MEDBOT: CHATTING FOR HEALTHCARE SERVICES

Pedro Manuel Simões de Sousa Ferreira
pedro.m.s.ferreira@tecnico.ulisboa.pt

Instituto Superior Técnico, Lisboa, Portugal
October 2019

Keywords: Use Cases, Chatbot, Conversational User Interface, Natural Language Understanding, Dialog Systems.

Abstract: The heavy usage of chat platforms by users, allied with developments in natural language understanding, offer a favourable scenario for organizations to implement use cases in chatbots. This research identifies factors that influence the suitability of use cases for conversational user interfaces, enabling organizations to make more informed decisions regarding chatbots implementations. The factors identified are grouped in three categories: (i) general factors; (ii) factors to be considered to implement a chatbot over a human operator; and (iii) factors that should be considered when implementing a chatbot over a traditional GUI application. A use case selection process is created that uses the factors gathered and enables to prioritize the more appropriate use cases between a set of potential use cases for chatbot implementation. The selection process is applied to several use cases in the health care domain, yielding the use case of scheduling a medical appointment with highest priority. A reference architecture of a chatbot for scheduling an appointment is defined and implemented. User tests are conducted, comparing the interactions with the chatbot with a traditional graphical user interface (website). User tests indicate the chatbot as more efficient than the website, and most testers indicate it as a preferable method for scheduling an appointment, when comparing to the website.

1. INTRODUCTION

The developments in the artificial intelligence field have been responsible to the rise of new, more intelligent systems. Specifically, the developments in the Natural Language Processing, drive the development of chatbots. Chatbots are systems that interact with the user using natural language, as if the user were talking with another human. Today people are using chat platforms as one of the main channel of communication, using applications such as Facebook Messenger or WhatsApp. The heavy usage of chat platforms allied with the developments in natural language process (NLP) create a favourable scenario to organizations to offer their services using conversational user interfaces. Services can be accessed directly from the chat platforms the users already use, in a more natural way, instead of requiring the users to install a specific app or access the organization website.

Health Care is one field that can benefit from the implementation of chatbots, both in customer service, enabling, for instance, users to schedule appointments or even in the clinical field, by assisting the decision process of healthcare professionals.

Although chatbots can offer the beforementioned advantages, it is important to assess if a use case is suitable for a conversational interface. Moreover, organizations should be able to select the more appropriate use cases between several use case candidates, in order to understand the more suitable ones and prioritize them.

This work is realized in the context of Medclick. Medclick is a company that will provide a one-stop platform to book a medical appointment in a fast and user-friendly way, across multiple medical service providers.

1.1 Research Problem

This research aims to understand what use cases are adequate to expose via a conversational user interface. This raises the following questions:

- How can suitable use cases be identified in a systematic way?
- What use case factors influence its suitability to be implemented in a dialog agent?
- How can organizations prioritize use cases for a chatbot implementation?

It is therefore a goal of this research to find factors that impact the suitability of a use case for a chatbot, and use such factors to create an use case evaluation process, that can be used by any organization to evaluate the suitability of its use cases for a conversational user interfaces, and prioritize the more suitable ones.

Furthermore, this research aims to create a reference architecture of a chatbot in the healthcare domain. The use
**case evaluation process** is used as the starting point to answer the following question:

- What healthcare use cases are more appropriate to be exposed via a chatbot?

After answering this question, it is possible to select a suitable healthcare use case and create a reference chatbot architecture.

### 2. BACKGROUND AND RELATED WORK

#### 2.1 Dialog Systems

**Conversational agents or dialog systems** are programs that communicate with users in natural language. This kind of systems can be classified in two categories [1]:

- **Task-oriented dialog agents:** are designed for a particular task and set up to have short conversations to get information from the user to help complete the task. These include the digital assistants that can give travel directions, control home appliances, find restaurants, or help make phone calls or send texts.

