Project Analysis

# Background of Project

1. Background:

IBM is an American funded technology company headquartered in New York, USA. IBM offers a wide variety of cloud platform and cognitive solutions to its customers, Watson being their most popular and sophisticated creation.

1. Identification of the problem:

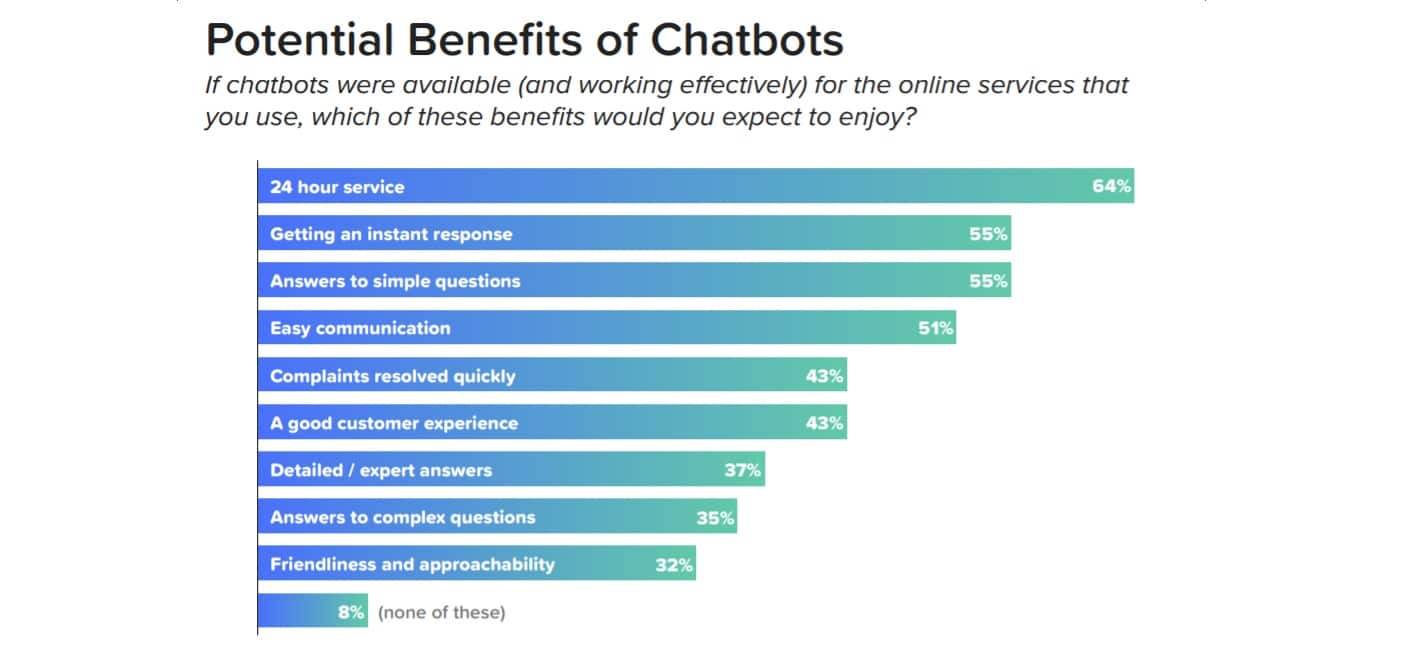
The IBM’s website currently offers many types of services such as business consulting and financing, and also many products like IBM’s software and cloud services. This wide variety of solutions make it very difficult for users to find what they want, which makes the website time consuming and less useful for the user. One of the suggested solutions to this problem is to facilitate the use of the website by creating a chatbot. The main problem of the website is that users may find difficult to find what they need, and they always end up leaving the page or emailing/contacting customer service. The latter will waste the time that customer service could have had if the chat bot was implemented, as it would automatically and independently answer the questions of the user without human help. This would greatly reduce the time humans spend answering email, leaving them more time to do tasks that benefit the company and making it more efficient. Another problem that IBM faces with their website is that information about IBM or its products is sometimes they are hidden in the many pages the website has, making it difficult for a client to find them. This problem makes the website less user-friendly and more prone to not being revisited by the client. A chatbot could solve this problem by offering users links to the information that they are searching for, in order to make the website feel more useful and efficient to the user, and therefore more likely to be visited again.

