



SCHOOL OF ECONOMICS AND MANAGEMENT

Master's Programme in Data Analytics and Business Economics

A framework for implementing customer-service chatbots:

The case for LUZ shopping centre in Jerez, Spain

by Ana Clara C. F. de Lima

DABN01

Master's Thesis (15 credits ECTS)

May 2024

Supervisors: Antonio Marañon & Peter Gustafsson

Abstract

With the advancement of artificial intelligence, customers and users in general have been more used to having a virtual assistance method for service-related tasks. However, in the shopping centre industry, the self-help solutions mostly follow the traditional pattern of having a concierge or a totem where you can look for the location of a particular store. This thesis aims to look at this with a different approach, by proposing a chatbot that can be integrated in the shopping centres and prepared to recommend stores based on a product search. To create the required database for product search, web-scraping techniques were applied, combined with natural language processing to make the inputs machine-readable. The chatbot was created based on a set of rules and conversational scenarios common to a self-help desk solution.

The evaluation of the proposed method was based on expert interviews for feedback. The main qualities of the model are its straightforward use, and clear communication style. Regarding the limitations of the model, the most commonly mentioned were the robotic response style, and the absence of the capabilities of providing a store map after the recommendation. Potential improvements that could be adopted are a rank of the suggested stores, based on experts' inputs, and link the outputs of the conversation with discounts and promotions. However, even with the limited current setup, all experts saw value in the implementation of the model, especially in the shopping centre website. Expected positive effects include more visitation to the facility and more visibility for smaller stores.

Keywords: Shopping Centre, Chatbot, Web-scraping, Natural Language Processing, Expert Feedback

Acknowledgements

First of all, I would like to thank the Swedish Institute for providing me with an opportunity to pursue a Master's in such a renowned institution as Lund University. Having a scholarship as this one made this dream a possibility. Additionally, I would like to express my gratitude for my family, partner and friends, both in Brazil and in Sweden, who always showed their unstoppable support towards me. I could not have made it without you!

Contents

1. Introduction	7
1.1. Research Questions	7
1.2. Thesis' Structure	8
2. Literature Review	9
2.1. Web-Scraping.....	9
2.2. Chatbots: history, development, evaluation, and applications.....	9
2.2.1. History	9
2.2.2. Development	10
2.2.3. Evaluation.....	12
2.2.4. Applications.....	12
3. LUZ Shopping Centre Bot Framework.....	14
3.1. LUZ Shopping overview.....	14
3.2. Store and products database.....	14
3.2.1. Natural Language Processing.....	17
3.3. Chatbot Development	18
4. Expert Feedback Evaluation	24
4.1. Individual results from the interviews	25
4.2. Summary of Interviews.....	30
5. Conclusion	32
5.1. Limitations	33
5.2. Future Research.....	33
References.....	35
Appendix A.....	37
Appendix B.....	38
Appendix C.....	39

List of Figures

<i>Figure 3.1: Flowchart of LUZ Shopping chatbot set of rules for response.....</i>	<i>19</i>
<i>Figure 3.2: Example of chatbot conversation when the user's inputs are greetings or ending the conversation.....</i>	<i>19</i>
<i>Figure 3.3: Example of chatbot conversation considering different product types</i>	<i>21</i>
<i>Figure 3.4: Example of chatbot conversation with unexpected inputs.....</i>	<i>22</i>
<i>Figure 3.5: Example of expected chatbot conversation.....</i>	<i>22</i>

List of Tables

<i>Table 2.1: Chatbots' categories.....</i>	<i>11</i>
<i>Table 3.1: Set of rules specified on web-scraping mechanism for product by store retrieval</i>	<i>15</i>
<i>Table 3.2: Web-scraping mechanism set for store names and categories retrieval from LUZ Shopping Website.....</i>	<i>16</i>
<i>Table 4.1: Compilation of the responses provided by the participants of the interviews, for the objective questions.</i>	<i>30</i>

1. Introduction

With the advancement of technology, especially with regards to artificial intelligence, chatbots are rising in importance and their usage has been growing to the most diverse domains, as healthcare, customer service and education (Limna *et al.*, 2023). In this paper, we developed a framework for the application and usage of chatbots within the shopping centres industry. This thesis was written in conjunction with Ingka Centres, a company part of the Ingka Group, that is responsible for managing Ikea-owned shopping centres globally.

The most common self help desk kiosks present in shopping centres have search-engines that are ‘store-based’. This means that when you arrive at one of these points, you can search for a specific store and the system will provide you with its location. Our development, on the other hand, proposes a product-based search engine, where customers can search for a product instead, and the chatbot will provide them with store recommendations.

For the use case of this study, we focused on LUZ Shopping, a shopping centre located in Jerez, Spain. The solution was customised for the usage in this specific facility. The store database was created using web-scraping techniques, for the stores present in this shopping centre, natural language processing was applied in the database to facilitate product search, and the developed chatbot access to the database for providing store recommendations.

For measuring the performance of the chatbot, we have executed expert interviews. Those were divided into two sections of questions: the first one being a rank, from 1 to 5, on specific aspects that are found in the literature being the most relevant for chatbot’s evaluation. In the second one, we collected qualitative feedback on the tool, to understand what is working well in the current development, and what could be incorporated to make it even more valuable for the industry.

1.1. Research Questions

The main problem being addressed in this thesis is to develop a sustainable, easy replicable framework for the deployment of a product-base search chatbot, able to provide reliable store recommendations for customers real-time. The research covers the following aspects:

1. Development of a web-scraping application, that can successfully create a product by store database, used for providing recommendations.
2. Development of a chatbot that, based on user inputs, can effectively recommend stores based on product search.
3. Expert interviews with Ingka Centres employees to evaluate the solution over different aspects.

1.2. Thesis' Structure

This paper is structured in the following way: in Chapter 2, we review the existing literature on web-scraping techniques, and the history, development, and current study questions on chatbots. Then, in Chapter 3, we showcase the tailored solution development for LUZ Shopping. Chapter 4 contains the discussions on expert feedback, and their review on the application itself. Finally, on Chapter 5, the conclusions of this study are presented, along with its limitations and recommendations for future research.

2. Literature Review

2.1. Web-Scraping

Henry (2021) defines web-scraping in the following way: “web scraping, also known as data mining, is the process of collecting large amounts of data from the web and then placing it in databases for future analysis and later use”. Even though this method has been around ever since the internet was invented, it is still a powerful way to collect and organise information extracted from the web and has applications into a varied range of fields (Glez-Peña *et al.*, 2014).

You can perform web-scraping through different sets of techniques or programming languages. One option is to use Python programming language, and more specifically the BeautifulSoup library (Abodayeh *et al.*, 2023). BeautifulSoup is a library that is used to extract information from HTML files, which corresponds to the majority of websites online, and it has pre-built-in functions to save work time from whoever is programming it (Richardson, 2014).

After the content is scraped, the usage of the data can go beyond the intent of the website owner (Henry, 2021). The objective of web-scraping is to analyse and store this data to make it possible to use it in a machine-readable way. What is the usage of the data, and what insights will be extracted from it, are the programmer’s choice (Glez-Peña *et al.*, 2014).

2.2. Chatbots: history, development, evaluation, and applications

2.2.1. History

According to the Oxford English Dictionary (2023), a Chatbot can be defined as “A computer program designed to simulate conversation with a human user, usually over the internet; ... used to provide information or assistance to the user as part of an automated service.”. Many argue that the concept of chatbots was developed by Alan Turing and the imitation game, where he believed that it was possible to have a conversation without the interlocutor realising if they were speaking with a human or a machine (Adamopoulou & Moussiades, 2020). The main question brought by Alan Turing was “can machines think?” (Turing, 1950), and while this is still not yet settled nowadays, for sure we have had many advancements in this area.