- **Chatbots:** Chatbots are systems designed for extended conversations, set up to mimic the unstructured conversational characteristic of human-human interaction, rather than focused on a particular task. These systems often have an entertainment value. Chatbots are also often attempts to pass the Turing test [2]. Chatbots can also have some practical uses such as testing theories of psychological counselling.

The word “chatbot” is often used in the media and in industry as a synonym for conversational agent [1]. In this document the term “chatbot” is used in that same more general sense. In reality, the kind of systems explored in this thesis are typically task-oriented dialog agents, even though we may refer them using the word “chatbot” instead of “task-oriented dialog agent”.

It is also important to notice that even though dialog systems communicate with users in natural language, other form of GUI elements are often used such as pre-defined quick replies that the user can click in order to make the interaction faster and easier.

#### 2.2 Natural Language Processing

Natural language processing (NLP) is a subfield of computer science concerned with using computational techniques to learn, understand, and produce human language content [3]. Some applications of NLP include: **information extraction**, transforming unstructured data found in texts into structured data [1]; **conversational agents**, that aid human-machine communication [3]; or **machine translation**, the use of computers to automate the process of translating from one language to another, aiding human-human communication [1] [3].

The factors that have allowed the development of NLP in the last years twenty years, according to [3], are: (i) increase in computing power, (ii) the availability of large amounts of linguistic data, (iii) the development of successful machine learning methods, and (iv) a richer understanding of the structure of human language and its deployment in social context.

#### 2.2.1 Natural Language Understanding in Dialog Systems

There are various possible structures to represent the meaning of linguistic expressions. Modern task-based dialog systems are based on a domain ontology, a knowledge structure representing the kinds of intentions the system can extract from user sentences [4]. The ontology defines a frame-based representation, with one or more frames, each a collection of slots, and defines the values that each slot can take.

Dialog agents typically have a natural language understanding module. NLU is responsible for the semantic parsing of user utterance, i.e., it gives semantic meaning to user utterances. This module is responsible for selecting the appropriate frames and filling the slots of the before mentioned domain ontology structure. This module objective is therefore to extract three things from the user’s utterance [4]:

- **domain classification:** if the systems is not single-domain, there is the need to determine what domain is the user referring to.

- **intent determination:** what general task or goal is the user trying to accomplish. For example, the task could be to Find a Movie, or Show a Flight, or Remove a Calendar Appointment.

- **slot filling:** extract the particular slots and fillers that the user intends the system to understand from their utterance with respect to their intent.

Consider the sentence “Book me a table for two for Friday night at Sushi Place”. The NLU module would recognize the domain as “restaurant”; the intent as “book table” and would fill the time slots with “night” and “Friday”; the restaurant name slot as “Sushi Place”; and finally the slot for the number of seats as “two”.

The domain and intent determination are usually treated as a semantic utterance classification (SUC) problem and the slot filling as a sequence labelling problem [5].

Possible methods used by for domain/intent recognition and slot filling include: (i) hand-written rules; (ii) semantic grammars, that are context-free semantic grammar in which the left-hand side of each rule corresponds to the slot names; and (iii) supervised machine learning, using a training set that associates each sentence with the semantics, we can train a classifier to map from sentences to intents and domains, and for slot filling a sequence model can be used [4].
2.3 Chatbots, Humans and GUI Applications

This section explores what are the chatbots benefits, when replacing human interaction and traditional applications (GUI). It is also explored the intrinsic differences of the interactions.

**Human-Human vs. Human-Chatbot Interaction.** There are some differences in the way that people interact with a bot compared to a human. A study [6] concluded that people communicate with the chatbot for longer durations, using shorter messages, than they did with another human. Additionally, human–chatbot communication used simpler vocabulary than what is found in conversations among people and exhibited greater profanity. Factors such as number of words per conversation, shorthand terms, and emoticons were found to have no statistically significant differences.

The usage of chatbots in some scenarios bring advantages over humans, namely [7]:

- **Consistency:** chatbots can be consistent in services, which is important in certain sectors and may be hard to achieve with human operators.
- **Scalability:** chatbots can easily scale up to handle periods of unregular increased traffic, which is much harder with human operators.

