Disease Knowledge Base Generation using Natural Language Processing Techniques

Submitted in partial fulfillment of the requirements for the degree of

Bachelor of Technology

in

Computer Science Engineering

by

Bukkasamudram Maheedhar Reddy - 19BCE0121

Vadarevu Hemanth Sai Sri Harsha - 19BCE2237

Gonuguntla Srinivasa Rao – 19BCE0119

Under the guidance of

Dr. Swathi J.N

SCOPE

VIT, Vellore.

VIT®

Vellore Institute of Technology

(Deemed to be University under section 3 of UGC Act, 1956)

2023, May

DECLARATION

I hereby declare that the thesis entitled "Disease Knowledge Base Generation using Natural Language Processing Techniques" submitted by me, for the award of the degree of *Bachelor of Technology in Computer Science Engineering* VIT is a record of Bonafide work carried out by me under the supervision of Swathi J N.

I further declare that the work reported in this thesis has not been submitted and will not be submitted, either in part or in full, for the award of any other degree or diploma in this institute or any other institute or university.

Place: Vellore

Date: 20-05-23

Hemanth

V. Hemanth (19BCE2237)

Mahee

B. Maheedhar (19BCE0121)

G. Srinivasa rao

G.Srinivas (19BCE0119)

Signature of the Candidate

CERTIFICATE

This is to certify that the thesis entitled "Disease knowledge Base Generation Using Natural Language Processing Techniques" submitted by **Bukkasamudram Maheedhar Reddy (19BCE0121), Vadarevu Hemanth Sai Sri Harsha (19BCE2237), Gonuguntla Srinivasa Rao (19BCE0119)** , **SCOPE**, VIT, for the award of the degree of *Bachelor of Technology in Computer Science Engineering,* is a record of Bonafide work carried out by him / her under my supervision during the period, 01. 07. 2022 to 30.04.2023, as per the VIT code of academic and research ethics.

The contents of this report have not been submitted and will not be submitted either in part or in full, for the award of any other degree or diploma in this institute or any other institute or university. The thesis fulfills the requirements and regulations of the University and in my opinion meets the necessary standards for submission.

J. N. Swathi

Place: Vellore

Date: 20-05-23

Signature of the Guide

Internal Examiner

External Examiner

Head of the Department

CSE (Core)

ACKNOWLEDGEMENTS

We would like to express our sincere thanks to our project guide **Dr. Swathi J.N** for their guidance and support in completing the project. Your thoughtful suggestions and feedback were of great assistance to us due to which we were able to successfully finish the project.

Throughout the project, our guide has helped us perform research in the specified area and improve our understanding in the area of Natural Language Processing. We greatly appreciate the assistance of the **Head of Department, Dr. Vairamuthu S** and **Dean of SCOPE** School, **Dr. Ramesh Babu** in keeping us informed of all the necessary formalities.

We would like to thank **Vellore Institute of Technology** for providing us with the opportunity to work on this project. Finally, we would like to express our gratitude to our families for their assistance and emotional support during the entire duration of this project, and we are deeply grateful to everyone who has contributed to the successful completion of this project.

Student Names

Bukkasamudram Maheedhar Reddy
Vadarevu Hemanth Sai Sri Harsha
Gonuguntla Srinivasa Rao

Executive Summary

Knowledge bases are a form of online databases having information about various attributes of a particular subject or topic. These provide easy and open access to information at an attribute level. They also help in the process of data collection, sharing and duplication among both end-users and researchers. The data in a knowledge base usually comes from a collection of trusted sources under the guidance of experts in that subject. Since a knowledge base contains information gathered from various sources and is then refined according to specific subject related entities, it is very useful for customers (end-users) to gain an overall perspective at a single place. Knowledge bases also promote the concept of collaboration where researchers or other knowledge bases can use information from one another to increase their productivity.

Disease or Medical knowledge bases are extremely useful in cases where a new disease outbreak has occurred, since the professionals and researchers need information quickly regarding any gene, symptomatic, phenotypes, clinical and drug trials conducted previously on similar disease groups. These also prove helpful for further research on a particular drug or a surgical method for any existing diseases. Presently, this information is collected in the form of Electronic Health Records (EHRs) which are generally created by health care professionals in hospitals and testing centers with data regarding their basic clinical trials along with some gene information. Creating a knowledge base in the medical domain has been a challenge due to limited and complex knowledge of the medicine language. We used Natural Language Processing techniques like Named Entity Recognition, Web Scraping and Text Summarization to gather information at a scale regarding various useful entities. The information was gathered from trusted sources and journals in the medical domain and Sci-Spacy, a trained NER model for processing biomedical text was used. We gathered information through web scraping and NER which then was processed using text summarization to remove any trivial and redundant data. We gathered information both on general information like symptoms, causes and treatment methods for diseases along with some information regarding the gene formations and drugs/clinical trials done. We then stored all gathered information in a NoSQL database and provided it to the end-users using a front-end website.

Keywords: Knowledge Base, Electronic Health Record, Medical Knowledge Base, Named Entity Recognition, Web Scraping, Text Summarization.

CONTENTS   Page No.

List of Figures

List of Abbreviations

NLP	Natural Language Processing
NER	Named-Entity Recognition
BERT	Bidirectional Encoder Representations from Transformers

1. INTRODUCTION

1.1. THEORETICAL BACKGROUND

Healthcare databases are an important part of the healthcare system, and these are commonly used in the form of EHR (Electronic Health Records) where healthcare providers enter basic clinical data and some advanced data regarding diseases and their gene encodings.

These EHRs help in bringing data regarding various entities about a particular disease together which would help in the accuracy and speed of further research and study regarding these diseases either it being in testing the benefits of a new drug or assessing the reactions and symptoms of a disease and thus understanding its evolution.

A major limitation that is seen with such EHRs is that healthcare professionals are not trained in gathering and storing this data and the data may sometimes be imprecise with variable quality and completeness and not completely useful for research purposes and that is where Natural Language processing and NoSQL databases would help.

1.2. MOTIVATION

Knowledge Base creation in the medical domain has always been a challenging task and the present Electronic Health Records held by the healthcare professionals cannot handle data on a scale and various limitations occur when newer technologies like Machine Learning are used on the data for research due to imprecision of the data and the anomalies present. Gathering information and maintaining it on a scale is also a task that cannot be done using manual work and requires technologies like AI and Natural Language Processing. Some EHRs contain only basic information about diseases while some contain advanced information regarding phenotypes and gene encoding. We try to include both general information and data about drug and clinical trials along with gene encodings of diseases using Natural Language Processing.

1.3. AIM OF THE PROPOSED WORK

The aim of this project is to build an online “database website” displaying information about diseases. The interface should be simple and open to all. The data collected should contain both basic and advanced information from the diseases` domain. Information should be gathered optimally using Natural Language Processing techniques like Web Scraping, Text Summarization and Named Entity recognition.

1.4. OBJECTIVES OF THE PROPOSED WORK

- Identify necessary entities for every disease and identify the sources for gathering information.
- Use Natural Language processing techniques like Named Entity Recognition and Web Scraping to gather useful information from sources.
- Use text summarization to remove redundant and trivial information.
- Deploy the resultant data into a NoSQL Database (MongoDB).
- Create a front-end website to display all the gathered information as per their entities thus creating the final knowledge base.

2. LITERATURE SURVEY

2.1. SURVEY OF EXISTING MODELS/WORK

When a new illness outbreak occurs like we witness 3 years ago outbreak of covid -19, professionals and researchers need to know the basic and advanced information about that particular or similar diseases as soon as possible like the information about any gene, symptomatic, phenotypic information and clinical trials carried out in the past on similar disease groups.In such cases we need a disease database which give specific information about diseases in one interface. Due to the complex and limited knowledge of the medical language, developing a knowledge base in the medical field has been difficult because there are limited trusted sources and continuous updates of new diseases and drugs which makes it difficult to maintain a knowledge base up-to-date. We gathered information both on general information like symptoms, causes and treatment methods for diseases along with some information regarding the gene formations and drugs/clinical trials done.

Existing methodologies for NER include using BERT where they initialize BioBERT with weights from BERT, which was pretrained on general domain corpora and then BioBERT is pre-trained on biomedical domain corpora. Finally, BioBERT is fine-tuned and evaluated on three popular biomedical text mining tasks. They also test various pre-training strategies with different combinations and sizes of general domain corpora and biomedical corpora and analyze the effect of each corpus on pre-training[1]. While implementing NER techniques on biomedical related data, other researchers' approach for medical NER is using pre-trained language models and a domain dictionary. First, they preprocessed labeled medical texts and constructed a medical entity dictionary. Second, they introduced a pseudo labeling method and used this to annotate un-labelled texts with pseudo labels. Third, they used the BiLSTM-CRF technique to fine-tune the pre-training of language models[2]. Another work done on biomedical related literature used the concept of hybrid NER where the methodology is divided into two phases, as follows: I Creating the dataset; (ii) training and testing the model. The resulting dataset is saved in JSON format. The saved data is loaded into the main Python script to train the model. The proposed method has been tested with both a blank model and an existing model. Using the Transfer Learning concept, the existing model is retrained. CNN is crucial in training the model. The hybridization of dictionary and model retraining is known as Hybrid-NER (hNER)[3]. They propose an NER system for biomedical entities by incorporating n-grams with bi-directional long short-term memory (BiLSTM) and CRF .It is called a contextual long short-term memory network with CRF (CLSTM).The CLSTM was compared with several deep learning approaches, such as BiLSTM, BiLSTM with CRF, GRAM-CNN, and BERT[12].