The first recognized chatbot was developed in 1966, by Joseph Weizenbaum, was called ELIZA (Cahn, 2017). ELIZA was set to replicate a psychotherapy human conversation, by rephrasing the inputs provided by the users (Adamopoulou & Moussiades, 2020). Ever since, the chatbots framework has become much more developed, with the biggest representation of this advancement being ChatGPT. Launched by Open AI in 2022, it uses deep learning mechanisms and reinforcement learning to provide inputs based on natural language processing (Open AI, 2022). It can capture long-term dependencies, which makes it a very powerful tool for context interpretation (Franciscu, 2023). This is the current benchmark for many artificial intelligence applications, and different usages of ChatGPT have been registered in different industries (Limna et. al, 2023).

2.2.2. Development

Chatbots can be developed through an extensive list of techniques. Peters (2018) showcases the development of a chatbot used for customer support using natural language processing (NLP), combined with neural networks, in an approach that we can refer to as deep learning. This is similar, yet simpler, than the approach adopted by Zheng *et al.* (2023), that developed a NLP chatbot using multi-headed deep neural networks focused on improving profits through marketing within companies. One can state that leveraging NLP and deep learning techniques are the most common approaches for chatbot development currently. Also, both in Peters (2018) and in Zheng *et al.* (2023) the chatbots were developed using Python, a very common programming language for this field.

However, when developing a chatbot, it is very important to have in mind what characteristics it will have. In Table 2.1, Adamopoulou & Moussiades (2020), compiled a summary table that exemplifies the thought-process the developer needs to follow when creating a chatbot, choosing what are the capabilities they want the bot to have.

Chatbot Categories	Knowledge domain	Generic
		Open Domain
		Closed Domain
	Service provided	Interpersonal
		Intrapersonal
		Inter-agent
	Goals	Informative
		Chat based/Conversational
		Task based
	Response Generation Method	Rule based
	Retrieval based	
	Generative	
Human-aid	Human-mediated	
	Autonomous	
Permissions	Open-source	
	Commercial	
Communication channel	Text	
	Voice	
	Image	

Table 2.1: Chatbots' categories

Source: Adamopoulou, E. & Moussiades, L. (2020). *Chatbots: History, Technology, and Applications*.

A generic chatbot can answer questions related to any domain, while an open domain chatbot can navigate between two or three domains, and a closed domain can only respond regarding one topic. Interpersonal chatbots can render services for its users, while intrapersonal are domain-experts in their users' needs and inter-agent can communicate with other chatbots. The goals of a chatbot can differ between informing users about something (e.g. the status of an order), or even keeping a conversation with the user (conversational) or assisting in completing a task (task-based). The response method can be set up being rule-based (pre-defined by the programmer), generative - when the bot can generate, by itself, some combination of words for an appropriate response, or retrieval-based, when it combines pre-trained models with information retrieval techniques (Adamopoulou & Moussiades, 2020).

When it comes to human-aid, there are two potential types of chatbots: the ones that need a human to mediate the conversation, and the autonomous type, meaning that it runs

without user interference after it's made available for the public. Open-source permissions mean that the bot can be accessed openly, without any restrictions from its developers, and the commercial permission refers to chatbots that are being used with commercial intent. Finally, there are mainly three communication channels for chatbots to communicate: via text, very commonly used in web-versions, as ChatGPT for example. Via voice, which is a common format for the so-called 'virtual assistants' as Alexa or Siri. Also, via images, where the bot can display images in a conversational format or as an output, as in Dall-E (Adamopoulou & Moussiades, 2020).

The discussion on the category of the chatbot is crucial for its development. Because, depending on the capacities the developer wants the bot to have, a different set of techniques are required on its conception.

2.2.3. Evaluation

After the development and before the deployment of any artificial intelligence solution, it is crucial to test it and evaluate if it is reacting as expected. One framework that can be used for assessing a chatbot is the PARADISE (Paradigm for Dialogue System Evaluation), which considers six different domains: "(i) ease of usage, (ii) clarity, (iii) naturalness, (iv) friendliness, (v) robustness regarding misunderstandings, and (vi) willingness to use the system again" (Cahn, 2017). For doing that, there is a need to rank the bot capabilities in all of them, based on interviews with customers or experts.

Adiwardana *et al.* (2020) propose a much simpler approach: that chatbots need to be specific and make sense. Humans would be the judges of these two criteria. Additionally, a different way of measuring effectiveness could be the usage of confusion matrices (Cahn, 2017) and regular machine learning metrics as precision, accuracy, F1-score and recall (Zheng *et al.*, 2023, Peters, 2018) depending on the type of the problem the chatbot is dealing with, and the output of the model.

2.2.4. Applications

There are extensive applications for chatbots currently. Tran *et al.* (2021) explored the perception of consumers when interacting with chatbots in the retail industry, by reviewing their sentiments regarding the application. They found that the way the company introduces

the technology can severely affect how customers feel about the chatbot. Also, when effectively set, the experience of being supported by a chatbot can reduce the satisfaction with human interactions as the expectations for this last one are increased significantly (Tran *et al.*, 2021). This aligns with the expectations that users usually react better to chatbots and other self-service technologies when they find it to be convenient (Collier & Kimes, 2013).

Chung *et al.* (2020) explored the usage of chatbots and virtual assistants in the luxury retail segment, focusing on enhancing customer experience within the brand. In this case, the virtual agents would represent an additional layer of customer support, generally used to save time, or assist in the decision-making process (Chung *et al.*, 2020). Chatbots, and especially ChatGPT, can also be used in the education sector. Students mention that leveraging it as a tool that can serve as an assistant tutor, always available, while teachers mention that it can free up some of their time outside the classroom, by replying to simple inquiries from the students that would come to them if the technology was not in place (Limna *et al.*, 2023).

3. LUZ Shopping Centre Bot Framework

3.1. LUZ Shopping overview

LUZ Shopping is located in the suburbs of Jerez, a city in the region of Andalucía, in southern Spain. It was opened in 2010 and has over 123.000m² of leasable area. It is an open-air mall concept, which makes it unique compared to other properties owned by Ingka Centres, and it combines outlet and premium stores in the same facility (Ingka Centres, n.d.). Currently, there are 53 different stores, 18 restaurants and 7 playground areas in LUZ's facilities (LUZ Shopping, n.d.).

Jerez is a city with approximately 212.000 citizens, located in the western area of the Andalucía state. The municipality extends over 1.186km², and it is 597 km distant from Madrid, the capital of Spain (Ayuntamiento de Jerez, n.d.). It is the biggest city in the province of Cadiz, and the main driver of the city's economy is the wine industry, especially known for Sherry wine (Wikipedia, 2024).

3.2. Store and products database

The first step to create a chatbot that can recommend stores by searching for a specific product, was to develop a database that could be able to capture this information at the particular level of detail. To achieve this, web-scraping techniques were used to collect the following type of information:

1. Store names and main categories (fashion, home, and decor, etc.) were web-scraped from the LUZ Shopping Centre website.
2. Product details lists were extracted from the stores' websites directly.

It is important to mention that not all stores have functional websites, also in some situations where they did, the website could not be web-scraped due to security configurations. Because of that, for some general categories, such as fashion, one store was used as a product proxy. Meaning that the product list extracted for one website was inputted for all (or many) stores in that category. This method was applied after careful consideration and direct feedback from the shopping centres' support team, who understands the tenants deeply. However, in general, we can say that the web-scraping followed the process described below in Table 3.1.