With good design and implementation, Accenture [8] reports more than 80% of chat sessions resolved by a chatbot, that would otherwise been a human in a chat session or call.

**Human- GUI vs. Human-Chatbot interaction.** A report [9] by Forrest identifies the following factors that foster chatbots adoptability over traditional applications:

- **Chatbots promise a more convenient and natural user interface:** Typically, users must go to the process of discover, download, and install apps. Then, apps provide touch graphical interfaces to help consumers perform tasks. The experience isn’t natural, but it is effective. Conversations offer are more natural experience.

- **Mobile moment ownership is plateauing for enterprises:** Mobile is the first screen for consumers; however, consumers use only 25 to 30 apps on average each month and spend 88% of their time in just five downloaded apps.

- **Heavy use of instant messaging platforms:** Consumers spend 78% of their time on smartphones within apps. The median usage of instant messaging apps is 21.47 minutes per day among users of those apps and the pace of adoption is accelerating.

The fact that we are living an app fatigue moment, allied with the heavy usage of messaging apps, present an opportunity to replace traditional applications with chatbots available on the messaging applications that users are already using.

2.4 Existing Chatbot Services

As seen in section 2.2, it is possible to create NLU from scratch by either implementing rules or applying machine learning algorithms. Training machine learning models requires having access to rare expertise, large datasets, and complex tools, which presents a barrier to smaller companies [10]. The availability of NLP services in the cloud has powered the widespread use of chatbots. From the rise of open source tools to the arrival of cloud APIs, NLU capabilities once limited to the academic and research community are now accessible to a wider audience across industries [10].

In this section it compared three solutions offered by three major tech companies: (i) LUIS (Microsoft), (ii) Watson Assistant (IBM) and (iii) DialogFlow (Google).

A study [11] compared the performance of several NLU services, with the goal to enable both researchers and companies to make more educated decisions about which service to use. The results of the study, regarding intent and entity recognition for the platforms analysed in this research, can be seen in Figure 1.

These three solutions are further analysed concerning other features not present in [11]. In summary, Dialogflow offers more chat channels integrations and is the solution that supports more idioms. Both Dialogflow and Watson assistant offer graphical interfaces to create the chatbot, not requiring programming knowledge. This can accelerate development time, while Microsoft Bot Framework requires programming knowledge that makes it more extensible. Concerning the natural language understanding performance, i.e. intent and entity recognition, Microsoft’s solution has substantially better results with LUIS than both other solutions (see Figure 1). Finally, Microsoft Bot Framework is the only solution that offers support for authentication with external services, by providing support for OAuth 2 protocol.

![Figure 1 - F-scores for the different NLU services, grouped by corpus – source [11]](image)

3. CHATBOT USE CASE EVALUATOR

This section identifies use case factors that should be considered when assessing the suitability of a use case for a chatbot implementation. A use case selection process is
then proposed. This process allows the evaluation and ordering of a set of use cases, enabling the identification of the more suitable ones for chatbot implementation. After the definition of the use case selection process, it is applied to a set of use cases in the healthcare domain.

### 3.1 Use Case Evaluation Factors

In order to enable the evaluation of use cases, several factors are considered, reflecting the characteristics a use case should have in order to be appropriate to be implemented in a chatbot.

Chatbots lie between human operators and traditional graphical user interfaces (GUI) applications. In one hand, they can be used in the place of a human, offering a similar way of interaction, by using natural language. On the other hand, they can also be used instead of a traditional application, replacing a traditional graphical user interface with natural language.

The fact that chatbots share characteristics of both humans and GUI applications, fosters the division in the factors in three major groups:

1. **General Factors**: general factors that are essential to be considered to assure the viability of the use case to be implemented in a conversational UI.
2. **Factors over GUI application**: this group of factors reflect characteristics of a use case that can indicate that a chatbot is more adequate to expose it, instead of a traditional GUI application.
3. **Factors over Human**: factors that reflect characteristics of a use case that can indicate that the use case would benefit from being implemented by a chatbot instead of a human operator.