Another Natural Language Processing technique useful for classifying the unstructured text is text summarization and existing work done on pre-trained text summarization include methodologies like hey implemented BERTSUM using Py Torch, Open MT, and the 'Bert-base-uncased 2 version of BERT for both extractive and abstractive settings. BERT's sub-word tokenizer was used to tokenize both the source and target texts. On three GPUs, the extractive models were trained for 50,000 steps. Every 1,000 steps, model checkpoints were saved and evaluated on the validation set. Abstractive models were trained on four GPUs for 200,000 steps, with gradient accumulation every five steps. Every 2,500 steps, model checkpoints were saved and evaluated on the validation set[4]. Another similar work using PEGASUS for abstractive summarization follows testing the proposed pre-training objective on a variety of downstream summarization tasks, using careful ablations to select the best model settings, which we then use to train a 568M parameter PEGASUS model that outperforms or is on par with the state-of-the-art on all 12 downstream datasets considered. They demonstrate how, by fine-tuning the PEGASUS model, they can achieve good abstractive summarization performance across broad domains with little supervision, outperforming previous state-of-the-art results on many tasks with as few as 1000 examples[5].

Summarization can also be helpful in creating meaningful datasets which can be further used to train our NER model. A study in this particular field does this by compressing the content in large document collections into short summaries and providing a quantitative analysis of the dataset and empirical results for several state-of-the-art MDS techniques[11].

Combining multiple Natural Language techniques can provide very accurate or inadequate results based on the conditions of the data present and collected, type of information and compatibility between different NLP models. This study shows a model built by combining text summarization and Named Entity Recognition where they propose a pipeline where the texts are summarized before the analysis begins. With this, the source articles are reduced significantly, to a compact version which contains only the most encountered entities. They show that by reducing the text size, they get knowledge extraction results comparable to the full text analysis approach and, at the same time, they significantly reduce the processing time, which is essential for getting both real-time results on single text sources, and faster results when analyzing entire batches of collected articles from the domain[10].

Web Scraping is a very useful Natural Language Processing technique where any kind of information can be gathered from the internet. The type of the software or technology used for scraping information may vary depending on the type of website and software. For example this study shows using a software called twint for scraping information from Twitter using the python script[8]. Another study shows scraping information at a scale from cloud based applications where they address both scraping and feasibility for big data applications in a single cloud-based architecture for data-based industries and analyze the scalability and performance of the proposed cloud-based scrapper[9].

Using a NoSQL database for storing unstructured data or when there is no specified schema in mind is always better than using generic SQL databases. Existing study shows that MongoDB is also good at handling data at a scale in big data applications, for analysis case-study application was created using the document-based MongoDB and MySQL databases, with the goal of modeling and streamlining the activity of service providers who use a large amount of data. Performance tests were done on both MySQL and mongo dB databases on operations create, insert, delete, update and one or more specific relevant operations[6].In the architecture, to insert patient records into the MongoDB database, the user must first log in to the HR application. To store sensitive data, users can encrypt it with the RSA algorithm and then insert it. Individual patient details are kept up to date in order to perform analysis for early disease detection or to discover knowledge based on query retrieval[7].

2.2. SUMMARY/GAPS IDENTIFIED IN STUDY

BioBERT (Bidirectional Encoder Representations from Transformers for Biomedical Text Mining) is a domain-specific language representation model that has been pre-trained on large-scale biomedical corpora. In a variety of biomedical text mining tasks, it significantly outperforms BERT and previous state-of-the-art models[1]. Medical records are important resources that document patients' diagnosis and treatment activities in hospitals. This paper presents a medical NER approach for recognizing medical named entities that incorporates a medical domain dictionary and pre-trained language models and focuses on how to extract knowledge from unlabeled medical texts[2]. In this article, a new hybrid-based approach for identifying named entities from medical literature documents is proposed. To annotate the entities in medical documents, a new dictionary for route of administration, dosage forms, and symptoms has been created. The blank Spacy machine learning model is used to train the annotated entities. Comparing the trained model to the existing model demonstrates decent accuracy[3]. They investigated neural architectures with contextual information for biomedical named entity recognition based on various corpora and word embeddings. The experimental results show that the system outperforms several other NER approaches and exhibits similar performance to the transfer learning approach. The results of this study will help to make biomedical text mining more accurate and more robust irrespective of the entity type[12]. They demonstrated how pretrained BERT can be useful in text summarization. They proposed a general framework for both abstractive and extractive summarization as well as a novel document-level encoder. Experiment results across three datasets show that their model consistently achieves good results under both automatic and human-based evaluation protocols[4].

The existing study proposes a new self-supervised pre-training goal for abstractive summarization, the generation of gap sentences, and research strategies for selecting those sentences. They conducted human evaluation studies on XSum, CNN/Daily Mail, and Reddit TIFU to validate their experimental design and demonstrate human-level summarization performance[5]. They present a new large-scale MDS dataset for the news domain, consisting of large clusters of news articles, associated with short summaries about news events. They build this dataset by leveraging the Wikipedia Current Events Portal (WCEP), which provides concise and neutral human-written summaries of news events, with links to external source articles[11].

Analyzing long text articles in the pharmaceutical domain, for the purpose of knowledge extraction and recognizing entities of interest[10]. The disease information collected from Twitter are Demam Berdarah Dengue (DBD), malaria disease, Antraks Disease, Canine Madness (Anjing Gila), Bird Flu (flu burung), and Ebola Disease. They use the twint to get the data of disease information, related tweets and twitter user details. Data downloaded from the Twitter server in the form of users and tweets along with their attributes[8]. They use selenium for web scraping because of web drivers it supports which simulates a real user working with a browser. They described the advantages of the proposed cloud-based scraping over other cloud-based scrapers and discussed advantages of proposed architecture and briefly compared it with traditional/other web scraping architecture[9].

The primary goal of this paper is to conduct a comparative analysis of the impact that each database has on application performance when performing CRUD operations. The obtained results demonstrate the performance of both databases for different data volumes; based on these, a detailed analysis and several conclusions were presented to support a decision for selecting an appropriate solution for use in a big-data application[6]. One of the Big Data applications is the Healthcare Record (HR), which contains valuable information entered by clinicians. HR data includes data generated by body vitals such as blood pressure, body temperature, lab reports, and prescriptions, which are stored in MongoDB. Before inserting sensitive information into a database, the user should explicitly encrypt it using RSA Algorithms[7].

To conclude, there are a very minimal number of knowledge bases/ online database websites in the disease domain and combining more than one natural language processing technique can be beneficial. MongoDB provides best performance in handling data at a scale among other NoSQL databases and we would need to use different web scraping techniques for different types of websites from which the data needs to be scrapped. Although BERT provides best results in NER classification, Sci-Spacy proves better than Bio-BERT in handling biomedical data in terms of accuracy of classification and the variety of entities present in Spacy.

3. OVERVIEW OF THE PROPOSED SYSTEM

3.1. INTRODUCTION AND RELATED CONCEPTS

We used Natural Language processing techniques like Web Scraping, Named Entity recognition and Text Summarization to gather information for building the knowledge base. We started by breaking down the required entities and finding accurate sources for each of these entities and then used the three mentioned techniques according to the need and type of data.

Web Scraping:

Web Scraping or Web Extraction is a method or software used to retrieve data from the internet through the Hypertext Transfer protocol. This can be extracting data from a static storage or even from a dynamic website using a crawler to simulate the actions of an end-user. We used both these techniques to gather our information.

Web Scraping is done using Python and we used libraries like BeautifulSoup, requests and selenium for the same. For static websites, we can use the requests library to access the URL of the website and then use BeautifulSoup which is a python package mainly used for parsing HTML or XML webpages. In the case of a dynamic website where the information is only available after a certain action done by the user, we use selenium which is a browser automation package available in python to simulate the actions and then use the parser to extract the information.

Named Entity Recognition:

NER is a popular Natural Language processing technique for information extraction where it classifies unstructured text into named entities. We used Sci-spacy, a pre-trained python package of a spaCy model for processing bio-medical data. SpaCy is an open-source software library for advanced natural language processing.

Sci-spacy has a certain number of bio-related entities that the extracted data will be classified into. We used abstracts of research journals from sources used by scopus as the input for the sci-spacy model and used the resultant entities to gather information regarding pathological formation, tissues or organs etc. for these diseases.

Text Summarization:

Text summarization is a process of shortening extracted information while retaining its original meaning and showing the most important or relevant parts of the actual text. We used text summarization to gather information of symptoms and related entities to eliminate any trivial information in order to increase the efficiency in storing the extracted data and also in displaying the useful information for the end-users.

We used the gensim.summarization library present in Python which uses a TextRank algorithm to summarize the text sentences. A ratio can be given to the summarizer to determine what percent of the actual text should be the length of the result.