<p>Define: scrape products as a function</p> <p>Require: URL or list of URLs from stores to web-scrape</p> <p>Require: user-agent setup to access HTML information</p> <p>Require: response from the website</p> <p>If response code = 200, means connection was established with the website</p> <p>Set up the soup argument, which enables python the capability of reading HTML</p> <p>Inspect web page to define what is the HTML element storing product information¹</p> <p>Create a repository adding all items found with specified HTML element</p> <p>Keep only unique entries in the repository, excluding duplicates</p> <p>Exclude undesired words²</p> <p>Split words that were combined with additive conjunctions³</p> <p>Return unique, split categories</p> <p>Else</p> <p>Return failed to fetch webpage, meaning the scraper was unable to work.</p>

Table 3.1: Set of rules specified on web-scraping mechanism for product by store retrieval

This method was used to extract the product list from store’s websites, to compose the product by store database. One of the steps of this process is to identify the HTML element storing product information. This is a manual task that needs to be performed by the programmer, and it can differ from website to website, as it will depend on the HTML hierarchy established for them. To define what stores would be scraped, first we reviewed all the stores present on LUZ Shopping Centre facility. There is a list present on their website, counting 53 different stores. As mentioned above, we used some stores as a proxy for other stores in the category. For stores in fashion womenswear, we used the Desigual store as a proxy for all stores. For some specific sportswear stores, such as Adidas and Puma, we used Nike as a proxy, as they mainly sell similar products. And in the case of menswear, Alvaro Moreno was a proxy for the same category stores. Ikea was the proxy for home and decor, which encompassed Jysk as well, and Pepco was a proxy for KIK, as they sell an extensive portfolio of products, from decor to clothing.

¹ Each page needed to be independently verified for the identification of the HTML object which was most informative in terms of product details. This identification is done by the programmer.

² Undesired words were selected by manually analysing the extract of each web-scraper output. There are common words such as ‘News’, ‘Deals’, ‘Offers’ or ‘Work with us’ that were excluded from the database to not correspond to any product detail category.

³ For some stores, categories were combined with an ‘and’ or ‘+’ sign. For the database to comprehend the most expressive number of products, those were split and kept in different cells.

Another manual adjustment required for the database to be functional is the case of stores that sell very specific products. There are five in this case: electronic cigarettes, motorcycles, trailers (mobile houses), solar panels and a gas station in the shopping facility. Those were set manually to report only one product as described before, as most likely someone looking for these products will not be searching for a specific model (level of detail contained on the websites), but the product itself.

Lastly, we also automated the scraping for store name and category, directly from the LUZ Shopping website. This required a combination of Python libraries, because of the format of the website. In the store webpage, the user needs to perform clicks on ‘show more’ buttons available to be able to capture them all. Precisely, you need to click the button three times to see all 53 stores. Then, for this task, we used the Selenium library available in Python along with BeautifulSoup. Selenium allows the code to interact with a web-browser simulating user engagement (Software Freedom Conservancy, 2023), which is useful for button-clicking, for example. So, the framework for scraping store names and categories is described in Table 3.2 and the codes for web-scraping are available in Appendixes A and B.

<p>Define: Scrape store names and categories</p> <p>Require: LUZ Shopping website URL</p> <p>Require: selenium web-driver that will be used to simulate the user action⁴</p> <p>Define: button clicking function</p> <p> When find the button containing the words 'show more', click⁵</p> <p> Wait for 5 seconds</p> <p> Look again for the button, and click</p> <p> Repeat the process up to 3 times⁶</p> <p>Define: Set up the soup argument</p> <p> Scrape for pre-defined HTML object</p> <p> Strip the text and store categories and store names into different columns</p> <p>Return store names and categories</p>
--

Table 3.2: Web-scraping mechanism set for store names and categories retrieval from LUZ Shopping Website

⁴ In this case, we are using Microsoft Edge.

⁵ The actual text being searched is ‘mostrar más’, which is Spanish for ‘Show More’. This was defined after reviewing the website manually.

⁶ This number was defined because of the number of the stores. Each page shows 24 stores, as we have 53 in total, we need to click three times to see them all in one page.

After collecting the required information from stores, product categories and product details, the last step was to organise it into a data frame. The product details column can be quite extensive, but it is more effective for information retrieval if each product has its own cell, that needs to be related to the store name and category. To achieve this objective, we repeated the rows for store name and categories for each product detail row. This approach resulted in a table with a size of 3150 rows and 3 columns. It is important to call-out that the number of rows of the database can vary if the stores present at LUZ Shopping change, as this can modify the type and number of products available for customers. This specific format represents the figure of May 2024. Additionally, the size of the database does not affect the final output of the model, as it is used for an information retrieval process.

After the data frame was finalised, we inspected it manually to understand what type of information it was storing. We identified that the product details column, as it was extracted directly from stores websites, had no standard naming convention: it mixed product names all in upper, lower, or mixed case depending on the store. Some contained stop words (words that do not bring any context). This can be tricky to handle effectively in Python, as it is a case-sensitive language model (Bird *et al.*, 2009). So, in order to have an effective database, we needed to organise information in all columns in a machine-readable way, using Natural Language Processing.

3.2.1. Natural Language Processing

Hirschberg & Manning (2019) define natural language processing (NLP) as being a science that learns, understands, and produces human-like linguistic content. In our specific case, we are not using natural language processing to learn or produce anything based on the dataset, but to standardise the database columns, making it easier for our program to perform its intent of recommending stores based on product search. So, we can state that in this thesis scope we are using NLP to understand linguistic content.

The challenges of using human-generated text on a computer-based system is the nature of this content: it is usually vast, unrestricted, and often ambiguous (Nadkarni *et al.*, 2011). To deal with that, we applied some techniques for standardisation of text available in the NLTK library in Python. This library is commonly used by programmers as it counts with a set of easy-to-use resources to process text (Bird *et al.*, 2009).

The first step we applied was to lowercase all words contained in the product details column, as Python is a case-sensitive language (Bird *et al.*, 2009), which means that it will classify the words ‘Blouse’ and ‘blouse’ as different words if the inputs are not lower-cased. Then, we tokenized our text, which means splitting it into a list of words and punctuation marks, if any (Bird *et al.*, 2009). We also used the pre-built stop words function from NLTK library, in Spanish. This set contains a predefined list of words that are very frequent and do not add context to the text being analysed, as prepositions, pronouns, and conjunctions (Bird *et al.*, 2009). Finally, the text was lemmatized, which corresponds to transforming the words to their root form (Nadkarni *et al.*, 2011). The root form of running, for example, is run. After NLP was applied to our product database, we could start the development of the chatbot as the words it will search were set in a machine-accessible way.

3.3. Chatbot Development

The first conceptual phase of the chatbot for product-based search was to define what type of characteristics we wanted the bot to have. As it is going to serve as a self-help for customers in the shopping centre facility, the goal is to provide information, not designed to keep long conversations or assist with tasks. The domain is closed to only the stores and products available in the shopping centre, and the responses are defined based on a set of rules created by the programmer. The idea was also that the bot needed to be autonomous, meaning it can look for responses automatically after the setup, and communicates with customers only via text, not needing to understand or produce other media types such as audio or images.

After the conceptual phase, we needed to determine what is the set of rules that guide chatbot response when triggered. These are represented in the process flow in Figure 3.1.