The analysis of such factors for each category yielded the following result.

#### General Factors:

- **Business Rules well defined**: Chatbots perform better solving specific requests were the process to solve it is standard [12]. This facilitates the creation of the flow of the conversation based on that business rules.
- **Integration with Existing Systems**: concerns if it’s possible to integrate the bot with the organization systems, via existing APIs. This factor guarantees that the chatbot can access the business logic and data required to the use case in question.

#### Factors Over GUI Application:

- **Multiple steps or multiple input parameters** [8]: A simple traditional UI might be more practical to use cases that are simple and require only one step, but for tasks that require several user data, using NLU we can sometimes get all the information that the user would input in a form, for instance, in a simple sentence. Consider the sentence “Can you rebook my flight to Madrid to the following Monday after 3pm and get me a window seat”. A traditional GUI would require the user to insert the different pieces of information in different steps of the process, while a chatbot would recognize all the information parameters directly from the user sentence.
- **Notifications required**: Messaging applications already include an efficient and functional push notification system, which is available by default without any additional implementation effort [13].
- **Authentication required** [13]: Usually, for each new application, users must create a new account to be uniquely identified. With bots, user authentication is not necessarily needed. The messaging platform used already provides a reliable identification of the user. Users are uniquely identified by default. This reduces the effort asked to the user to start using the service, not requiring to create an additional account.

#### Factors over Humans

- **High Volume, Simple tasks, performed by humans**: For simple, well defined, repetitive tasks, a chatbot can be more suitable than a human, in the sense that is more economical and frees the human resources for another tasks [8].
- **Consistency required**: For use cases that is important consistency in the performance, i.e., the use case must be performed the same way in every occurrence, chatbots can be more suitable than a human operator [7]. Bots are intrinsically more consistent than human operators.
- **Scalability required** [7]: some use cases have unstable loads of requests from users. Bots can scale-up to fulfill the requests. Using human operators, is hard to handle sudden increases of requests.

### 3.2 Use Case Selection Process

This section describes how the use case factors, defined in the previous section, can be used in order to select use cases suitable to implement in a chatbot. In order to accomplish the selection of use cases, a use case selection process is defined, consisting of three major steps. (see Figure 2).

![Figure 2 - Use Case Selection Process: overview](image)

The following use case selection process should be applied in order to identify and prioritize use cases to be implemented in a chatbot:

1. **Assessment for each use case**:
   1.1. **Assess if the UC is compliant with all the General factors**
   a. If the use case is not compliant with any of the factors, it should not be implemented in a chatbot altogether and the process stops. The general factors function as minimum
requirements to a UC to be considered for a chatbot implementation;

b. If the use case is compliant with all general factors, continue to step 1.2.

1.2. Assess Factors over GUI application

a. If some UC factors are compliant, a chatbot implementation is considered advantageous over a GUI application;

b. If none of the factors are compliant, a chatbot is not considered as more suitable than a GUI application.

1.3. Assess Factors over Human operator

a. If some UC factors are compliant, a chatbot implementation is considered advantageous over a Human operator;

b. If none of the factors are compliant, a chatbot is not considered as more suitable than a Human operator.

2. Filtering use cases: the use cases considered for chatbot implementation are only the ones that fulfil the following requirements:

• Meet the general factors (step 1.1.b);
• Advantageous to implement over both a GUI application (step 1.2.a);
• Advantageous to implement over a Human operator (step 1.3.a).

3. After applying this evaluation process, and filtering candidate use cases using the before mentioned requirements, there might still be several use cases left as candidates to implement in a chatbot. The method purposed to prioritize the use cases is to prioritize the ones that meet a greater number of factors, or that meet factors that are of greater importance to the organization implementing the use case. Organizations can define weights for each factor and prioritize use cases that have a greater sum of weighted factors.