Finally, we store the extracted information into MongoDB divided according to their respective entities. Nodejs and Express were used as a backend web server to retrieve and perform CRUD operations on the data while HTML, CSS and JavaScript were used to build the frontend website.

3.2. ARCHITECTURE FOR THE PROPOSED SYSTEM

The architecture for this project includes three major steps, (i) Information Gathering, (ii) Database and (iii) Front end Website.

The information gathering process is done using three Natural Language Processing techniques namely Web Scraping, Named Entity Recognition and Text Summarization simultaneously from different trusted sources providing information about various entities about diseases like Symptoms, gene information, drug information etc. These three techniques gather information independently and the gathered information is directly deployed in the database using pymongo library in Python. For Web Scraping the frameworks used were Selenium and BeautifulSoup and for text summarization we used the gensim.summarization library present in Python. For NER, we used a pre-trained model named Sci-spacy which is designed to process biomedical data.

A NoSQL database(MongoDB) was used to handle schemaless information that is being directed into the database. NodeJS and Express were used as the web server connecting the database to our front-end website which was built using HTML, CSS and JavaScript along with Bootstrap. This website can be accessed through any web browser that supports HTML5 and ES3.

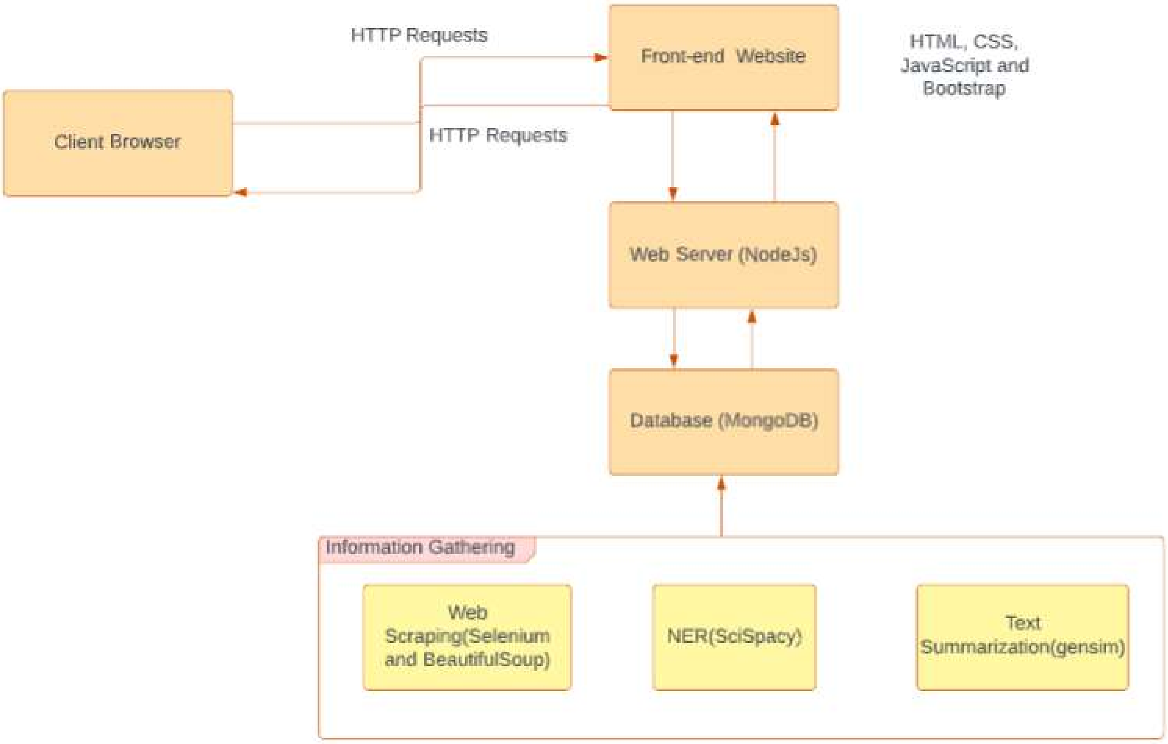


***Figure No. 3.0** Architecture Diagram with frameworks used*

3.3. PROPOSED SYSTEM MODEL

Class Diagram

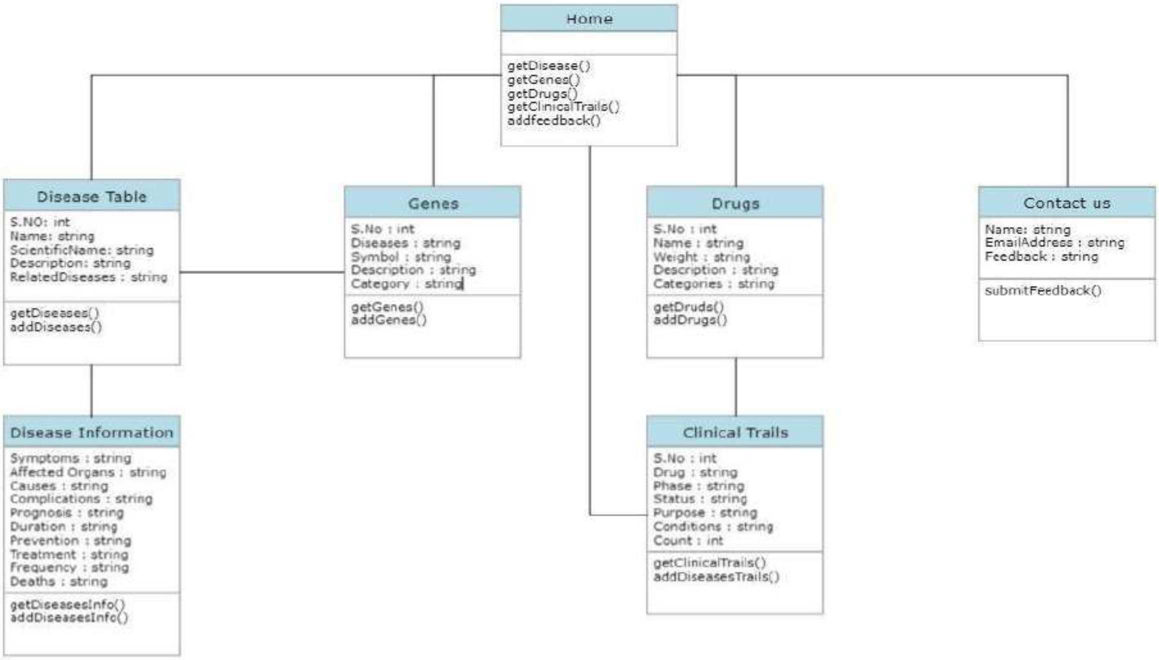


***Figure No. 3.1** Class Diagram*

Activity Diagram

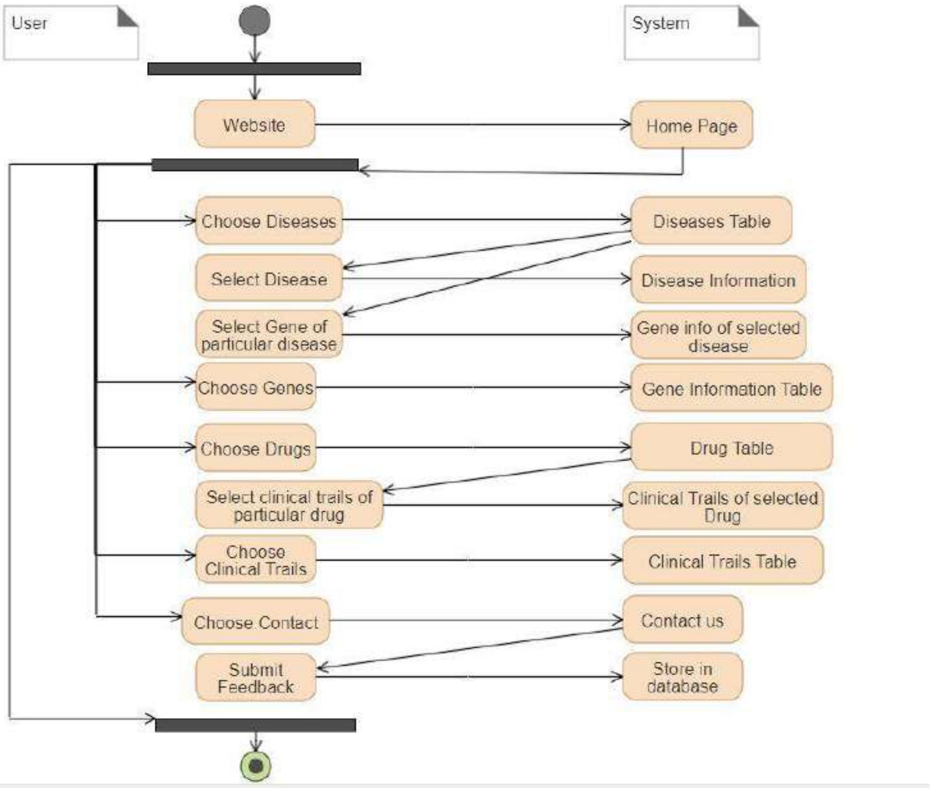


Figure No. 3.2 *Activity Diagram*

Use-Case Diagram

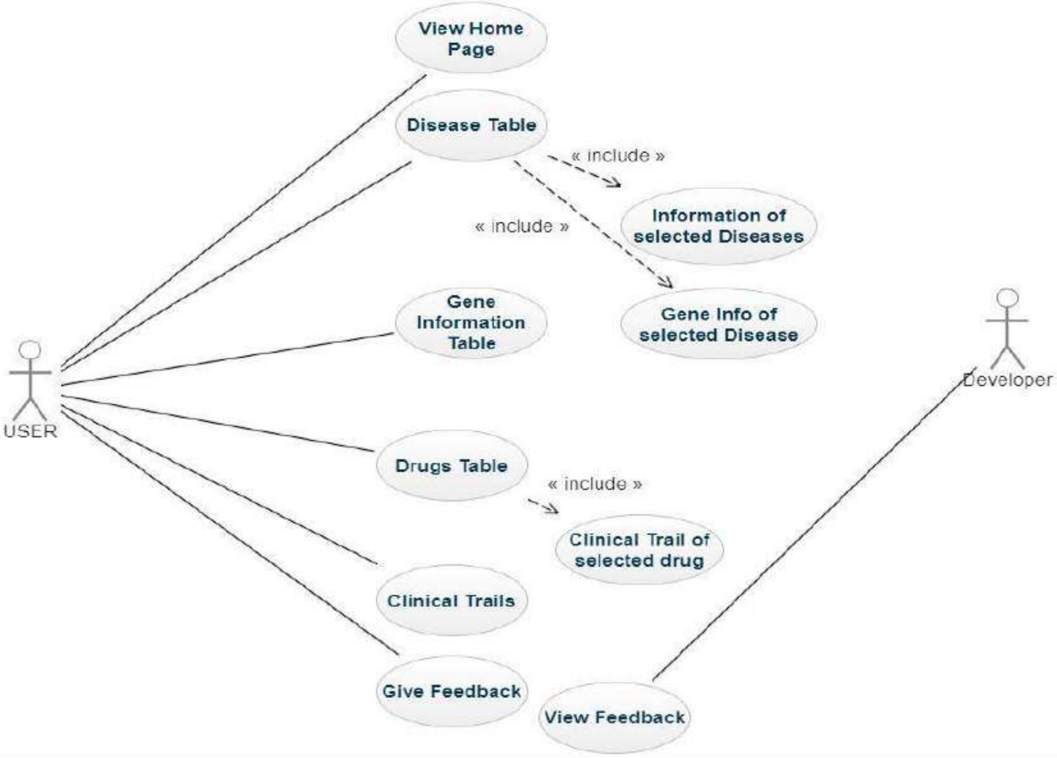


Figure No. 3.3 *Use Case Diagram*

Sequence Diagram

A) To get Disease Information

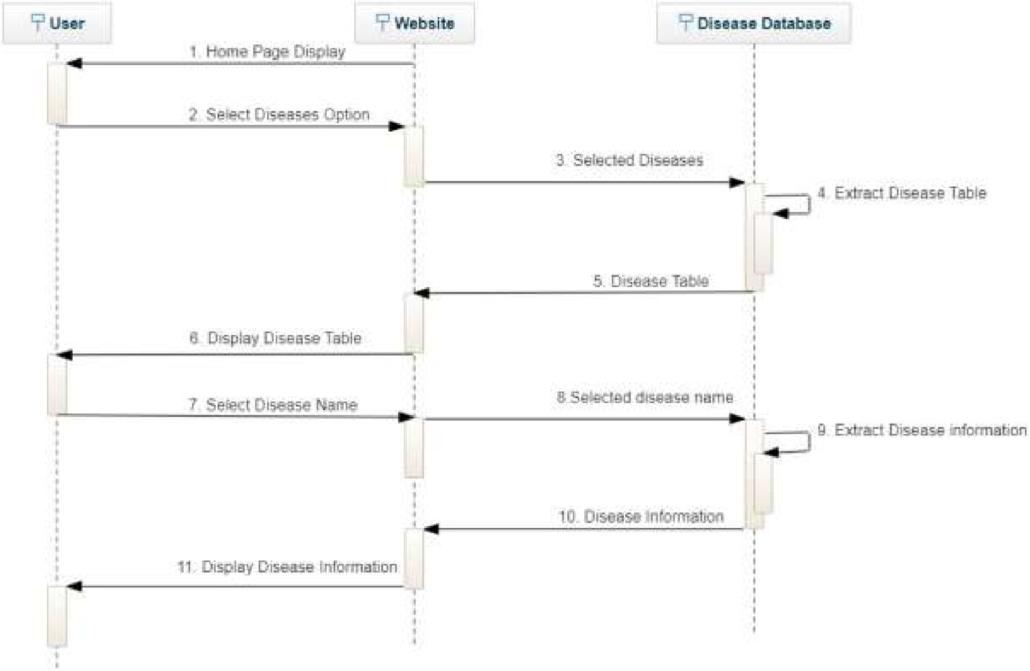


Figure No. 3.4.1 *Sequence Diagram for disease information*

B) To get Clinical Trials of selected Drugs

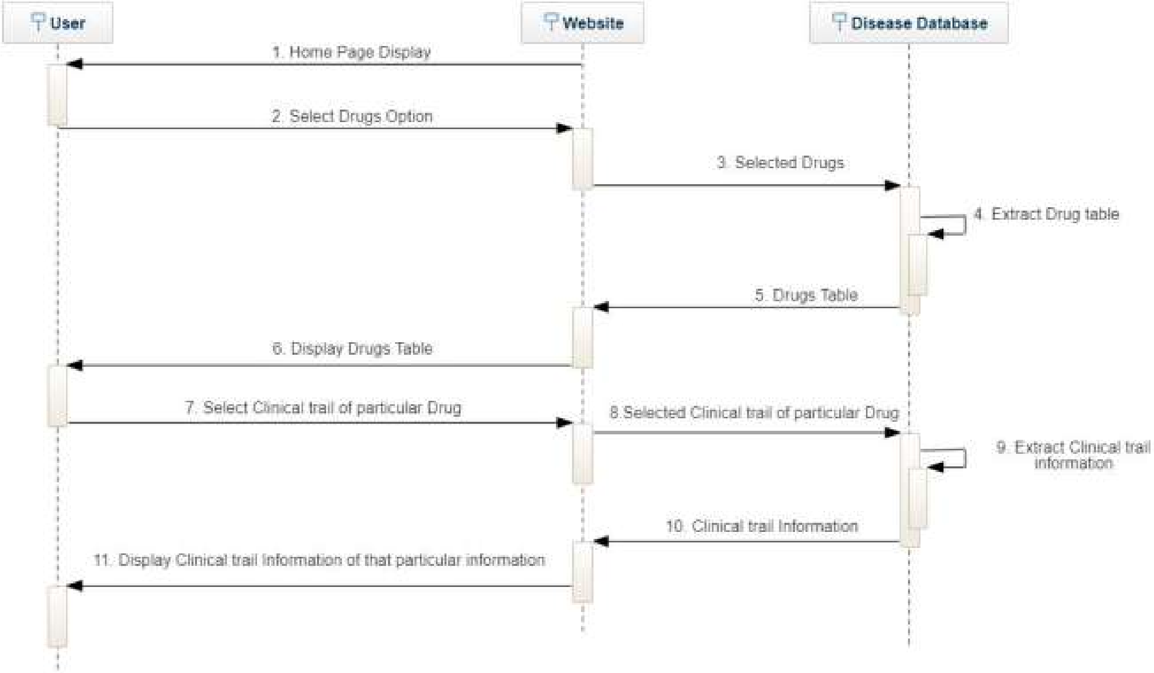

User
Website
Disease Database
1. Home Page Display
2. Select Drugs Option
3. Selected Drugs
4. Extract Drug table
5. Drugs Table
6. Display Drugs Table
7. Select Clinical trail of particular Drug
8.Selected Clinical trail of particular Drug
9. Extract Clinical trail information
10. Clinical trail Information
11. Display Clinical trail Information of that particular information