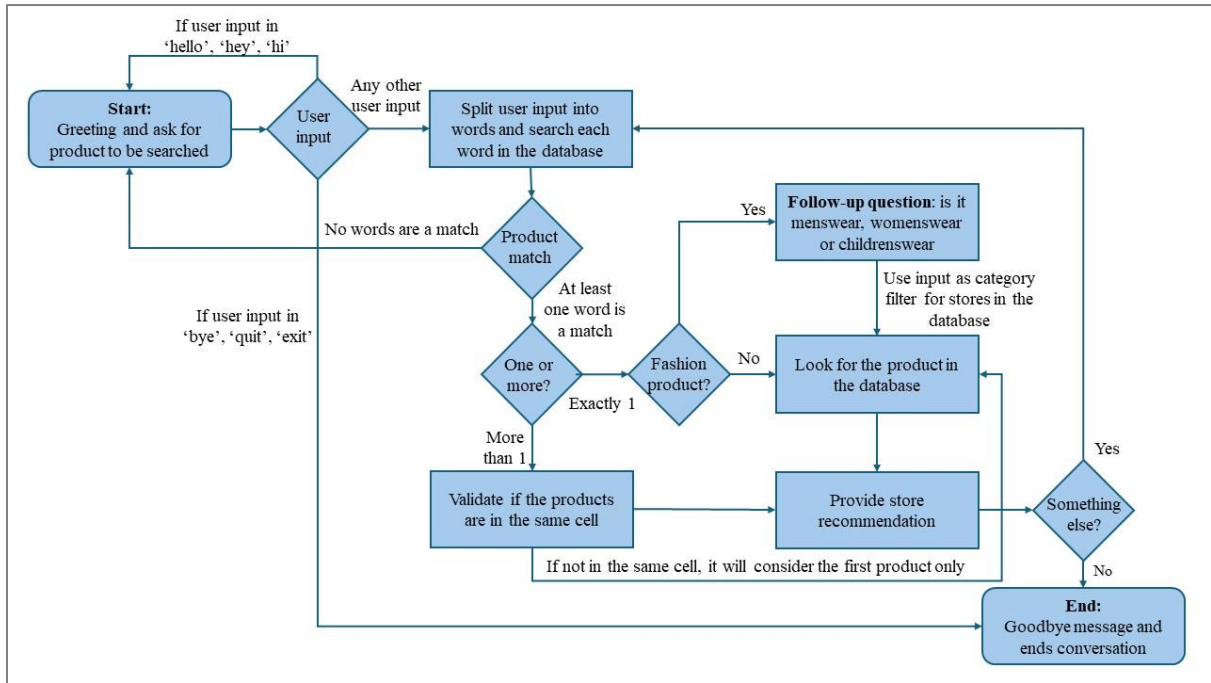


Figure 3.1: Flowchart of LUZ Shopping chatbot set of rules for response

The conversation is kicked-off by the bot, greeting the user and asking for what product they are looking for. One call out the bot does is that the user is supposed to look for one product at a time. If the user responds by greeting the bot back, it will repeat the same welcoming message, waiting for the user to input the product they are looking for. If the user responds with bye, quit or exit, the session will be automatically ended. Examples of both cases are shown in Figure 3.2.

```

Bot: Hello! What product are you looking for today? Please search for one product at a time.
You: hello
Bot: Hello! What product are you looking for today?
You: bye
Bot: Goodbye! Hope to see you soon.

```

Figure 3.2: Example of chatbot conversation when the user's inputs are greetings or ending the conversation

When the input is different from repeating a greeting or exiting the chat, the flow will continue. In this case, the first thing the bot does is to apply similar natural language processing mechanisms as it did to clean the database. So, it sets the user input to lowercase, remove stop words, and tokenize the text. Then, for each token, it identifies whether it is present on the product details column or not. Something that it is important to mention is that some words

that characterise products in English can be composed for more than one word in Spanish. An example would be sunglasses, which translates as ‘gafas de sol’. Because of cases like that, the bot validates for how many words it finds correspondence in the database. If more than one, it will search for a combination where it finds them in the same cell, to provide a recommendation that covers requests for products composed of more than one word. If the same cell combination is not found, it will provide a recommendation for the first word being searched (in our example, “gafas”, or glasses in English).

If the number of corresponding words is equal to one, it validates if the product belongs to the fashion category. This was a condition we added based on expert feedback, as there are stores that specialise on selling menswear, womenswear or childrenswear items. Consequently, depending on what category of fashion the user is looking for, it can significantly change the output of store recommendations. After validations are done, the bot randomly selects five stores (if it finds more than five stores that sell the specified product) and recommend them to the customer.

Finally, the last decision point the bot takes is to ask if the customer is looking for another product. If yes, it will ask for the user input, and start again from the third step in our process flow. If not, it will end the conversation. Also important to mention is that all the user inputs provided are stored for later analysis in a list variable. The complete code for the chatbot is provided in Appendix C.

In Figure 3.3 we demonstrate a conversational example that encompasses all the characteristics described in the paragraphs above. For now, the expectation is that the user will type in Spanish, as the database was configured this way. The chatbot's responses are still in English, to facilitate readers' comprehension, but when deployed it will be translated to Spanish.

```

Bot: Hello! What product are you looking for today? Please search for one product at a time.
You: gafas de sol
Bot: Here are some stores where you can find it:
Moda-re
El Corte Inglés Outlet
Alvaro Moreno
Levi's Outlet
C&A
Bot: Are you looking for any other product today? (yes/no) yes
You: yo busco por camisetas
Bot: Are you looking for moda masculina, moda femenina o infantil? You: femenina
Bot: Here are some stores where you can find it:
Fifty Outlet
Guess Outlet
Xti Outlet
Simorra Outlet
KIK
Bot: Are you looking for any other product today? (yes/no) yes
You: motocicletas
Bot: Here are some stores where you can find it:
MercaBox
Bot: Are you looking for any other product today? (yes/no) no
Bot: Ok, hope I was able to help. Bye!

```

Figure 3.3: Example of chatbot conversation considering different product types

In the conversation above, the first thing the user input is a direct search for sunglasses. In this case, the bot will follow the flowchart specified before and provide store recommendations as an output. As a follow-up question, the bot always asks if the user is looking for something else. As the response was yes, it opens an input field where the consumer can type what else they are looking for. In this case, we used a longer sentence format, saying “I am looking for T-shirts” (“yo busco por camisetas”, in Spanish). It breaks down this sentence into words, and searches for them in the database. In this specific example, T-shirts belong to the fashion category, so it asks for the user to specify if they are looking for menswear, womenswear or childrenswear. In our example, we simulated womenswear (“moda femenina”, in Spanish). The response is then a filtered categorization for the recommended stores.

After each recommendation the bot is set to always ask if the customer is looking for anything else, and in this example, we explored with a new category: motorcycles (“motocicletas”, in Spanish). There is only one store in the shopping centre that sells it, and it is the recommendation output. Finally, the user ends the session replying no to the question if they are looking for something else.

The conversational examples provided so far correspond to controlled situations where the user always types exactly what the bot is expected to respond, which is not necessarily how

users would behave in real life scenarios. So, when developing the bot, some conditions were created to deal with unexpected inputs. Examples are shown in Figure 3.4.

```
Bot: Hello! What product are you looking for today? Please search for one product at a time.
You: no me gustas
Bot: I could not find the product you are looking for. Can you try to rephrase it for me? If you want to quit, just text bye.
You: bla bla bla
Bot: I could not find the product you are looking for. Can you try to rephrase it for me? If you want to quit, just text bye.
You: pantalones
Bot: Are you looking for moda masculina, moda femenina o infantil? You: masculina
Bot: Here are some stores where you can find it:
Fifty Outlet
C&A
Moda-re
Lefties
Superdry Outlet
Bot: Are you looking for any other product today? (yes/no) no
Bot: Ok, hope I was able to help. Bye!
```

Figure 3.4: Example of chatbot conversation with unexpected inputs

In the conversational example above, after the greeting the user input a phrase saying ‘I do not like you’, in Spanish, “no me gustas”. Evidently, there is no correspondence for these words in the database, so the bot returns saying that it could not find the product the user is looking for and asking for a rephrase. It also suggests a way to quit the conversation, by texting bye. If the user keeps trying with non-matched words, this is the message the bot will keep providing. And when the user inputs correspondence to the database, as in this example “pantalones” (pants, in Spanish) it moves forward with the store recommendation as shown previously.