3.3 Use Case Evaluator Application

In this section, the use case evaluation factors defined in section 3.1 are applied using the use case selection process, described in section 3.2, in order to select suitable chatbot use cases in the healthcare domain. The healthcare use cases considered are:

• Scheduling an appointment;
• Paying for an appointment;
• Performing a medical diagnosis.

Table 1 is the result of applying the step 1 of the use case selection process (see section 3.2).

With the results of the assessment of each use case, it is possible to perform the step 2 of the process, that filters out use cases that do not meet the defined requirements.

Because the use case of performing a medical diagnosis (UC3) does not meet the business rules well defined (general factor), it is excluded from the candidates to implement in chatbot (step 1.1.a of the process). Both UC1 and UC2 meet all the general factors and are considered advantageous over a GUI and over a Human, because meet at least one factor of both factor over GUI and factor over human categories. Therefore, UC1 and UC2 are not filtered out.

Finally step 3 is applied, and the use cases that were not filtered out are sorted by order of priority. UC 1 and UC 2 both meet the same factors over human. The differences are present in the Factors over GUI. UC 1 meets all the three factors over a GUI, while UC 2 only meets one factor (notifications required). Because the scheduling appointment (UC 1) meets more factors than paying appointment it is considered as more adequate and having more priority for a chatbot implementation (see Table 2). Note that the all the factors that UC 2 meets, are also met by UC 1, making the decision of priority clearer and a possible attribution of weights for each factor not needed entirely.

4. CHATBOT IMPLEMENTATION

This section explains how the chatbot that supports the use case of scheduling a medical appointment is implemented.
4.1 Components and Integrations Overview

The main goal of the chatbot implementation is to support users in the Schedule Appointment business process. To enable this use case, a bot application is created, that is served by other external components. The bot application is exposed via chat platforms, such as the Facebook Messenger or Skype, where users can interact with the bot. The high-level architecture of the solution for the implementation of the chatbot is defined in Figure 3, where the several applications and services that interact with the bot application are represented. The end of each component is explained next.

**Medclick Platform.** The Medclick platform is the system that provides the business data needed to schedule an appointment. Medclick exposes the business objects via a REST API, where objects such as the health professionals or the list of medical departments can be fetched. It also exposes the relations between the several business objects. It is possible to get, for instance, the list of all the health professionals that practice a given specialty.

**Microsoft LUIS.** Microsoft LUIS is the natural language understanding component that is responsible to give semantic meaning to the users’ utterances. It is exposed via an internet endpoint, that receives an utterance, and returns the intent and entities identified. Luis was chosen as the NLU service due to the fact that it is the service with better intent and entity detection performance from the services explored in section 2.4.

**Bot Application.** The Bot Application is a node.js web application. It is where the dialog management is defined, using the botbuilder sdk from the Microsoft Bot Framework. It is exposed in the chat channels such as Facebook Messenger. It receives the user utterances and defines the answers. This central component interacts with all the other, namely: (i) The Medclick platform, to obtain the business data needed for scheduling the appointment with the user; (ii) Microsoft LUIS application, to understand the meaning of the users’ utterances; (iii) The azure bot service, in order to connect to chat platforms such as the Facebook messenger or Skype. The botbuilder sdk, from Microsoft Bot Framework, has the unique benefit of supporting login using the OAuth2 protocol. None of the other solutions analysed in section 2.4 support external application login, that is a requirement to enable login in Medclick system from the bot application. This reason makes it the only suitable solution from the ones analysed in this research.

**Heroku Hosting Service.** Heroku is the cloud platform as a service (PAS) where the node.js Bot Application is deployed.

4.2 Dialog Management

The dialog management is based on a finite-state automata (see Figure 4). The user enters the schedule state when the intention of scheduling an appointment is detected in the user utterance. The next states of the automata have the goal of filling the different slots needed to complete the scheduling in this frame-based conversation, i.e. the department, doctor and the timeslot frames. The dialog system adapts to the information that the user provides in the first utterance, allowing to skip steps all together, or prompt a restrict subset of domain objects based on the NLU entities found in the user sentence.