Figure No. 3.4.2 *Sequence Diagram for Clinical trials*

C) To get Gene Information Table

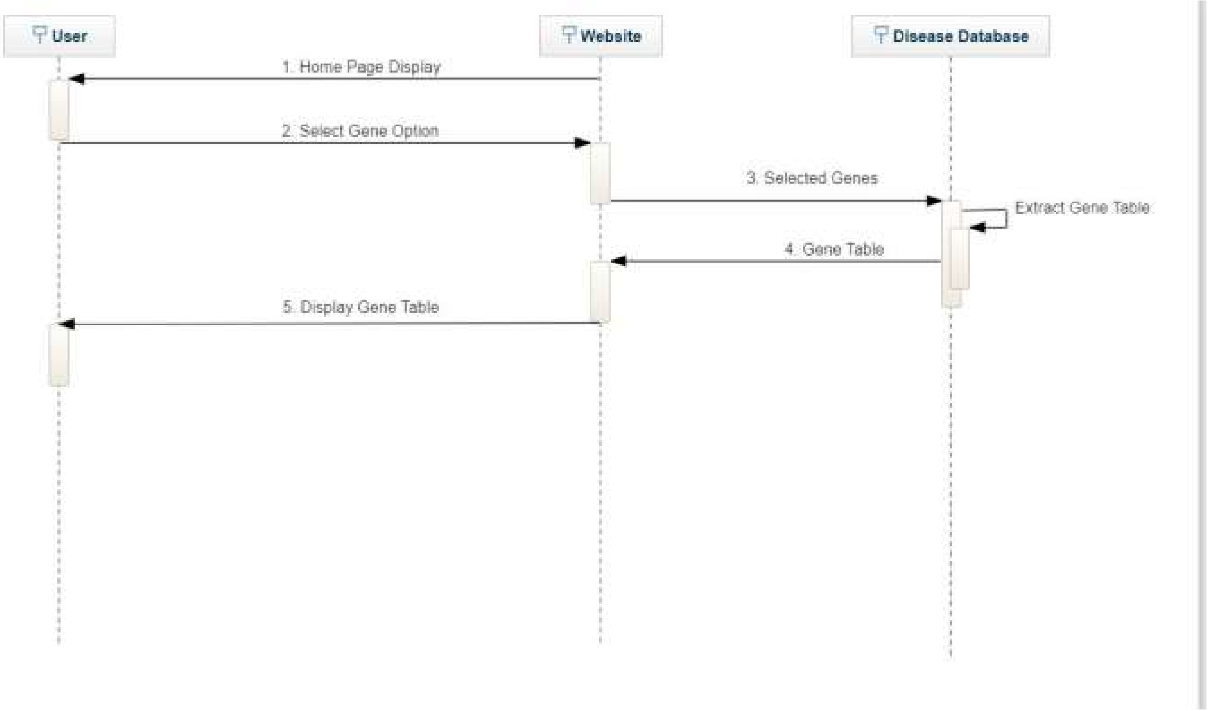


Figure No. 3.4.3 *Sequence Diagram for gene information*

Collaboration Diagram

A) To get Disease information

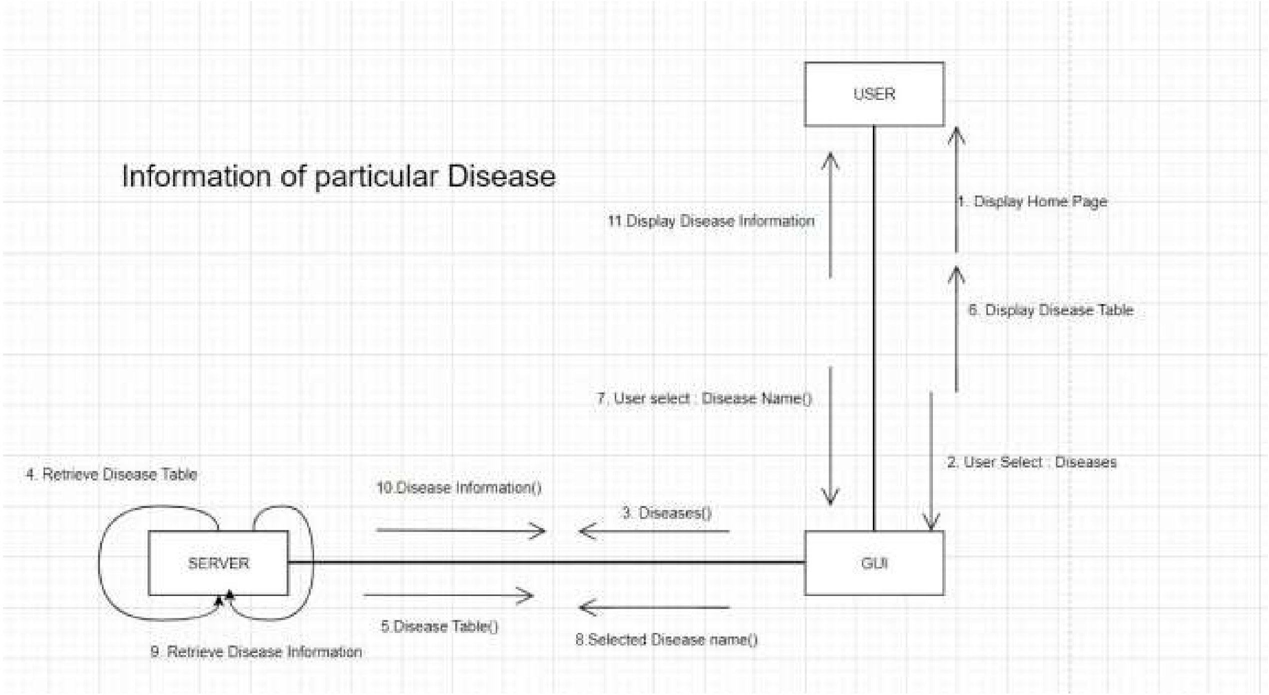

Information of particular Disease
USER
GUI
SERVER
1. Display Home Page
2. User Select : Diseases
3. Diseases()
4. Retrieve Disease Table
5.Disease Table()
6. Display Disease Table
7. User select : Disease Name()
8.Selected Disease name()
9. Retrieve Disease Information
10.Disease Information()
11.Display Disease Information

Figure No. 3.5.1 *Collaboration Diagram for disease information*

B) To get clinical trials of selected drugs

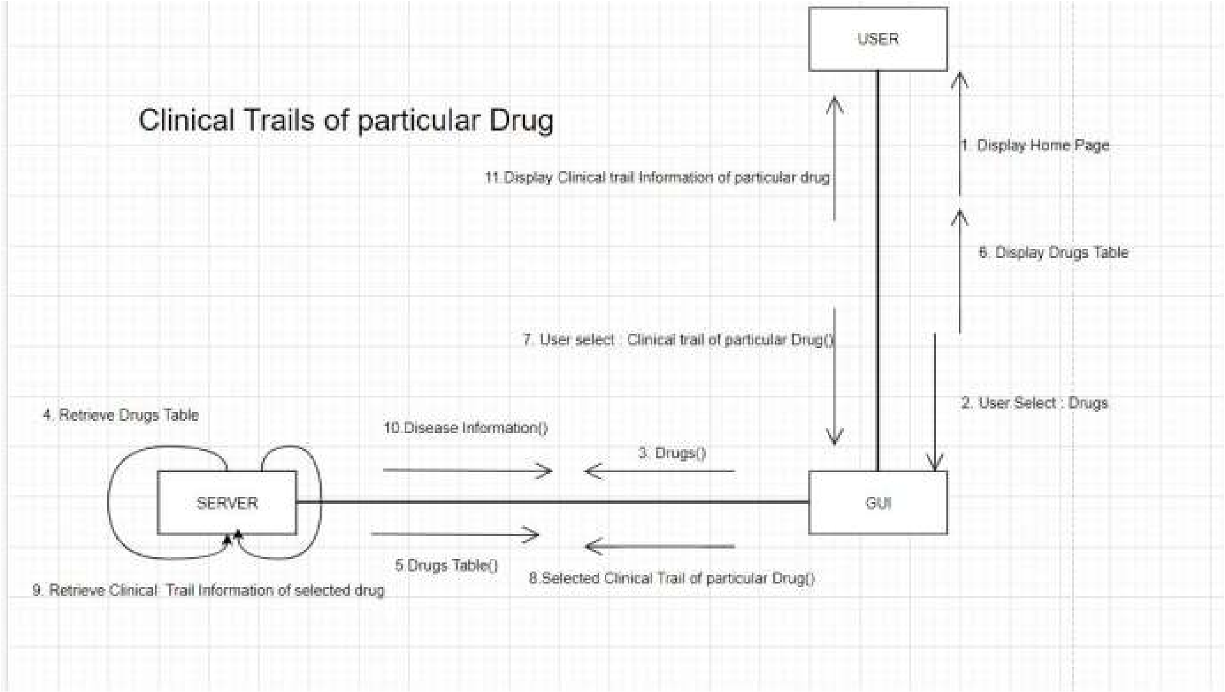

Clinical Trails of particular Drug
USER
GUI
SERVER
1. Display Home Page
2. User Select : Drugs
3. Drugs()
4. Retrieve Drugs Table
5.Drugs Table()
6. Display Drugs Table
7. User select : Clinical trail of particular Drug()
8.Selected Clinical Trail of particular Drug()
9. Retrieve Clinical Trail Information of selected drug
10.Disease Information()
11.Display Clinical trail Information of particular drug