However, we believe that most users will use the tool as a way of finding a specific product they are looking for. To simulate this situation, we have Figure 3.5 showcasing what could be a common use of the chatbot.

```
Bot: Hello! What product are you looking for today? Please search for one product at a time.
You: almohadas
Bot: Here are some stores where you can find it:
['IKEA' 'JYSK' 'Maxcolchon' 'Que Sofás']
Bot: Are you looking for any other product today? (yes/no) no
Bot: Ok, hope I was able to help. Bye!
```

Figure 3.5: Example of expected chatbot conversation

In this case, it provides a direct response for a search of a specific product, in this case ‘almohadas’, which translates to pillows in English. As this is a product from the home and decor category, no follow-up questions are asked, and the output is an effective store

recommendation. When the customer replies that they are not looking for anything else in this visit, the bot quickly ends the conversation.

4. Expert Feedback Evaluation

For the evaluation of the chatbot, and its potential implications if applied to the shopping centre industry, we have interviewed seven different Ingka Centres representatives to assess their point of view on the tool. These are experts in the shopping centre area and/or in the technological infrastructure of Ingka Centres. To conduct the interview, eleven questions were asked to rank the chatbot according to the PARADISE method explored in Chapter 2. The questions asked were the following:

1. On a scale from 1 to 5, how do you rate the ease of usage of the chatbot? With 5 being very easy, and 1 being very hard.
2. On a scale from 1 to 5, how do you rate the clarity of the responses of the chatbot? With 5 being very clear and 1 being not clear at all.
3. On a scale from 1 to 5, how natural do you consider the response of the chatbot to be, compared to a human response? With 5 being natural, compared to a human response, and 1 being very robotic, non-natural.
4. On a scale from 1 to 5, how friendly do you believe this chatbot is? With 5 being very friendly and 1 being not friendly at all.
5. On a scale from 1 to 5, how do you rate the ability to navigate through non expected inputs? With 5 being navigating very well, and 1 being navigating poorly.
6. On a scale from 1 to 5, how willing would you be to use the system again? With 5 being very willing and 1 being not willing at all.
7. Do you believe that the bot is providing accurate responses, based on the products being searched?
8. What do you think is working well in this chatbot?
9. Are there any capabilities that you would like the bot to have, but it currently has not? (limitations)
10. Do you see value in the usage of this chatbot in a shopping centre facility? In what context?
11. In your opinion, what would be the expected impact of the chatbot in the shopping centre? What can this chatbot do for the tenants?

These questions were established with the intent to cover the six aspects present in the article of Adamopoulou & Moussiades (2020) when they review different methodologies for

chatbot evaluation. Six questions were established with a rank setup, where 5 would represent being very good in the aspect being evaluated, and 1 being very bad. This approach was adopted to capture the general perception of the strengths and opportunities of the chatbot and compare the different criteria considering an objective form. After the interviewee provided a rate, the interviewer asked the reason behind it as a form to capture potential drivers of the good, neutral, or bad result in a specific item.

The open-ended questions, from seven to eleven, have this format in order to better explore the insights provided before. They review the identified strengths (what is working well), opportunities (existing limitations), and the general perception of how the adoption of a system could benefit the shopping centre industry and the tenants on a shopping centre facility. Even though customers' perception was not phrased as part of the questions, several experts mentioned impacts for customers as well, so these are also present in the results.

Additionally, question seven was created with the intent of covering an aspect discussed in Adiwardana *et al.* (2020), that for the further adoption of chatbots, they need to make sense. So, the experts were also evaluating the accuracy of the responses being provided by the bot, to the best of their knowledge.

4.1. Individual results from the interviews

In the *first interview*, the lowest ranked aspect of the chatbot was the ability to navigate through unexpected inputs, which was ranked 3 out of 5. Reason for it is that even though the response was clear, the bot provides the same text over and over. Asking for the user to rephrase it because it could not find the specific product is polite and friendly, but loses the opportunity of redirecting the response with some examples or suggestion of usage.

The other aspects were ranked with either 4 or 5 scores. The main strength mentioned for these ratings, and in the question where we ask directly about what works well on the chatbot, was its ease of use, requiring little to no knowledge on the customer side. The respondent also mentioned that, in this specific shopping centre, there is a frequent change in some of the tenants (stores available). Therefore, adopting a recommendation technology as the proposed one could be even more useful in this scenario, making the brands more visible for potential customers. Regarding the friendliness of the bot, a very important aspect to be taken into consideration is the Ikea hospitality culture. This is a key piece for all the companies

under the Ikea umbrella, and thus there is significant importance of measuring it with all the respondents.

Regarding the limitations of the system, some improvements mentioned were a ranking for the recommendation of the stores when the product is a match. With the proposed development of this thesis, the bot randomly selects 5 stores after a product is found on the database, if found in more than 5. The proposal brought on the interview was to manually align a recommendation order for stores, by product category. To effectively redirect customers to the stores where they are more likely to find a particular product. Furthermore, another item mentioned was the possibility of using the output of the chatbot in the context of upselling and cross-selling. After recommending stores, the bot could provide a map to the store and other products related to the one being searched that could interest the customer.

Still, in their opinion, this implementation would add value to the shopping centre, especially in its website, as it can serve as a welcome tool for the customers, redirecting them to places where they can solve their needs. This could eventually bring more awareness to all brands in the facility, increasing visitation and cross-visitation to the stores. Lastly, there could be an increase in revenue for the tenants as well.

The *second interviewee* had the lowest rate as a 4 out of 5, and the lowest ranked parameters were the navigation on unexpected inputs, the naturalness of responses of the bot, and its friendliness. They mentioned that its lack of proposing a variety of different text responses, especially in the unexpected input situation, make it robotic. The good aspects responsible for the five-star rating in the other questions are the clarity of the chatbot, and being a technology that can be easily adopted by the users. The store recommendations were perceived as correct, and one of the main aspects that was working well in the bot is its convenient use for customers that want to explore locations where they can find a specific product they are looking for.

Regarding the limitations of this study, one that was mentioned is the need of the bot providing a store map, after the recommendations are done, based on the users' preferential store. They saw the value of the implementation of this technology, especially on the webpage, serving as a form to attract users to visit the shopping centre facility. Regarding its impacts, it can help customers to navigate the stores physically, improving the customer journey and eventually customer satisfaction.

In the *third interview*, the lowest score given was a 3, regarding the naturality of the response of the chatbot compared to an expected human response. The reason for this rate was the nature of the bot itself, being based on a set of rules, instead of more conversational or generational technologies. All other aspects were ranked as a 4 out of 5, the positive aspects that ensure this grading was the accuracy of the recommendations provided by the model, and the way it aligns with the conceptual idea behind this study. Regarding improvement points, the main one would be integrating with a store map, for more effective guidance of the customer within the shopping centre facility. Also, a suggestion for enhancing the user experience was a pop-up window with targeted promotions on the items being searched, in a partnership with the stores based on the facility.

