 68,1 (2020): 176-179.

[10] Bérubé, Caterina et al. *Voice-Based Conversational Agents for the Prevention and Management of Chronic and Mental Health Conditions: Systematic Literature Review.* Journal of medical Internet research vol. 23,3 e25933. 29 Mar. 2021.

[11] Sezgin, Emre et al. *A scoping review of patient-facing, behavioral health interventions with voice assistant technology targeting self-management and healthy lifestyle behaviors.* Translational behavioral medicine vol. 10,3 606-628, 2020.

[12] Schachner, Theresa et al., *Artificial Intelligence-Based Conversational Agents for Chronic Conditions: Systematic Literature Review.*, Journal of medical Internet research, vol. 22,9 e20701. 14 Sep. 2020.

[13] Oh, Yoo Jung et al. *A systematic review of artificial intelligence chatbots for promoting physical activity, healthy diet, and weight loss.* The international journal of behavioral nutrition and physical activity vol. 18,1 160. 11 Dec. 2021

[14] Roshan, *Standardized Architecture for Conversational Agents a.k.a. ChatBots.* International Journal of Computer Trends and Technology. 50. 114-121. 2017.

[15] Google Inc., *Dialogflow Documentation*, 2021.

[16] Microsoft, image website from https://dev.botframework.com/

[17] Siddartha, *Top Chatbot Development Frameworks You Should Know – Exploring Paid Solutions (Part 1)*, https://thirdeyedata.io/top-chatbot-development-frameworks-you-should-know-exploring-paid-solutions-part-1/, Dec. 2020.

[18] Handy, *Amazon introduces Lambda, Containers at AWS re:Invent*, 2014.

[19] Warwick, Nelson, Jetty, *Integrating Amazon Connect and Amazon Lex with Third-party Systems*, Jul 2021.

[20] Rasa, *Rasa Architecture Overview*, Rasa documentation, 2022.

[21] IBM, https://www.ibm.com/products/watson-assistant/pricing/, 2022.

[22] IBM, https://www.ibm.com/cloud/architecture/architectures/aiAnalyticsArchitecture/reference-architecture, 2022.

[23] Mator, Janine D et al., *Usability: Adoption, Measurement, Value.*, Human factors vol. 63,6. 2021.

[24] Nielsen, *Usability 101: Introduction to Usability*, https://www.nngroup.com/articles/usability-101-introduction-to-usability/, 2012.

[25] Budiu, *Interaction Cost*, https://www.nngroup.com/articles/interaction-cost-definition/, 2013.

[26] Norman, Nielsen, *The Definition of User Experience (UX)*, https://www.nngroup.com/articles/definition-user-experience/

[27] Budiu, Laubheimer, *Intelligent Assistants Have Poor Usability: A User Study of Alexa, Google Assistant, and Siri*, https://www.nngroup.com/articles/intelligent-assistant-usability/, 2018.

[28] Budiu, Whitenton, *What Could an Intelligent Assistant Do for You? A Diary Study of User Needs*, https://www.nngroup.com/articles/intelligent-assistant-user-needs/, 2018.

[29] Laubheimer, Budiu, *Intelligent Assistants: Creepy, Childish, or a Tool? Users' Attitudes Toward Alexa, Google Assistant, and Siri*, https://www.nngroup.com/articles/voice-assistant-attitudes/, 2018.

[30] Whitenton, Budiu, *The Paradox of Intelligent Assistants: Poor Usability, High Adoption*, https://www.nngroup.com/articles/intelligent-assistants-poor-usability-high-adoption/, 2018.

[31] Budiu, *The User Experience of Chatbots*, https://www.nngroup.com/articles/chatbots/, 2018.

[32] Ren, Castro, Acuña, Lara, *Usability of Chatbots: A Systematic Mapping Study*, 479-484. 10. 2019.

[33] Nielsen, emph10 Usability Heuristics for User Interface Design, 1994.

[34] Kirkpatrick, O Connor, Campbell, Cooper, *Web content accessibility guidelines (WCAG) 2.1.*, W3C Recommendation, June 2018.

[35] Lister, Coughlan, Iniesto, Freear, Devine. emphAccessible conversational user interfaces: considerations for design. In Proceedings of the 17th International Web for All Conference (W4A '20), Association for Computing Machinery, New York, NY, USA, Article 5, 1–11, 2020.

[36] Harms, Kucherbaev, Bozzon, Houben, *Approaches for Dialog Management in Conversational Agents*, in IEEE Internet Computing, vol. 23, no. 2, pp. 13-22, 1 March-April 2019.

[37] Hussain, Ameri Sianaki, Ababneh, *A Survey on Conversational Agents/Chatbots Classification and Design Techniques.* In: Barolli, Takizawa, Xhafa, Enokido, (eds) *Web, Artificial Intelligence and Network Applications. WAINA 2019*, Advances in Intelligent Systems and Computing, vol 927. Springer, Cham. 2019.

[38] Clark et al. *What Makes a Good Conversation? Challenges in Designing Truly Conversational Agents.* Proceedings of the 2019 CHI Conference on Human Factors in Computing Systems, Association for Computing Machinery, New York, NY, USA, Paper 475, 1–12, 2019.

[39] Cowan et al. *What Can I Help You With?: Infrequent Users' Experiences of Intelligent Personal Assistants.* In Proceedings of the 19th International Conference on Human-Computer Interaction with Mobile Devices and Services, 2017.

[40] Porcheron, Fischer, Reeves, Sharples. *Voice Interfaces in Everyday Life.* In Proceedings of the 2018 CHI Conference on Human Factors in Computing Systems, 2018.

[41] Norman, *The Design of Every Day Things – Revised and Expanded Version*, Basic Books, Philadelphia, USA, 38-40, 2013.

[42] Luger, Sellen. 2016. *"Like Having a Really Bad PA": The Gulf between User Expectation and Experience of Conversational Agents.* In: Proceedings of the 2016 CHI Conference on Human Factors in Computing Systems(CHI '16). Association for Computing Machinery, New York, NY, USA, 5286–5297. 2016.