# Description of the current system

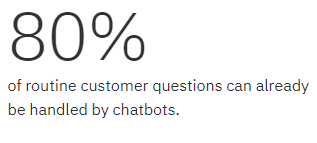
The current system that IBM has implemented in their website is very conventional. They have created a page called ‘Contact IBM’ where a user can contact the support team via phone numbers, email and even mail. All these options may vary as different countries have different phone numbers but in general, the layout of the contact page is the same. In all of the options that they offer, human contact must be made between the user and the support team meaning that the user may have to wait for an answer as the system is not automated. IBM’s system also offers an option where users can ask general questions (or FAQ) and give website feedback. This allows customers to get a quick answer to their questions, but they will still need to use email.

# Research into chatbots

A chatbot is a computer program or an artificial intelligence which conducts a conversation via a textual method. This type of programs are commonly used to simulate a conversation with a human, therefore being really useful tasks that can be automatized such as customer service or information gathering. Some chatbots use many simpler systems that scan for keywords within the input, then pull a reply with the most matching keywords, or the most similar wording pattern, from a database.

Customer help service chatbots are quickly increasing in popularity, and it’s easy to see why. This evolving technology makes it easier for businesses to engage their customers, cut down on their workload and proactively prevent minor customer service issues from becoming major problems. This enables your customer service agents to dedicate more time to resolving more complicated customer issues and provide faster feedback for common customer questions (FAQ).

Conversational chatbots designed appropriately makes a customer feel that he or she is conversing with a human being, not a robot. It makes customers feel at home by offering information or services in a friendly manner. For a business website, a chatbot is just as a salesperson, who is read to help the visitor with the product information, business details/info and much more.

According to Oracle, 50 % of customers want a business or service provider to stay accessible 24/7. This means that by having chatbot, IBM could reach more customers as a result of longer availability hours because of the chatbot independence. Most of the participants were highly likely to interact with bots, and for multiple reasons. Among these reasons, 58% said they would even use a chatbot in an emergency. This level of trust carried over to the finding that only 14% of respondents would find the experience of speaking to a machine unnerving.

AI is the technology through which chatbots work as it helps in making customer service more efficient, productive, and cost effective by feeding customers with the relevant answers. Below are the steps that are involved in AI Chatbot Development:

1. Knowledge base software: A knowledge base software enables a chatbot to do more than just simply acting as a gateway to FAQs. Through a knowledge base, it delivers a seamless customer experience by providing instant resolutions to make interactions more user-friendly.
2. Front-end interface: It connects the chatbot to websites, e-mail or applications such Telegram. In order to deliver quick resolution on the same.
3. Natural Language Processing (NLP): It helps recognize the user intent by parsing their messages and understanding sentences to take actions.
4. Dialogue manager: It maintains the logic in a conversation. Examples of software that can be used to improve the dialog of the chatbot could be the many API’s that Watson offers.

It is the mix of knowledge management and technology that gets chatbots being put to effective use. With features like easy search capability, device database, and support in multiple languages, its integration with AI backed chatbot facilitates delivery of contextual responses to resolve customer queries without the need of human help.

# Prospective users

The primary users for this online tool would be any user that’s interested on IBM’s cloud products and wants to get more information about the website and the contents within it, without the trouble of spending time searching for it. The chat bot can also be used by companies to search through the catalogue of IBM’s products and to get business information that may be useful to them. The client that uses this help tool is usually a user that wants to be in minimum human contact as possible and wants an immediate response.

## Identification of user needs

1. The chatbot should be able to understand the user without the need of human help, so that responses can be given as fast as possible.
2. The chatbot should be able to be activated only by the user if he chooses to do so. The user should also be able to decide when he wants to finish the chat.
3. The program should be able to understand what’s required of him by the user, depending on how the question or response is formulated.
4. The program is expected to give the useful information in the form of a response or link to the user when a question or a request is made to the chatbot. This information has to be relevant to the subject, so it can be of help to the user.
5. The user should be able to click the link and remain in the chatbot conversation, so any further questions can be asked.
6. The program should be able to redirect a user to the support team by giving the user the email contact and the telephone number. This should be done only when the question made by the user can only be answered by a human.