In this interview, the respondent also saw value in the implementation of a system like this for LUZ Shopping, especially in the webpage, for customers to use when they are planning a visit to the shopping centre. The expected impact would be an increase in the number of visitors, as users would know exactly where to find beforehand certain products they are looking for. For the tenants, considering that it could include a pop-up with discounts, they could use it as another way to promote their internal campaigns, eventually leading to more sales. It also helps to simplify the decision process of the clients, making it easier to navigate a building with several different store options.

For the *fourth participant*, the aspects with the lowest ranks were the naturality of the responses, the ability to navigate unexpected inputs and the willingness to use the system again, all ranked at a 3. The main drivers for the low ratings in these categories were the formality in the form of communication, which is not very compatible with the human-like type of responses in most cases. Furthermore, Python being a case-sensitive language, and the bot being based on a set of rules, makes it weak against misspelling. Every time a product is misspelt, it will be classified as an unexpected input, which damages the customer experience with the tool itself.

Regarding other evaluated aspects, the bot was ranked with 4 or 5 ratings, mainly due to the simplicity of its usage, requiring no training or adjustment prior to the first contact with the tool, and the direct responses to customer inquiries. Another aspect that was mentioned as strengths of the current method was the follow-up question of specific categories for fashion products being searched. The respondent classifies this as very important to redirect the user to

the correct store. At the same time, they also mentioned that it is a good thing not to be asking follow-up questions on the other categories, creating a simpler flow for responses and a more straight-forward communication style. Additionally, the consideration of unexpected inputs in the set of rules used to develop the bot was perceived as a good thing, as there is a need to anticipate different ways it could be used.

Looking at the limitations of the current setup, having a store location map after the recommendation provided was the first item mentioned. Besides that, the ability to handle misspelling and making it more interactive for the users were also improvements mentioned, in their point of view. The interactive factor could be a recommendation of a brand based on a store-rank for specific products, something similar to what respondent one has suggested. They also considered the responses of the bot to be correct and see value in the implementation of this type of technology for the shopping centres, in the webpage but also physically, to attend all types of publics.

The expected impact of implementing the chatbot is an increase on visitations, as the tenants could use it as a way to promote their products, and smaller brands could be known by a public that they usually do not reach. It would also be a good exposure method for the shopping centre itself, if it had good results, attracting more tenants over time.

During the *fifth interview*, the respondent ranked the naturality of the chatbot responses as being the lowest, with 2 out of 5. Mainly since the bot is based on a set of predefined response rules, which makes it more robotic than generative models, for example. The other aspects were rated from 3 to 5 stars, and the biggest strengths recognized in the model are the simple use, and the accuracy of recommendations provided. Considering the current setup, the respondent evaluated the answers from the bot as being clear, direct, and easy to understand, even in cases of unexpected inputs.

Regarding limitations and potential improvements to be incorporated in the model, they suggested that it asks more questions to the user before providing a recommendation, to understand in more details what is that the customer is looking for, being more precise with the output. Also, ranking the stores by order of importance from some products could be a good addition, together with the reason why one particular store could be a top pick for that product. In their point of view, this could stimulate the visitation of the shopping centre even further, as the user would be more certain that their needs could be fulfilled.

Regarding the expected impacts of the adoption of the bot, they believe it could increase visitation and make the stores, especially smaller ones, more visible to customers, which would lead to more sales eventually. For the customers, this would align with a global tendency seen in retail of saving time when looking for products. Having the bot on the website would be a good way to ease the customer journey through the shopping facility.

In the *sixth interview*, the aspect ranked the lowest was again the naturality of the responses of the bot, with 3 out of 5. However, the respondent said it can be a positive thing for the responses to be more robotic, this way the users would be sure they are not talking to a human and adjust expectations accordingly. The other aspects were ranked mainly with 5 out of 5, and this aligns with the strengths they mentioned about the model: being direct, clear, and simple to use.

In this interview, one interesting perspective mentioned was that the bot could make the customer experience more accessible for the ones that do not want, or cannot for some reason, talk to the physical reception in the shopping centre. To support all types of customers, they recommended this solution to be implemented both on-site, and also on the web, as many visitors sometimes do not know where to find a physical support when they are visiting the shopping centre.

Regarding enhancements on the tool, it could recommend discounts or promotions in the products being searched. Further, a potential effect of applying this to the shopping centre industry would be more visibility for the stores, and accessibility for the customers who already know what they are looking for, but do not know where they can find it.

In the *seventh and final interview*, the facet ranked the lowest, with a 3 out of 5, was the naturality of the responses from the bot, as they believe that a human would handle the flow of the conversation differently. The other aspects were ranked at a 5, and mainly because the system is fast, easy, and simple to use, while providing accurate responses. Regarding the existing limitations, they would like the follow up question for fashion items to be separate from the flow of the search, starting with it from the beginning. Also, having a way to communicate discounts and promotions would be appreciated. Lastly, the expected impact in their opinion would be an increase in the visitations from the shopping centre.

4.2. Summary of Interviews

To compare the results provided from the objective questions (questions 1 to 6), Table 4.1 was formatted compiling the feedback received.

Aspect of Evaluation	Participant 1	Participant 2	Participant 3	Participant 4	Participant 5	Participant 6	Participant 7	Index Avg.	Index Median
Ease of usage	5	5	4	4	5	5	5	4,7	5,0
Clarity of responses	5	5	4	5	5	5	5	4,9	5,0
Naturality of responses	4	4	3	3	2	3	3	3,1	3,0
Friendliness	4	4	4	4	3	4	5	4,0	4,0
Navigation of unexpected inputs	3	4	4	3	4	5	5	4,0	4,0
Willing to use the system again	5	5	4	3	3	5	5	4,3	5,0

Table 4.1: Compilation of the responses provided by the participants of the interviews, for the objective questions.

Based on Table 4.1, we can see that the clarity of the responses, and the ease of usage from the chatbot are the two best ranked features, when it comes to the average but also to the median. This result also aligns with the qualitative feedback given on question eight, where many interviewees have referred to one of the strengths of the chatbot being its straight-forward communication style, and no need of training or understanding the technology before using it, as it is very simple and easy to handle.

The criterion with the lowest average ranking was the naturality of the responses, when compared to a human response. This also resonates with the qualitative feedback collected, that mentioned the rule-based nature of the chatbot as something unnatural when compared to the ways humans communicate. However, it is interesting to notice that this was not perceived as a bad thing for all participants, as some enjoyed that it differentiates from a human support, and others appreciate the way it guides the user to type inputs the bot can recognize.

All the respondents classified the suggestions coming from the bot as being correct, which is essential for the effective evaluation of the model. When we focus on the areas of improvements, the most mentioned items were the need of providing a location map for the chosen store the customer prefers, incorporating a ranking for recommending a store, based on experts' perception, and linking it to promotions and discounts to attract more customers to visit the shopping centre.

Additionally, when asked, the experts saw value in the implementation of this type of solution for the shopping centre industry, especially in the webpage as a way of inviting customers to visit the location. For the expected benefits of having it, the most commonly mentioned were an improvement of the customer journey in the facility, guiding them to visit stores that can successfully fulfil their needs. For the tenants, it was almost a consensus between all interviewees that it would increase their visibility and could eventually generate more revenue from it. One interesting aspect mentioned in only one of the interviews is the accessibility, considering that it could provide help for different types of customers, with different needs.

5. Conclusion

Even though the chatbot's field has been advancing quickly, there's still no conclusion to Alan Turing's question "can machines think?". It is, however, noticeable that they have much more applications now than they did in the past. Chatbots are being adopted with different objectives across several industries, and they can support human agents in some tasks, such as customer service.