In Figure 5, it is possible to observe an interaction with a user that mentions the name of the doctor and the date wanted in the beginning of the conversation. The system adapts to this information, by directly showing the most similar doctors found in medclick database, and by
searching the doctor availabilities in the time frame requested by the user.

Figure 5 - Chatbot on Facebook Messenger

4.3 Natural Language Understanding

The main goal of the dialog agent is to allow scheduling of appointments. Therefore, should exist a schedule intent. The pieces of information (entities) we want to extract from the user that wants to schedule a medical appointment are summarized in Figure 6.

After training the LUIS app with labelled utterance, it is able to identify the intent of the user and the parts of the utterance that map to the modelled entities. But we still need to map the NLU entity identified to the specific domain business object the user is referring to. The pipeline that resolves entities needs to be able to receive a user utterance, extract the NLU entities and map them to business objects. Figure 7 demonstrates the sequence of interactions required to perform the mapping.

The intrinsic differences between the entities result in different strategies regarding the type of LUIS entity chosen and how should the mapping from the entity returned by LUIS and the business to the object in medclick’s database should be performed. The summary of the strategies applied are represented in Table 3.

Table 3 - Entity Recognition and Mapping Summary

<table>
<thead>
<tr>
<th>Entity</th>
<th>LUIS Entity Type</th>
<th>NLU entity to domain Object mapping</th>
</tr>
</thead>
<tbody>
<tr>
<td>Department</td>
<td>List Entity</td>
<td>Direct match between canonical form returned by LUIS and Department Name in database</td>
</tr>
<tr>
<td>Doctor</td>
<td>Simple Entity</td>
<td>Calculating most similar name (least edit distance) in database using jaro-winkler algorithm</td>
</tr>
<tr>
<td>Date</td>
<td>Built-in (dateTimeV2)</td>
<td>Querying API for availabilities in the date or time frame resolved by LUIS</td>
</tr>
</tbody>
</table>

5. EVALUATION

This section discusses the evaluation of the chatbot use case evaluator and of the chatbot implementation. The use case of scheduling a medical appointment was selected as the more suitable (see section 3) for a chatbot, between the evaluated use cases. It was then implemented as described in section 4. Using the chatbot use case evaluator, it was defined that the implementation in a chatbot over a GUI is advantageous. In order to confirm if the UC is indeed advantageous in a chatbot, user tests are performed both in Medclick website (traditional GUI) and using the chatbot in Facebook messenger. It is also important to test the quality of the chatbot itself. The use case might be appropriate for a chatbot, but the bot implementation might not be performant enough.

5.1 Evaluation Process

In this section, it is described the process that enables the evaluation of both the chatbot and the use case selection process. The process consists of three major steps (see Figure 9) described in detail next.
5.1 Website and Chatbot Scheduling

Users are asked to schedule an appointment using both the chatbot and the website. Metrics are extracted from logs of both interactions.

- **Task completion success**: was the user able to perform the desired task?
- **Efficiency**: how long did the interaction take?

The metrics extracted exclusively from the bot interaction are:

- **Intent detection error rate**: how often did the system misclassify the user intention.
- **Entity detection error rate**: how often did the system fill an entity slot with the wrong value or did not detect that an entity was given.

With the aforementioned metrics, it is possible to directly compare the task completion success and efficiency of the GUI and chatbot interaction. The intent and entity detection error rate indicate the quality of the natural language understanding of the bot. Both website and chatbot were adapted to produce logs of the interactions, to enable later extraction of these metrics.

The information given to the user for scheduling both the appointments is:

- **Doctor Name**: The name of the doctor that the user wants to schedule
- **Time Frame**: The date when the user desires to visit the doctor.