# Objectives

Project objective:

The main objective of the project is to create a fully functional chatbot that is autonomous and non-dependent of human help. The chatbot should be created in Python (a scripting language) with the help of different libraries such as: openpyxl (a Python library to read/write Excel documents), telebot (A Python implementation for the Telegram Bot API). Furthermore, the chatbot will be hosted in Telegram with help of its BOT API framework by using the mentioned ‘telebot’ library. The BOT API is an HTTP-based interface created for developers keen on building bots for Telegram.

The chatbot should have a basic AI that reads the sentences sent by the user, and finds patterns or keywords that will guide the AI to the correct answer that contains useful information for the user. This AI will find the patterns/keywords by comparing every word to a database of known patterns/keywords (database made in excel and managed by openpyxl), and if successfully found, the AI should select an answer suitable to that specific pattern/keyword and then display it to the user.

User Need 1: ‘The chatbot should be able to understand the user without the need of human help, so that responses can be given as fast as possible.’

Specific objectives:

1. Create a simple AI that can understand the requests that the user makes to the chatbot. This means that I have to create an algorithm that tears down the messages of the user and logically chooses a suitable answer.
2. Make the AI independent, meaning that the AI should only rely on itself to analyse and output data. Independence is the key to this project as a human may not always be there to help it when a user is talking to it.
3. If the user uses another language, the program will tell the person that the language he is using is not supported by the chatbot. This feature has to be implemented in as the only language that my bot is going to be based on is English.

User Need 2: ‘The chatbot should be able to be activated only by the user if he chooses to do so. The user should also be able to decide when he wants to finish the chat.’

Specific objectives:

1. Allow the user to find the chatbot in Telegram, by searching for it under a specific name (name of the chatbot created later on the program). The bot can be created using Telegram’ API, and then manipulated with my python program.
2. Allow the user to start a conversation by sending his message to the bot. This means that I have to write a function that connects my program to the Telegram bot and then create another procedure that reads whatever the user wrote in Telegram.
3. Allow the user to see the response on the chat. This is achieved by creating a procedure that sends the response of the chatbot to Telegram, via an API call.

User Need 3: ‘The program should be able to understand what’s required of him by the user, depending on how the question or response is formulated.’ -- ‘The program is expected to give the useful information in the form of a response or link to the user when a question or a request is made to the chatbot. This information has to be relevant to the subject, so it can be of help to the user.’

Specific objectives:

1. The program should retrieve information from a database and then it will answer the question with a sentence. This has to be done by making a procedure in python that connects to the database in order to read/write information in it. The library ‘openpxl’ will allow me to perform the task of manipulating databases created in Excel documents.
2. If is an information request, the program should retrieve a link from the ‘link’ database that sends the user a link/specific data depending on the request.
3. The program will choose a suitable option depending on the keywords that the user has used. This should be achieved by creating an algorithm that dissects the sentence into an array of strings. After this is done, the program should search through the ‘keyword’ database in order to search for keywords within the user’s sentence.
4. If the program detects that the user has used a keyword, it should then give out a response that is related to that keyword. This responses should be stored in the ‘keyword’ database and should only be used if their corresponding keyword is triggered by the program.
5. If the user hasn’t used any keywords, the program will ask the user to specify more his request. This should only happen when the program doesn’t detect any keywords being used I a sentence.
6. Allow the user to see the information or link in the chat. When a response is selected by the program, it should be sent to the Telegram bot via API and then displayed to the user.

User Need 4: ‘The user should be able to click the link and remain in the chatbot conversation, so any further questions can be asked.’

Specific objectives:

1. Allow the user to see the link on the chat.
2. The link should open a new window/tab that will redirect the user to the information that he is requesting.
3. Allow the user to remain in the chat if he wishes after clicking the link.

User Need 5: ‘The program should be able to redirect a user to the support team by giving the user the email contact and the telephone number. This should be done only when the question made by the user can only be answered by a human.’

Specific objectives:

1. Detect if the question that the user has asked contains personal matters, or is too complex for the bot to answer.
2. If the question is personal or confidential, the bot will display the contact details of the support team of IBM.
3. Create a procedure that detects when the user is requesting business information such as a phone number or the email of the support team. Then, the program should message the user back with IBM’s contact info, allowing the user to see the contact details.

# Chosen solution

The system will be programmed and developed in Python, as it is a high-level programming language that I’m very familiar to. Python it's also often used as a “scripting language”. This means that it can automate specific series of tasks, which is very convenient when creating a chatbot, making it more efficient and a better language for this project than Java or the C family.

