**SRS**

Software requirements specification

**By**

MUKESH A

111722106014

SAKTHIVEL V

111722106017

SEBASTIN JERRY J

111722106018

**From**

BE Mechanical Engineering

**On**

**Chat Bot**

# ***CHAT BOT***

**Table of content**

**1. Introduction**

**1.1 Purpose of Document**

**1.2 Project Scope**

**1.3 Overview of Document**

**2. Description**

**2.1 Product Perspective**

**2.2 Product Features**

**2.3 User Classes and Characteristics**

**2.4 Constraints**

**2.4.1 Limited Question Scope**

**2.4.2 Language**

**2.5 Assumptions and Dependencies**

**2.6 Requirements Apportioning**

**3. Non-functional Requirements**

**3.1.1 Modularity**

**3.1.2 Accuracy**

**3.1.3 Fast Response**

**3.1.4 Security**

**3.2 Web interface/Mobile application**

**3.2.1 Ease of Use**

**4.Glossary**

1. **Introduction**

****

**1.1 Purpose of Document**

This document will provide all of the requirements for the project Drexel Chatbot. It will serve as a reference for developers and customers during the development of the final version of the system.

**1.2 Project Scope**

Drexel Chatbot (Drexel natural language query service) is an AI chatbot that receives questions from users, tries to understand the question, and provides appropriate answers. It does this by converting an English sentence into a machine-friendly query, then going through relevant data to find the necessary information, and finally returning the answer in a natural language sentence. In other words, it answers your questions like a human does, instead of giving you the list of websites that may contain the answer. For example, when it receives the question "What time does the gym close today?", it will give a response “The gym closes at 10pm today.

” The main objective is creating a Web API, and sample web, mobile, and text messaging interfaces that demonstrate the use of the API. The goal is to provide Drexel students and faculty a quick and easy way to have their questions answered, as well as to offer other developers the means to incorporate Drexel Chatbot into their projects.

**1.3 Overview of Document**

**1.Revision History:​** Provide the date of, reason for, and people who were involved with the modification of this document.

**2.Introduction:​** Provide an overview of the application, explain the objectives and goal of the project and describe the document structure.

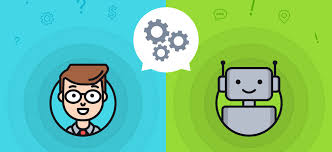
**3.Overall Description:​** Provide the specification of the system model, the classes model and the main constraints.

**4.Functional Requirements:​** Provide the analysis of the requirements by feature.

**5.Non functional requirements:​** Provide some other constraints that affect performance, safety and security.

**6.Use Cases:** ​Provide possible scenarios where the user interacts with the Web API and sample applications.

**7.Glossary:​** Definitions of terms used.



**2. Description**

**2.1 Product Perspective**

Most of the search engines today, like Google, use a system (The Pagerank Algorithm) to rank different web pages. When a user enters a query, the query is interpreted as keywords and the system returns a list of highest ranked web pages which may have the answer to the query. Then the user must go through the list of webpages to find the answer they are looking for. Drexel Chatbot, however, will try to understand the query and provide a definitive answer.

There will be four main units to the system working together to understand the question and return an appropriate answer:

● Generic question construction - capable of taking a natural language question and making it more generic.

● Generic answer construction - capable of taking a generic question template and providing a generic answer template.

● Generic answer population - capable of taking a generic answer template and populating it with information from the database to form an answer.

● Information extraction - capable of finding information through structured or unstructured websites, and storing that information in a database.

**2.2 Product Features**

The major features for Drexel Chatbot will be the following:

● Web API: An​ API call will include a question in the form of a query string url parameter and the service will reply in JSON.

● Natural Language Processing: T​he system will take in questions written in standard English.

● Natural Language Responses: T​he answer to the question will be written in standard and understandable English.

● Information Extraction: T​here will be a database containing all the information needed, populated using information extraction techniques.