5.1.2 Questionnaire

In the end of the website and chatbot interaction, users are also requested to answer a questionnaire. The questionnaire uses the questions of the short version of the “User experience questionnaire (UEQ)” [14], in order to evaluate the chatbot user experience (see Figure 9). Participants can rate each item on a 7-point Likert scale. The answers are scaled from -3 (fully agree with negative term) to +3 (fully agree with positive term). The questionnaire aims to evaluate the pragmatic and the hedonic quality of the bot. Pragmatic quality concerns task or goal related quality aspects, while hedonic quality relates to pleasure or fun while using the product. Half the questions concern pragmatic quality and the other half hedonic quality.

The use of the User Experience Questionnaire (UEQ) allows to quantify the user experience of the chatbot. The UEQ contains a benchmark that helps to judge how good or bad a measured product is in comparison to other products. The benchmark contains data from 9905 persons that evaluated 246 different interactive products. [15] adapted this benchmark for the short version of the UEQ, based on the same dataset.

![Figure 9 – Chatbot user experience questions asked in the questionnaire](image)

Additionally to the UEQ questions, the questionnaire created also contains a question inquiring the tester what the preferred method for scheduling appointments is, based on the two product interactions.

5.2 Evaluation Results

This section presents the results of applying the evaluation process described in section 5.1. The full process was performed by 10 test users. The demographic data collected indicates that test user age range from 18 to 64 years old with the following distribution: 18-24 (30%), 24-44 (50%), 45-64 (20%). Furthermore, regarding sex, 60% of the test users are male, and the remaining 40% female. Furthermore, half the users indicated that were using a chatbot for the first time, while the remaining half already had interacted with a chatbot before.

5.2.1 Chatbot Results

Users were able to complete the scheduling in all chatbot interactions. Furthermore, the bot was able to detect the intention of scheduling in the users’ utterance and successfully detected and map the doctor name and time entities in every interaction. The doctor name is considered successfully detected when the first doctor suggested is the one that was asked the user to request. The time entity is considered successfully detected when the bot searches for timeslots in the time frame mentioned by the user and presents the ones that exist in the database in such time frame.

5.2.2 Website and Chatbot Results Comparison

Comparing the average time users took to schedule the appointment in the website and in the chatbot, it is possible to conclude that the chatbot was more efficient, in the sense that users took less time to complete the task (see Table 4). All the users were able to complete the scheduling in both interactions, so the task completion success is the same.
Analyzing for each individual user test which ones were faster on the bot and in the website, it is concluded that most of the users (80%) were faster scheduling the appointment via chatbot than using the website.

### 5.2.3 Questionnaire Results

The results of the answers to the UEQ in the questionnaire can be found in Table 5. Note that the mean value is already in the -3 (most negative) to +3 (most positive) scale.

<table>
<thead>
<tr>
<th>Item</th>
<th>Mean</th>
<th>Variance</th>
<th>Std. Dev.</th>
<th>No.</th>
<th>Negative</th>
<th>Positive</th>
<th>Scale</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>2.2</td>
<td>0.6</td>
<td>0.8</td>
<td>10</td>
<td>obstructive</td>
<td>supportive</td>
<td>Pragmatic Quality</td>
</tr>
<tr>
<td>2</td>
<td>1.9</td>
<td>0.5</td>
<td>0.7</td>
<td>10</td>
<td>conventional</td>
<td>inventive</td>
<td>Hedonic Quality</td>
</tr>
<tr>
<td>3</td>
<td>2.0</td>
<td>0.9</td>
<td>0.8</td>
<td>10</td>
<td>usual</td>
<td>leading edge</td>
<td>Hedonic Quality</td>
</tr>
<tr>
<td>4</td>
<td>2.5</td>
<td>0.5</td>
<td>0.7</td>
<td>10</td>
<td>typical</td>
<td>imaginative</td>
<td>Hedonic Quality</td>
</tr>
<tr>
<td>5</td>
<td>1.8</td>
<td>0.2</td>
<td>0.7</td>
<td>10</td>
<td>not interesting</td>
<td>interesting</td>
<td>Hedonic Quality</td>
</tr>
<tr>
<td>6</td>
<td>1.9</td>
<td>0.4</td>
<td>0.6</td>
<td>10</td>
<td>usual</td>
<td>leading edge</td>
<td>Hedonic Quality</td>
</tr>
<tr>
<td>7</td>
<td>2.2</td>
<td>0.5</td>
<td>0.7</td>
<td>10</td>
<td>usual</td>
<td>leading edge</td>
<td>Hedonic Quality</td>
</tr>
</tbody>
</table>

Considering the averages of the Pragmatic and the Hedonic Quality fields, it is possible to yield the following general results:

- **Pragmatic Quality**: 2.457;
- **Hedonic Quality**: 1.950;
- **Overall**: 2.213.