Furthermore, regarding the methods used to create the model, even though web-scraping is an old method to extract information, it is still useful for accessing, storing, and analysing information from the internet. Natural Language Processing is a key step when dealing with human-like conversations, as there's no standard format in the inputs being provided, and it can have ambiguous meanings as well. If NLP is applied to pre-process the text, it makes it easier for the systems to access and understand it, before providing a recommendation. Finally, the bot was developed based on a set of rules, chosen by the writer and Inga Centres experts. This makes it have robotic communication, in the way that will always answer in the same way for unknown questions. However, it also ensures clear and straightforward communication, especially for the cases where it guides the customer to a particular store.

In this paper we used the combination of techniques mentioned above to develop a chatbot that can slightly transform the customer journey in the shopping centre experience. With web-scraping applied to build the database, natural language processing to make sure the data is machine-readable, and a rule based informational bot. The effectiveness and potential application of this model was measured by expert's interviews. All the participants, workers of the shopping centre industry, saw value in adopting this solution, especially on the webpage, to help guide the customer when looking for a particular item. Other benefits are the visibility of smaller tenants in a shopping centre, that could use the bot as a way of promoting the brand, and more visitation to the shopping facility.

The features that were best evaluated from the current setup are the easiness of usage and the clear and direct communication. These findings go along with the ones that Collier & Kimes (2013) described in their article, mentioning that users tend to appreciate the adoption of self-help technologies if they are perceived as convenient. The bot is also providing accurate

responses based on all experts' feedback, which is crucial for further adoption, as one important aspect that needs to be considered is the recommendation of the bot adding value to the customer journey.

5.1. Limitations

Even though the current model was, in general, well evaluated by all participants of the interviews, there were several improvement aspects that need to be considered. The first one, based on objective factors, is the way the bot is communicating. Although it is clear, it is unnatural and very robotic. A potential improvement on this factor could be the leverage of conversational models such as Open AI ChatGPT into the solution, through APIs. A simpler approach that could be adopted is, also, the pre-set of different responses per each interaction, making the bot vary between them when in a conversation.

There are several other facets that could be enhanced when reviewing the open-ended questions' responses. Including a store map at the final output would better support the customer on finding exactly where they need to go, and that can be interpreted as a mix between the common existing self-help solutions in shopping centres with this new approach. Another aspect that was commonly mentioned is the need of creating a ranking for the recommended stores based on the products being searched, explaining to the customer why a particular store could be the best fit for their needs. Finally, the third most mentioned item is to use this type of tool as a way of enhancing visibility for the tenants, including discounts and promotions of the searched items in the same screen. It could lead to even more sales and more visibility to the stores.

5.2. Future Research

This study serves as a starting point to re-think the traditional approach of self-help in the shopping centre industry. A further extension to this research could go in two different ways. First would be changing the development of the solution for a generative AI approach, creating more natural and complex responses for a more human-like interaction with customers. A different and more modern technique for creating the product database could also be explored, as in this study we adopted web-scraping, which is a very effective but old method of extracting information from the web.

A second approach for further research could be evaluating this type of solution with the customers directly, by making it available in a shopping centre website. This would be effective to understand what are the key items being searched by customers, what is the adoption range (number of uses per day or per week), and objective feedback from the users with a short survey in the end.

References

- Abodayeh, A., Hejazi, R., Najjar, W., Shihadeh, L. & Latif, R. (2023). Web Scraping for Data Analytics: A BeautifulSoup Implementation, in *2023 Sixth International Conference of Women in Data Science at Prince Sultan University (WiDS PSU)*, 2023 Sixth International Conference of Women in Data Science at Prince Sultan University (WiDS PSU), Riyadh, Saudi Arabia, March 2023, Riyadh, Saudi Arabia: IEEE, pp.65–69, Available Online: <https://ieeexplore.ieee.org/document/10145369/> [Accessed 22 April 2024]
- Adamopoulou, E. & Moussiades, L. (2020). Chatbots: History, Technology, and Applications, *Machine Learning with Applications*, vol. 2, p.100006
- Ayuntamiento de Jerez. (n.d.). Información Util, Available Online: <https://www.jerez.es/ciudad>
- Bird, S., Loper, E. & Klein, E. (2009). Natural Language Processing with Python, [e-book] O'Reilly Media Inc, Available Online: <https://www.nltk.org/book/>
- Cahn, J. (2017). CHATBOT: Architecture, Design, & Development
- Chu, M.-N. (2023). Assessing the Benefits of ChatGPT for Business: An Empirical Study on Organizational Performance, *IEEE Access*, vol. 11, pp.76427–76436
- Chung, M., Ko, E., Joung, H. & Kim, S. J. (2020). Chatbot E-Service and Customer Satisfaction Regarding Luxury Brands, *Journal of Business Research*, vol. 117, pp.587–595
- Collier, J. E. & Kimes, S. E. (2013). Only If It Is Convenient: Understanding How Convenience Influences Self-Service Technology Evaluation, *Journal of Service Research*, vol. 16, no. 1, pp.39–51
- Glez-Peña, D., Lourenço, A., López-Fernández, H., Reboiro-Jato, M. & Fdez-Riverola, F. (2014). Web Scraping Technologies in an API World, *Briefings in Bioinformatics*, vol. 15, no. 5, pp.788–797
- Henrys, K. (2021). Importance of Web Scraping in E-Commerce and E-Marketing, *SSRN Electronic Journal*, [e-journal], Available Online: <https://www.ssrn.com/abstract=3769593> [Accessed 22 April 2024]
- Hirschberg, J. & Manning, C. D. (2019). Advances in Natural Language Processing, *ARTIFICIAL INTELLIGENCE*
- Ingka Centres. (n.d.). LUZ Shopping, Available Online: <https://www.ingkacentres.com/en/where-we-are/spain/luz-shopping/>
- Limna, P., Kraiwanit, T., Jangjarat, K., Klayklung, P. & Chocksathaporn, P. (2023). The Use of ChatGPT in the Digital Era: Perspectives on Chatbot Implementation, *Journal of Applied*

Learning & Teaching, [e-journal] vol. 6, no. 1, Available Online: <https://journals.sfu.ca/jalt/index.php/jalt/article/view/797> [Accessed 22 April 2024]

LUZ Shopping. (n.d.). Tiendas y Restauracion, Available Online: <https://www.luzshopping.com/es/>

Nadkarni, P. M., Ohno-Machado, L. & Chapman, W. W. (2011). Natural Language Processing: An Introduction, *Journal of the American Medical Informatics Association*, vol. 18, no. 5, pp.544–551

Open AI. (2022). Introducing ChatGPT, Available Online: <https://openai.com/blog/chatgpt> [Accessed 22 April 2024]

Oxford English Dictionary. (2023). Chatbot, n., Available Through: Oxford English Dictionary <https://doi.org/10.1093/OED/2981785869>

Peters, F. (2018). Master Thesis : Design and Implementation of a Chatbot in the Context of Customer Support

Richardson, L. (2014). Beautiful Soup Documentation

Shehan Franciscu. (2023). ChatGPT: A Natural Language Generation Model for Chatbots, [e-journal], Available Online: <https://rgdoi.net/10.13140/RG.2.2.24777.83044> [Accessed 22 April 2024]

Software Freedom Conservancy. (2023). Selenium Overview, Available Online: <https://www.selenium.dev/documentation/> [Accessed 30 April 2024]

Tran, A. D., Pallant, J. I. & Johnson, L. W. (2021). Exploring the Impact of Chatbots on Consumer Sentiment and Expectations in Retail, *Journal of Retailing and Consumer Services*, vol. 63, p.102718