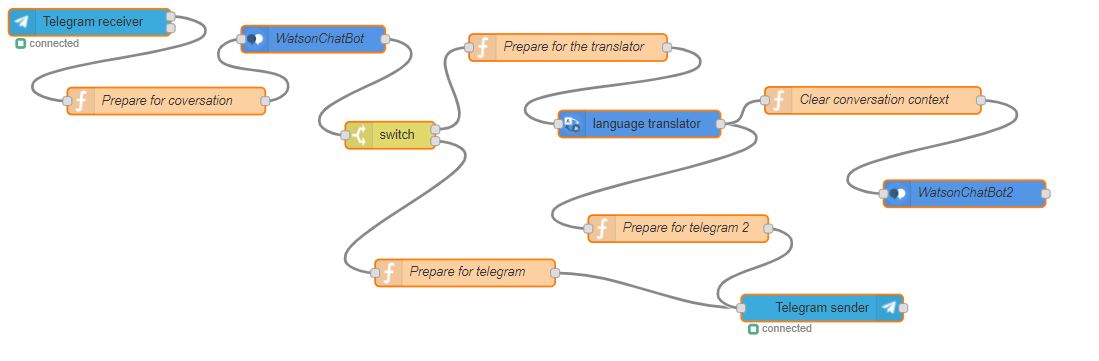
The objective of this project is to produce a multi-functional Chatbot, which has a GUI that is both easy to understand and manage for the user. The interface will be required to be intuitive and also easily accessible for IBM’s clients, meaning that it can be used in many different platforms such as phones, laptops or tablets. In order to meet this objective, I will use Telegram’s API which allows me to use their Bot system as a GUI for my Chatbot.

# Log/diary of research

|  |  |  |  |
| --- | --- | --- | --- |
| Source | Link | Used for | Date |
| 1 | https://www.ibm.com/uk-en/ | General view on how their website set-up | 17/08/18 |
| 2 | https://www.ibm.com/contact/uk/en/?lnk=flg-cont-uken | Research into what types of support IBM offers to its clients | 17/08/18 |
| 3 | https://en.wikipedia.org/wiki/Chatbot | Definition of a chatbot and its common uses | 28/08/18 |
| 4 | https://chatbotsmagazine.com/10-benefits-of-integrating-a-chatbot-with-your-business-website-851af2401379 | Benefits of chatbots in business websites | 29/08/18 |
| 5 | https://www.boostability.com/ai-chatbots-and-consumers-alexa-what-does-this-mean-for-business | Data about chatbots | 5/09/18 |
| 6 | https://www.quora.com/What-are-the-steps-in-an-AI-Chatbot-development | Steps on how to build a bot | 9/09/18 |
| 7 | https://www.pyxll.com/blog/tools-for-working-with-excel-and-python/ | Python libraries for excel | 11/10/2018 |

Product Design

# Prototype 1

For the first prototype of the project, I decided to create a simple chatbot that didn’t require much coding to work properly. I settled to do it this way, so I could gain a general idea of how a chatbot is built, and what requirements does it need to be able to function and think like a human.

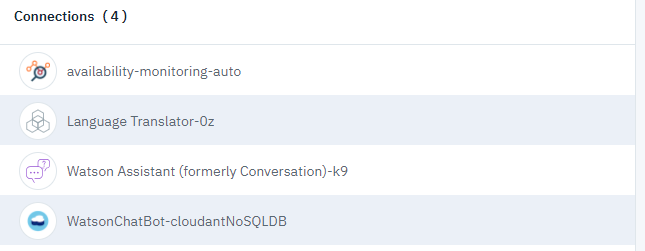
## Prototype 1 steps:

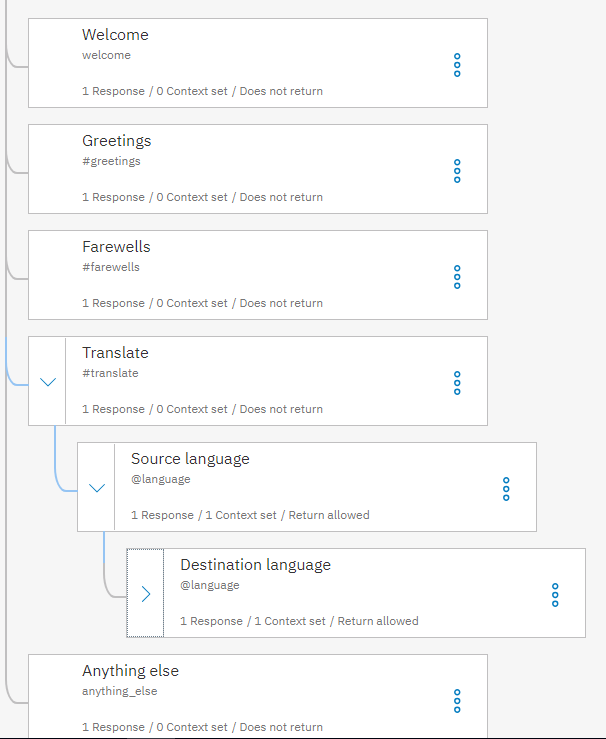
* First, the user will have to add the chatbot in the Telegram app, which is under the name ‘WatsonELchatbot’. After that, he will send a message to the bot via Telegram, where the chatbot is set-up. This message may contain a question or just a simple conversation starter.
* When the message is sent, it will go through my personal IBM cloud. This will then start the chatbot and the conversation.
* After the message is received in the cloud, Watson Conversation Assistant will read and understand the sentence by using IBM’s language AI.
* The switch statement in the flowchart divides the conversation in two, depending on what the user asks for in the message:
* If the user asked for the translator, then the program will choose the translator route. This way the IBM Language Translator will ask for an input (sentence) to be translated and will also ask for the language to be translated to. The AI will then translate the message to the required format and display it to the user. After this process is finished the user will get two choices:

1. Stay in the translator mode and keep translating messages.
2. Clear the conversation and start from the beginning.

* The other option is to keep talking to the chatbot in a normal conversation.

## Prototype 1 evidence:

This screenshot shows the connections that the chatbot cloud is using in order to make the AI work correctly. The first connection monitors the data going through the cloud and check for the availability of the cloud. The second connection manages the translator module that the chatbot uses within its conversation with the user. The third one connects the chatbot with Watson’s language AI. This AI is used so the program can understand basic English and formulate an answer to respond the user. The last connection directs the cloud to the conversation flow that was set by me (explained in the next paragraph).

Conversation flow:

This flowchart represents how the conversation will be managed by the AI while in conversation with the user.

Welcome: at the beginning of every new conversation the bot will welcome the user.

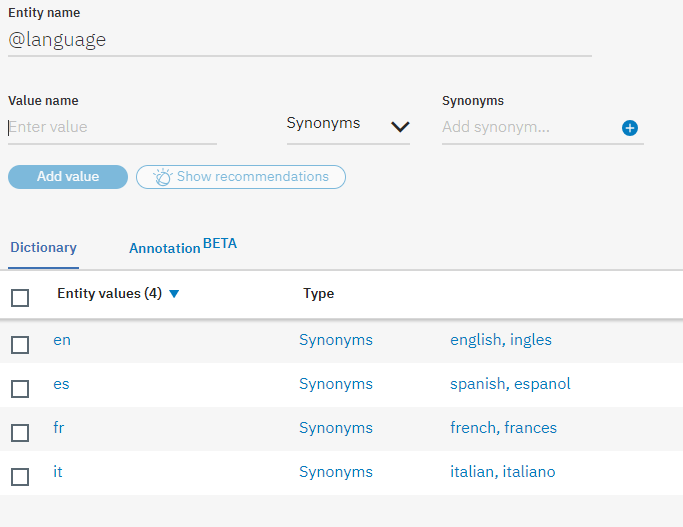
Greeting: if the user types a sentence containing words like ‘hi’ or ‘hello’ then the A will choose the welcome flow.

Farewells: if the user types a sentence containing words like ‘bye’ or ‘see you’ then the A will choose the welcome flow.

Translate: If the user chooses to translate something, the bot will take the sentence, and then the chatbot will first ask about source language of the sentence and then it will ask about the destination language.

Anything Else: if the user talks about a subject that the AI cannot understand, then the AI will tell the user that the conversation has been unsuccessful with a message like ‘sorry I didn’t understand what you said’.



Translate function:

The translate function is activated, as we can see from the screenshot, by different trigger sentences such as ‘Can you translate?’. The Watson’s Ai will also understand one of these sentences even If one of these sentences is miswritten by the user. Then, if the user chooses to translate some text, it will ask the user for the native and the foreign language so that the AI knows what is working with.