Figure No. 3.5.2 *Collaboration diagram for clinical trials*

C) To get Gene Information Table

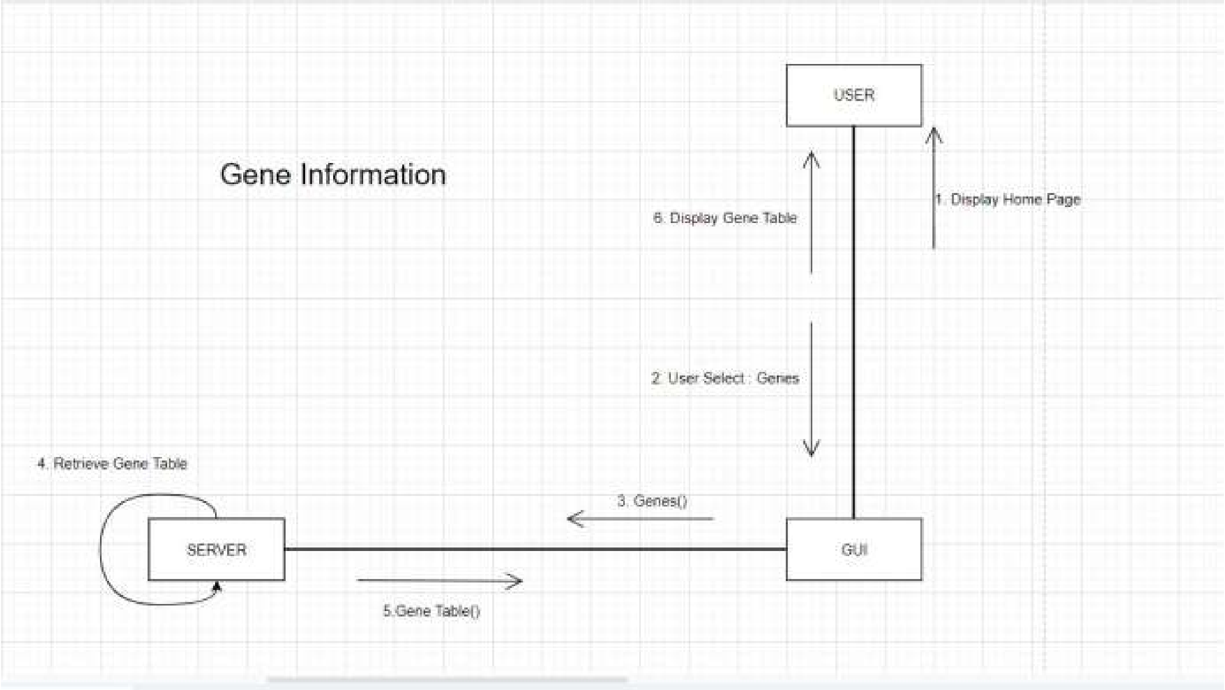


Figure No. 3.5.3 *Collaboration diagram for Gene information*

System Design

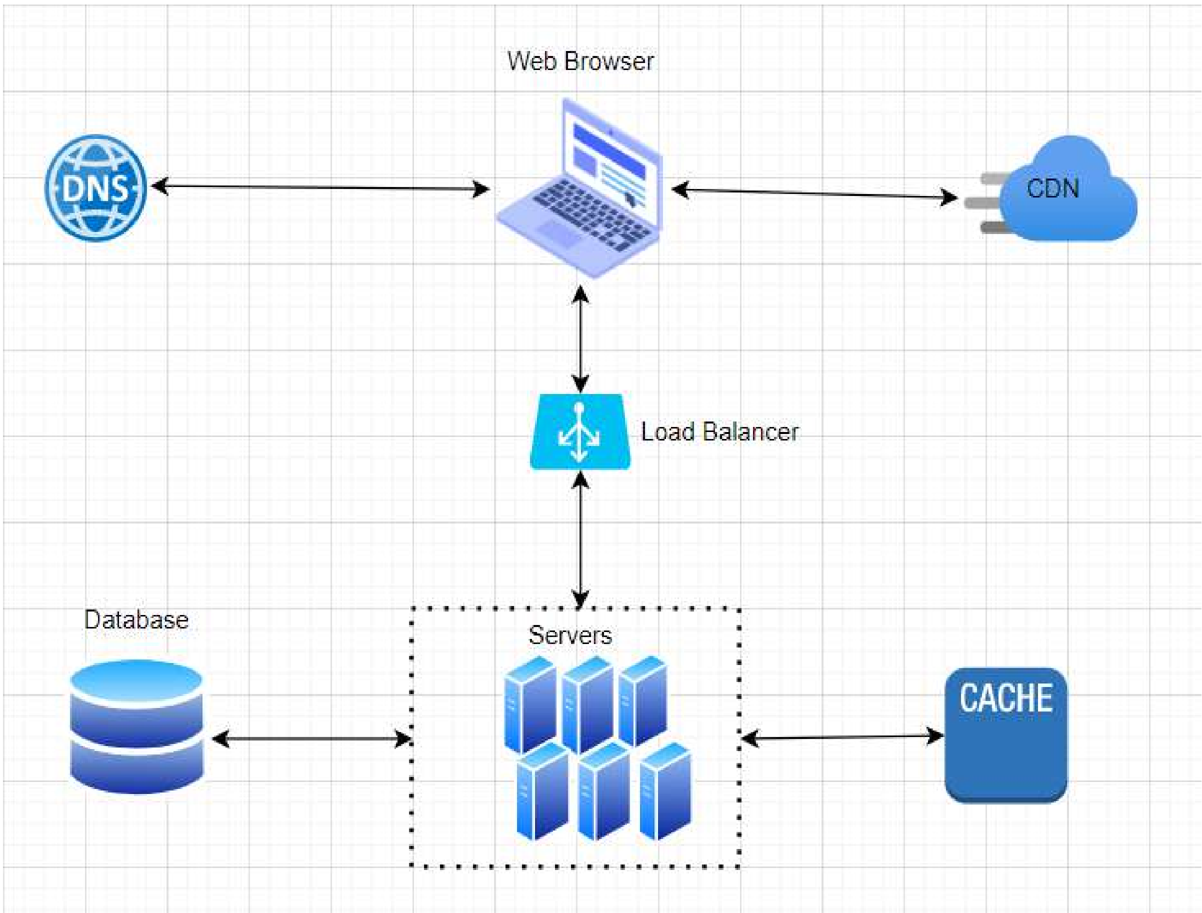


Figure No. 3.6 *System Design*

When the user enters the name of the domain in the web browser, the query is sent to Domain Name System (DNS) which sends back the ip address of the website. Now our request is redirected to one of the web servers and a load balancer is used to maintain the traffic across a number of servers. The server processes the user's request and sends the required data back to the browser. It also refers to back-end infrastructures such as database and cache. Database stores our data and can be used to perform create, read, update, delete operations on the data. Cache is used to store previous searched data to make computations faster which results in performing quick read or search operations. Content delivery system (CDN) sends the HTML/CSS/Javascript files to the web browser. It works by caching content close to where each end user is accessing the internet and distributes content from the end server around the world.

4. PROPOSED SYSTEM ANALYSIS AND DESIGN

4.1. INTRODUCTION

Background: Disease databases are a crucial component of the healthcare system into which healthcare professionals record fundamental clinical information as well as some sophisticated information on diseases and their gene encodings.

With the aid of these knowledge bases, data from various sources about a specific disease can be brought together, enhancing the accuracy and speed of future research and study into these conditions, whether it be to evaluate the effects of a new drug or to gauge a disease's reactions and symptoms to comprehend how it develops.

Building this with the help of Natural Language processing techniques like Named Entity Recognition, Text Summarization and Web Scraping and then displaying the collected information in the form of a website while storing the data in a NoSQL database is the aim of this product.

Purpose: The purpose of this document is to present a detailed description of the Disease Knowledge Base. It will explain the purpose and features of the system, the interfaces of the system, what the system will do, the constraints under which it must operate and how the system will react to the actions of the end-users. This document is intended for both the stakeholders and the developers of the system.

Scope: This product helps provide information about all major entities of a particular domain in a single place. With a very user-friendly interface, this is both useful to the public and to researchers as the product provides both basic and advanced information. Medical domain experts can validate the data and this data can be used to train further Machine Learning models.

References: The product uses various sources, which include trusted medical journals from the Scopus website. It uses articles and information from existing disease database websites to gather the required information about different entities related to diseases.

Document Organization: This document provides the information regarding the requirement analysis done for the Disease Knowledge Base website, functional requirements regarding the product, users and the domain were discussed first followed by the non-functional requirements like product, organization and operational requirements were discussed.

4.2. REQUIREMENT ANALYSIS

4.2.1. FUNCTIONAL REQUIREMENTS

4.2.1.1. PRODUCT PERSPECTIVE

The disease online database provides an alternative to existing Electronic Health Records and manual records maintained by various pharmacies and hospitals. The website provides information that is useful for a vast range of users through a simple and open to all user interfaces. Data is collected using efficient Natural Language Processing Techniques and is stored in a NoSQL database like MongoDB to eradicate the issue of irregular schema. More entities in the diseases' domain can be easily added and the interface can be replicated to be used for other domains as well.

4.2.1.2. PRODUCT FEATURES

- Home Page: The page provides information about us and all the resources that were used to gather the data that is presented on the website. This can be used by users who want to gain further information regarding any specific entity.

- Diseases Page: This page provides all the list of available diseases' information present on the website currently with some basic summarized description along with links to move to general information and gene information.

- General Information Page: The general information page provides information about entities like symptoms, organs affected, Prognosis, Prevention methods, complication etc.

- Gene Information Page: This page provides information about the GENEs related to the disease like the gene code, the category it belongs to and a simple description.

- Drugs Information Page: This page provides information about the drugs used for the diseases like the name of the drug, weight, category and any clinical trials done on them etc.

- Clinical Trials Page: This page provides information about the clinical trials done on each drug presented in the Drugs Information page if any are done.

- Contact Us Page: This page focuses mainly on gathering feedback on any issue faced by the end user to the development team. This can include any misinformation presented on the website or any addition of new features to the website.