**2.3 User Classes and Characteristics**

The two classes of users for this system are described below:

● API users API users consist of application developers who want to incorporate Drexel Chatbot API into other software applications.

● Mobile app/Web app/SMS users These users consist of non-technical users who want to get answers for their questions. These users ask questions and get answers with mobile, web, or text messaging interfaces. This class of users include Drexel’s current and prospective students.

**2.4 Constraints**

**2.4.1 Limited Question Scope**

Creating a chatbot able to answer every single question about Drexel is not possible to implement with current technology and within the duration of the project, so the system will be able to answer questions about limited topics.

**2.4.2 Language**

The system will only support questions in standard English.

**2.5 Assumptions and Dependencies**

***Keras***is a library for creating and using neural networks. It should provide us with all the functionality we need, however if it is in someway deficient, then it will be replaced with a different library.

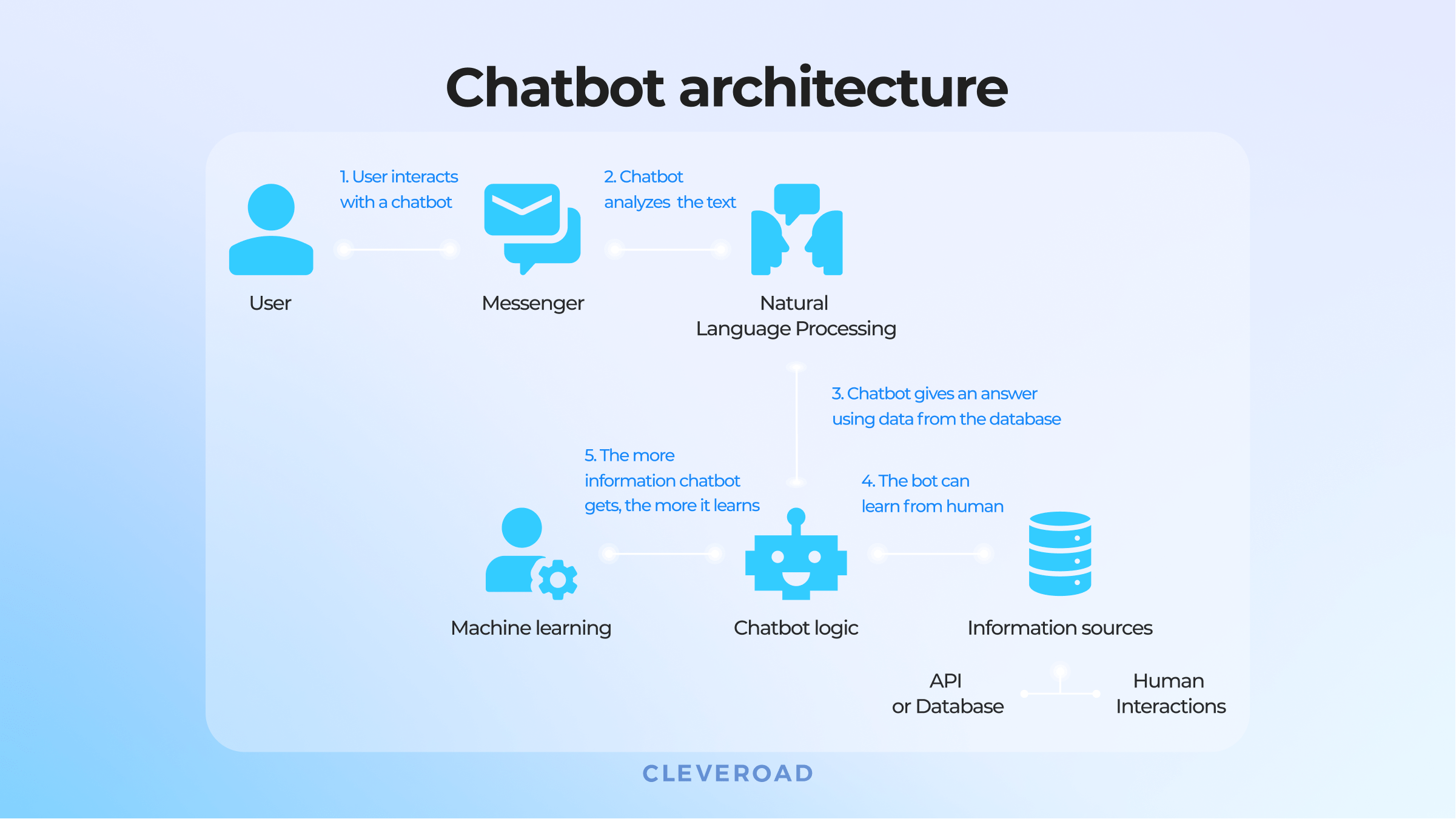
***BeautifulSoup*** is a library for parsing HTML documents. It should be all we need to extract text from a webpage, but may be replaced if necessary. We will develop the project using Python and MySQL database.