Turing, A. M. (1950). I.—COMPUTING MACHINERY AND INTELLIGENCE, *Mind*, vol. LIX, no. 236, pp.433–460

Wikipedia. (2024). Jerez de La Frontera, Available Online: https://en.wikipedia.org/wiki/Jerez_de_la_Frontera

Zheng, S., Yahya, Z., Wang, L., Zhang, R. & Hoshyar, A. N. (2023). Multiheaded Deep Learning Chatbot for Increasing Production and Marketing, *Information Processing & Management*, vol. 60, no. 5, p.103446

Appendix A

Web-scraping code for stores and categories

```
def scrape_store_categories_and_names():
    url = "https://www.luzshopping.com/es/tiendas/"
    categories_and_names = []

    os.environ["PATH"] += os.pathsep + 'C:/Users/ana_c/Downloads'

    driver = webdriver.Edge()
    driver.get(url)

    try:
        click_count = 0
        while click_count < 3:
            show_more_button = WebDriverWait(driver,
10).until(EC.element_to_be_clickable((By.XPATH,
"//button[contains(text(),'Mostrar más')]")))

            driver.execute_script("arguments[0].scrollIntoView();",
show_more_button)

            show_more_button.click()

            time.sleep(5)
            click_count += 1

    except ElementClickInterceptedException:
        print("ElementClickInterceptedException occurred. Trying again...")

        show_more_button.click()
        time.sleep(5)
        click_count += 1

    soup = BeautifulSoup(driver.page_source, 'html.parser')
    store_containers = soup.find_all('div', {'class': 'flex min-h-40 flex-1
flex-col flex-wrap rounded-b-xl bg-background-primary-light py-5 px-6
transition-all duration-200 group-hover:bg-background-secondary'})
    for container in store_containers:
        store_name = container.find('div', {'class': 'text-sm mt-1 font-title
font-bold'}).text.strip()
        categories = container.find('div', {'class': 'text-
bodys'}).text.strip().split(', ')
        for category in categories:
            categories_and_names.append([category, store_name])

    driver.quit()
    return categories_and_names
```

Appendix B

Example of web-scraping code for product details

```
def scrape_chiarajoyitas_product_categories():
    url = "https://chiarajoyitas.es/"
    headers = {'User-Agent': 'Mozilla/5.0 (Windows NT 10.0; Win64; x64)
AppleWebKit/537.36 (KHTML, like Gecko) Chrome/58.0.3029.110 Safari/537.3'}
    response = requests.get(url, headers=headers)

    if response.status_code == 200:
        soup = BeautifulSoup(response.text, 'html.parser')

        # Find the HTML elements containing product categories
        category_containers_1 = soup.find_all('a', {'class': 'cbp-column-title
nav-link cbp-category-title'})
        category_containers_2 = soup.find_all('span', {'class': 'cbp-tab-
title'})

        # Extract text content from each category container
        categories_1 = [container.text.strip() for container in
category_containers_1]
        categories_2 = [container.text.strip() for container in
category_containers_2]

        # Combine both sets of categories into a single list
        all_categories = categories_1 + categories_2

        # Exclude duplicates and undesired categories
        undesired_categories = {'Horóscopo', 'Novedades', 'Eclipse',
                                'Outlet'}
        unique_categories = [cat for cat in set(all_categories) if cat not in
undesired_categories]

        split_categories = []
        for category in unique_categories:
            if ' y ' in category:
                split_categories.extend(category.split(' y '))
            else:
                split_categories.append(category)

        return split_categories

    else:
        print(f"Failed to fetch webpage: {url}")
```

Appendix C

Code for the bot development

1. Function to identify in which store you can find a particular product

```
def find_stores_2(query, top_n=5, category=None):
    query_tokens = preprocess_text(query)
    matched_stores = df[df['Product Details'].apply(lambda x: any(token in x
for token in query_tokens))]['Store Name'].unique()
    if category is not None:
        matched_stores = df[(df['Store Name'].isin(matched_stores)) &
(df['Category'].apply(lambda x: category in x))]['Store Name'].unique()

    if len(matched_stores) <= top_n:
        return matched_stores
    else:
        return random.sample(matched_stores.tolist(), top_n)
```

2. Function to validate if a particular product belongs to the fashion category, to ask the follow-up question of what is the type of fashion the user is looking for

```
def check_if_fashion(user_input):

    for word in user_input:
        if word in fashion_products.values:
            response_1 = input("Bot: Are you looking for moda masculina, moda
femenina o infantil? You: ").strip().lower()
            clothes_find = find_stores_2(word, category=response_1)
            if clothes_find:
                print("Bot: Here are some stores where you can find it:")
                for store in clothes_find:
                    print(store)

        else:
            product_find = find_stores_2(word)
            if any(product_find):
                print("Bot: Here are some stores where you can find it:")
                print(product_find)
            elif product_find:
                print("Bot: Here are some stores where you can find it:")
                print(product_find)
```

3. Validate how many products are being searched in the user input

```
def check_if_more_than_one_product(word_to_search):
    number_products = 0
    for word in word_to_search:
        word_found = any(re.search(r'\b{}\b'.format(re.escape(word)), details)
    for details in df['Product Details'])
        if word_found == True:
            number_products += 1
    return(number_products)
```

4. Function to validate, if more than one product is being searched, if they are in the same cell (e.g. 'gafas de sol')

```
def check_multiple_products_in_same_cell(user_input):
    products_found = set()
    for index, details in enumerate(df['Product Details']):
        if all(w in details for w in user_input):
            products_found.add(df.loc[index, 'Store Name'])

    products_found = list(products_found)
    if len(products_found) > 5:
        products_found = random.sample(products_found, 5)

    return products_found
```

5. Chatbot response rule code

```
user_input_storage = []

def chatbot_luz_shopping_v4():
    print("Bot: Hello! What product are you looking for today? Please search for one product at a time.")

    while True:
        product_not_found = True
        user_input = input("You: ").strip().lower()
        user_input_storage.append(user_input)
        split_message = re.split(r'\s+|[,;?!.-]\s*', user_input.lower())
        filtered_split_message = [word for word in split_message if word not in stop_words]

        for word in filtered_split_message:
            if word in ['exit', 'bye', 'quit']:
                print("Bot: Goodbye! Hope to see you soon.")
                return
            elif word in ['hello', 'hola', 'hi', 'heyhey']:
                print("Bot: Hello! What product are you looking for today?")
                continue
```



```

        else:
            product_count =
check_if_more_than_one_product(filtered_split_message)

            if product_count >=1:
                if product_count>1:
                    stores =
check_multiple_products_in_same_cell(filtered_split_message)
                    print("Bot: Here are some stores where you can find
it:")

                    for store in stores:
                        print(store)

                elif product_count==1:
                    check_if_fashion(filtered_split_message)

                    response = input("Bot: Are you looking for any other product
today? (yes/no) ").strip().lower()
                    if response in ['yes', 'fine', 'ok', 'sure']:
                        break
                    elif response in ['no', 'nope', 'nevermind']:
                        print("Bot: Ok, hope I was able to help. Bye!")
                        return
                    elif response in ['exit', 'bye', 'quit']:
                        print("Bot: Goodbye! Hope to see you soon.")
                        return

            else:
                print("Bot: I do not think I know this one, please
answer with yes or no")
                continue

        else:
            if product_not_found:
                print("Bot: I could not find the product you are looking
for. Can you try to rephrase it for me? If you want to quit, just text bye.")
                product_not_found = False
            continue

```