- Cross Platform: The website can be accessed from both PC and mobile devices on any major Operating system.

4.2.1.3. USER CHARACTERISTICS

- The users of this product can vary from having little to no knowledge about diseases and having some knowledge about gene encoding, drugs etc, to gain the maximum experience from the website.
- The end users can be bio-medical students, the public who want to gain some basic knowledge about contagious diseases or even domain experts who can add more knowledge or correct information on the website by contacting the development team.

4.2.1.4. ASSUMPTION AND DEPENDENCIES

The product requires the user to have a web browser that can support HTML5, ES3 and Bootstrap and it is assumed that the user will have the required internet connectivity.
Since the data is stored in MongoDB atlas, a cloud data storage, the website depends on the server UP time of atlas for it to display the information.

4.2.1.5. DOMAIN REQUIREMENTS

- Access to reliable and up-to-date medical data sources: The website would need to access and retrieve data from reliable sources such as medical journals and current EHRs.

- Data management skills: The website would need to organize and manage large amounts of medical data in a way that is easy to search, filter, and retrieve by users. The use of a database management system, optimally any NoSQL database would be required.

- User interface design: The website would need to have a user-friendly interface that allows users to navigate and search for information easily. The interface should be accessible by users with varying knowledge regarding the subject.

4.2.1.6. USER REQUIREMENTS

Since the product is a website, the users are expected to have some basic knowledge of the operating system that they are working on along with some experience browsing the internet and accessing websites.

4.2.2. NON FUNCTIONAL REQUIREMENTS

4.2.2.1. PRODUCT REQUIREMENTS

4.2.2.1.1. EFFICIENCY

- The maximum time taken by webpages to load should be within 2 seconds.
- The maximum time taken to put data into the database should be less than 10 seconds.

4.2.2.1.2. RELIABILITY

- The system should allow at least 10,000 people to use it at the same time. It should not be crashed because of overload.
- The failure rate of the system must be less than 5 percent of use cases per week.

4.2.2.1.3. PORTABILITY

- The website should be able to run on all kinds of Web Browsers.
- The website should have a responsive design to function properly on different devices.

4.2.2.1.4. USABILITY

- The website should be self-explanatory and intuitive, such that the user should be able to familiarize themselves with it within 5 minutes of opening it for the first time.
- The system shall be usable by program developers within 4 weeks of training time.

4.2.2.2. ORGANIZATIONAL REQUIREMENTS

4.2.2.2.1. IMPLEMENTATION REQUIREMENTS

- The system should outline the compatible database management system, include instructions for setting up and customizing the database structure, and list the required access permissions.
- The software system should include instructions and methods to set up for disaster recovery techniques, such as backup plans, data replication.
- During deployment, the software system should enable rollback and version management procedures, allowing you to revert to a previous version if necessary.
- The system should specify the testing protocols and requirements, including

integration testing, and compatibility testing on the target environment, for evaluating the deployment process.

4.2.2.2.2. ENGINEERING STANDARD REQUIREMENTS

- The system must adhere to established security standards and information security management best practices. This comprises issues like data encryption and safe authentication methods.

- The system must follow the documentation standards that specify the framework, style, and content of different software documentation items such as user guides, API references, and release notes.

- The system should follow a standardized software development lifecycle model to ensure a planned and controlled development process.

4.2.2.3. OPERATIONAL REQUIREMENTS

- The system must follow all applicable health and safety laws to protect users' health and safety and to guarantee that the system complies with industry-specific standards.

- If third-party libraries or components are utilized, the right licenses and permissions must be secured so as not to violate anybody else's intellectual property.

- The software system should include proper legal paperwork, such as End-User License Agreements (EULAs), that clearly describe the rights and duties of users and the software developer.

- The software system needs to produce thorough audit trails and logs of user actions, system events, and error situations. These logs offer a record of system behavior and make inspection and troubleshooting easier.

4.2.3. SYSTEM REQUIREMENTS

4.2.3.1. HARDWARE REQUIREMENTS

A PC or a mobile device with 4GB of RAM and running on the latest version of any of the major operating systems (Windows, Linux, Mac, Android etc.)

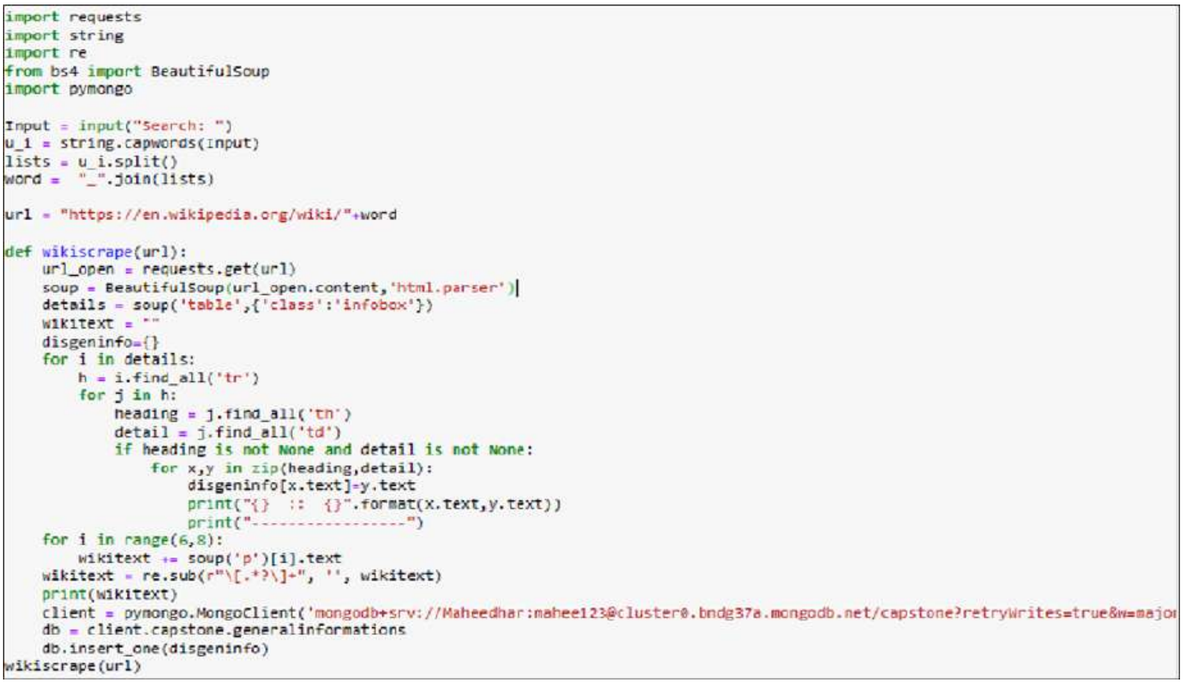
The device should be able to run any web browser that can support HTML5, ES3.

4.2.3.2. SOFTWARE REQUIREMENTS

Jupyter Notebook
MongoDB
Python libraries (pandas,selenium,BeautifulSoup,requests,pymongo, gensim,spacy)
HTML
CSS
JavaScript
Nodejs
Express
Web Browser that supports HTML5 and ES3.

5. RESULTS AND DISCUSSION

General Information

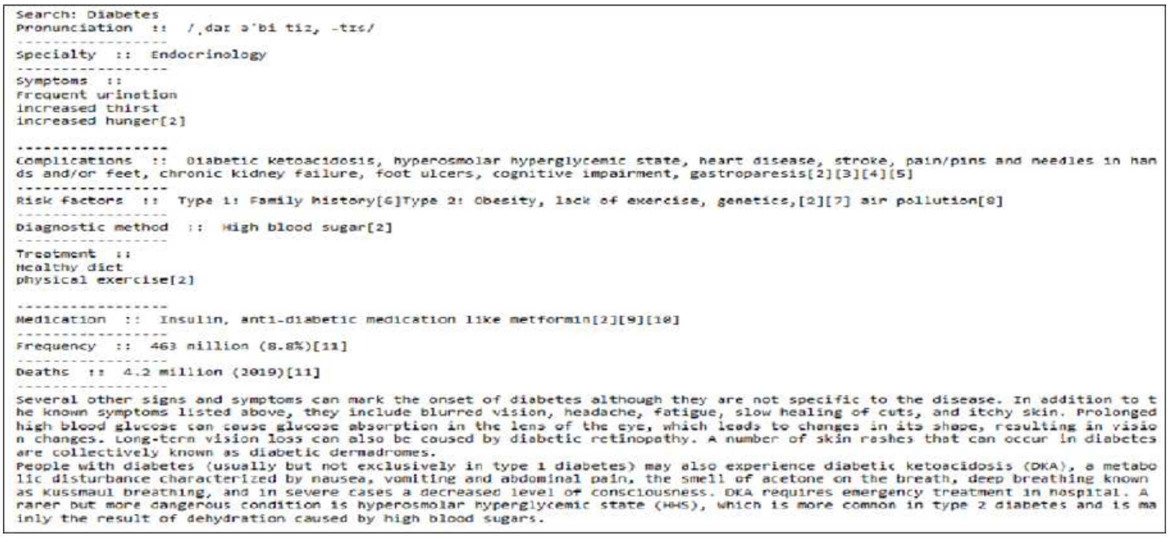


```
import requests
import string
import re
from bs4 import BeautifulSoup
import pymongo

Input = input("Search: ")
u_i = string.capwords(Input)
lists = u_i.split()
word =  "_".join(lists)

url = "https://en.wikipedia.org/wiki/"+word

def wikiscrape(url):
    url_open = requests.get(url)
    soup = BeautifulSoup(url_open.content,'html.parser')
    details = soup('table',{'class':'infobox'})
    wikitext = ""
    disgeninfo={}
    for i in details:
        h = i.find_all('tr')
        for j in h:
            heading = j.find_all('th')
            detail = j.find_all('td')
            if heading is not None and detail is not None:
                for x,y in zip(heading,detail):
                    disgeninfo[x.text]=y.text
                    print("{}  ::  {}".format(x.text,y.text))
                    print("-----------------")
    for i in range(6,8):
        wikitext += soup('p')[i].text
    wikitext = re.sub(r"\[.*?\]+", '', wikitext)
    print(wikitext)
    client = pymongo.MongoClient('mongodb+srv://Maheedhar:mahee123@cluster0.bndg37a.mongodb.net/capstone?retryWrites=true&w=major
    db = client.capstone.generalinformations
    db.insert_one(disgeninfo)
wikiscrape(url)
```