**2.6 Requirements Apportioning**

Each feature will be assigned an importance value. The project will be complete if all the features of Priority 1 and at least 50% of features of Priority 2 are implemented. No Priority 3 requirements are necessary.

|  |  |
| --- | --- |
| **Priority** | **Meaning** |
| **1** | *Essential, the project will not work without this feature. This feature will be implemented.* |
| **2** | *Important, the scope of the project will be significantly hindered without this feature. This feature will likely be implemented.* |
| **3** | *Desired, this feature complements the core functionality. This feature will be implemented, time allowing.* |

1. **Non-functional Requirements**

****

**3.1 API**

**3.1.1 Modularity**

The system will be designed in such a way that the algorithms for the four main units will be able to be easily swapped out. Priority 1.

**3.1.2 Accuracy**

The overall accuracy of the Web API’s response will be measured using a developer-made testing set. Priority 1

The accuracy of the Generic Answer Construction unit will be close to 70%. Priority 2

The accuracy of the Generic Answer Population unit will be close to 70%. Priority 2.

On-campus dining locations, hours, food types, etc. will have accuracy greater than 50%. Priority 2.

**3.1.3 Fast Response**

The average time for the server to respond, over the question testing set, will be less than or equal to 2 seconds. Priority 2.

**3.1.4 Security**

The connection between the Web API and the programs will use HTTPS, for security. Priority 3.

**3.2 Web interface/Mobile application**

**3.2.1 Ease of Use**

A new user will make less than 3 mistakes in 5 minutes after 5 minutes of use. Priority 1.

**4.Glossary**

● **Chatbot:** An interface, usually text based, specializing in the mimicry of natural language conversation. AKA “artificial conversational entity.”

● **GUI**: Graphic User Interface, a type of user interface that allows users to interact with the software through graphical icons (e.g. buttons, etc.).

● **HTML:** Hypertext Markup Language, a standardized system for tagging text files to achieve font, color, graphic, and hyperlink effects on webpages.

● **JSON:** JavaScript Object Notation, a data-interchange format that is commonly used in exchanging data over the Internet.

● **Pagerank:** PageRank is an algorithm used by Google to rank websites. It works by counting the number and quality of links to a page to determine a rough estimate of how important the website is. The underlying assumption is that more important websites are likely to receive more links from other websites.

● **SMS:** Short Message Service, the text messaging protocol of cellular telephones.

● **Standard English:** the language that can be understood by English-speaking high school graduates.

● **URL:** Uniform Resource Locator, an address to a resource on the Internet.

● **URL parameter:** parameters whose values are set in a webpage’s URL.

● **Web API:** an application programming interface (API) for either a web server.

● **Web scraping:** web scraping is a technique employed to extract large amounts of data from websites whereby the data is extracted and saved to a local file in your computer or to a database.