With these results the benchmark proposed by [15] is applied, as described in section 5.1.2. The chatbot is evaluated as Excellent (in the range of the 10% best results) in pragmatic, hedonic and overall quality, as the values obtained for each category are higher than the requirement of the benchmark level.

When asked what the preferred method for scheduling was, most users (70%) chose the chatbot as preferable, 10% selected the website as preferable and 20% do not prefer one method over another (indifferent).

### 5.3 Evaluation Conclusions

The fact that all the users were able to complete the scheduling using the chatbot, and that the UEQ benchmark positions the user experience in the range of the 10% best results, indicates the use case of scheduling a medical appointment as suitable for a chatbot. Furthermore, considering that: (i) 70% of the users answered that prefer to schedule an appointment using the bot instead of a website (GUI), and that (ii) the chatbot interactions were more efficient in 80% of the tests, it is possible to consider the chatbot implementation as advantageous over a traditional graphical user interface.

The suitability of the use case for a chatbot combined with the observable advantages over a GUI, indicates that the use case selection process (see section 3.2) selected an appropriate use case for implementation.

### 6. Conclusions

The heavy usage of chat platforms allied with the developments in natural language understanding create an opportunity for organizations to engage users using dialog agents. Chatbots lie between a human operator and a graphical user interface (GUI) application. Chatbots share the advantage of the more natural interaction of human operator and the automation benefits of GUI applications.

Although several advantages can be found in chatbots, it is not obvious how can organizations define a priori the use cases that are suitable for a conversational user interface implementation. Evaluating the factors that are required for a viable implementation (general factors), and the factors of the use case that make it beneficial over a GUI application or over a human, it is possible to further understand if a use case is not only viable for a chatbot, but if it is beneficial over a GUI or a human operator. This research proposes a process that using the evaluating factors gathered aims to aid organizations in this analysis.

The application of the use case selection process to the health care domain, indicates scheduling a medical appointment as beneficial to be implemented in a dialog agent. After conduction user tests using the bot implementation of this use case, it was possible to conclude that the chatbot was more efficient when comparing to a traditional GUI (website), and that most users preferred to schedule an appointment using the bot over the website.

This research enables organizations to make more informed decisions regarding dialog agent implementations, offering a systematic process for use case suitability assessment. In the healthcare domain, it was possible to evaluate several use cases using this process.

### 7. Future Work

Future work can be performed in order to further develop and evaluate the use case evaluation process. Regarding the evaluation of the process, it can be further validated by using it to assess other use cases and compare the success of the implementation with the result of the assessment using the process.

The evaluation of the chatbot implemented can also be improved, by adding more variability of entities given (doctor name, time, specialty) and by increasing the number of test users. It would be beneficial to use a corpus of real appointment schedule conversations, in order to train the chatbot natural language understanding module. The usage of such corpus would also enable to further evaluate and improve the intent and entity detection rate of the dialog agent, even though the proposed bot can learn as more interactions are performed.
ACKNOWLEDGEMENTS

I would like to thank my family for their support and encouragement throughout all my academic path.
I would also like to thank my thesis advisor, professor André Vasconcelos, for the guidance and feedback offered during the elaboration of this dissertation. I also want to thank Medclick and Rui Cruz, for the support and for the opportunity to develop this work in the context of Medclick.

Last, but not least, I would like to express my gratitude to my friends, that through their presence and support were also of great importance.

REFERENCES