Figure 5.0 Scraping general information about diseases

Starting with general information, we use the requests library to access the URL and then using the BeautifulSoup library we find the rows inside the table and then extract the information. To automate the process for numerous URLs, we also are taking the disease name as the input which would directly scrape information regarding that disease. By using the pymongo library we directly deploy the extracted information into our database using the Mongo connection string.

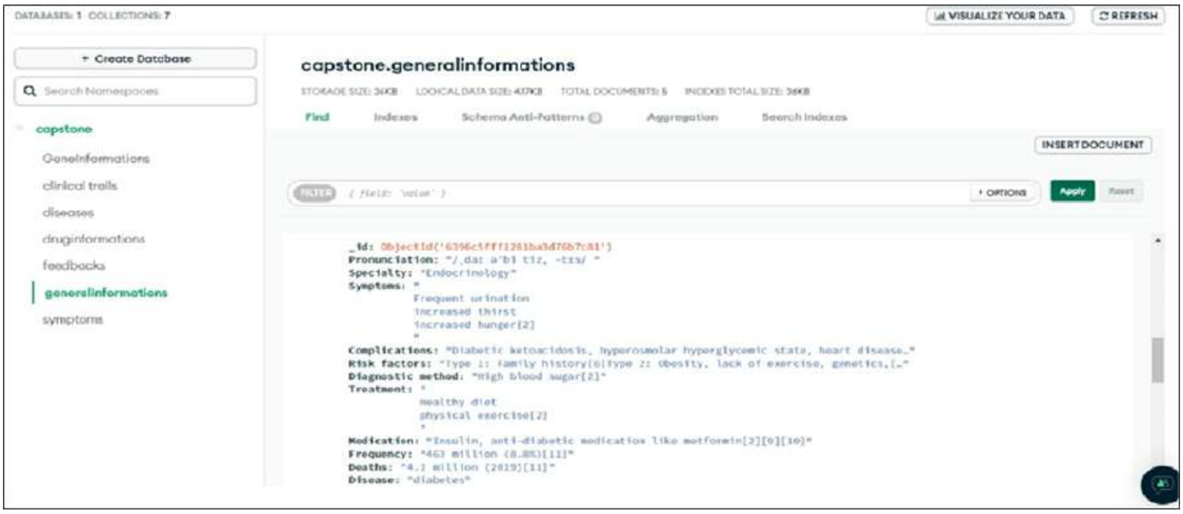
The result for one such disease information extraction can be seen below:



```
Search: Diabetes
Pronunciation  ::  /ˌdaɪ ə'bi tiz, -tɪs/
-----------------
Specialty  ::  Endocrinology
-----------------
Symptoms  ::
Frequent urination
increased thirst
increased hunger[2]

-----------------
Complications  ::  Diabetic ketoacidosis, hyperosmolar hyperglycemic state, heart disease, stroke, pain/pins and needles in han
ds and/or feet, chronic kidney failure, foot ulcers, cognitive impairment, gastroparesis[2][3][4][5]
-----------------
Risk factors  ::  Type 1: Family history[6]Type 2: Obesity, lack of exercise, genetics,[2][7] air pollution[8]
-----------------
Diagnostic method  ::  High blood sugar[2]
-----------------
Treatment  ::
Healthy diet
physical exercise[2]

-----------------
Medication  ::  Insulin, anti-diabetic medication like metformin[2][9][10]
-----------------
Frequency  ::  463 million (8.8%)[11]
-----------------
Deaths  ::  4.2 million (2019)[11]
-----------------
Several other signs and symptoms can mark the onset of diabetes although they are not specific to the disease. In addition to t
he known symptoms listed above, they include blurred vision, headache, fatigue, slow healing of cuts, and itchy skin. Prolonged
high blood glucose can cause glucose absorption in the lens of the eye, which leads to changes in its shape, resulting in visio
n changes. Long-term vision loss can also be caused by diabetic retinopathy. A number of skin rashes that can occur in diabetes
are collectively known as diabetic dermadromes.
People with diabetes (usually but not exclusively in type 1 diabetes) may also experience diabetic ketoacidosis (DKA), a metabo
lic disturbance characterized by nausea, vomiting and abdominal pain, the smell of acetone on the breath, deep breathing known
as Kussmaul breathing, and in severe cases a decreased level of consciousness. DKA requires emergency treatment in hospital. A
rarer but more dangerous condition is hyperosmolar hyperglycemic state (HHS), which is more common in type 2 diabetes and is ma
inly the result of dehydration caused by high blood sugars.
```

Figure 5.1 Resultant Scrapped general information for diabetes

Data inserted to MongoDB:

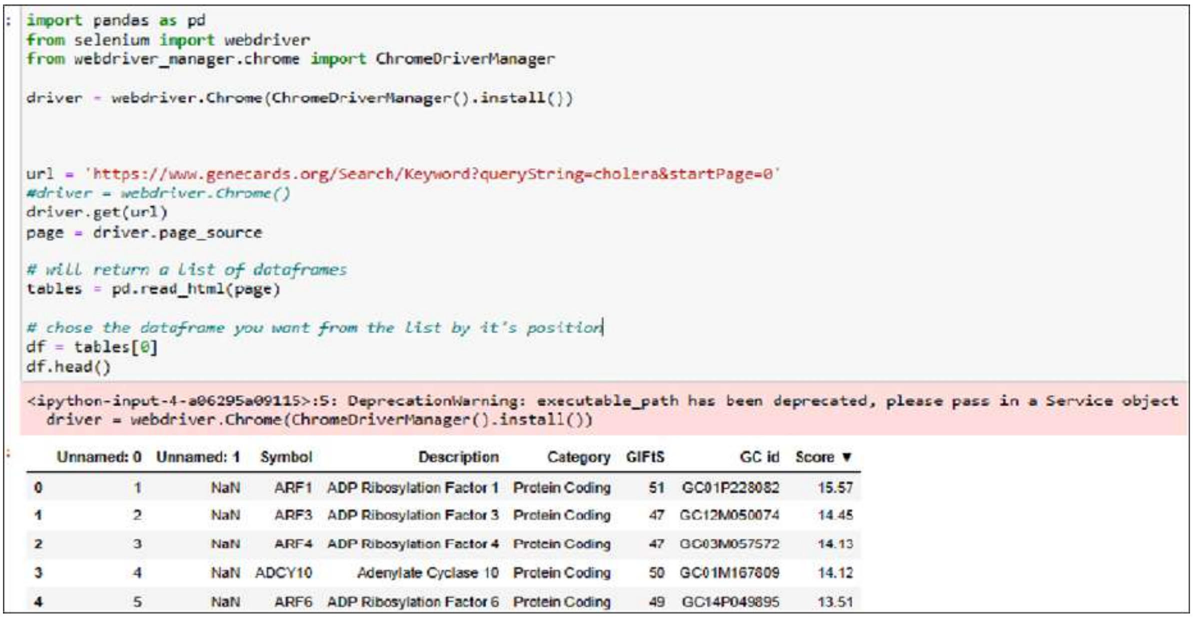


```
_id: ObjectId('6396c1fff1261ba3d76b7c81')
Pronunciation: "/ˌdaɪ əˈbiː tiːz, -tɪs/ "
Specialty: "Endocrinology"
Symptoms: "
          Frequent urination
          increased thirst
          increased hunger[2]
          "
Complications: "Diabetic ketoacidosis, hyperosmolar hyperglycemic state, heart disease…"
Risk factors: "Type 1: Family history[6]Type 2: Obesity, lack of exercise, genetics,[…"
Diagnostic method: "High blood sugar[2]"
Treatment: "
          Healthy diet
          physical exercise[2]
          "
Medication: "Insulin, anti-diabetic medication like metformin[2][9][10]"
Frequency: "463 million (8.8%)[11]"
Deaths: "4.2 million (2019)[11]"
Disease: "diabetes"
```

Figure 5.2 General Information deployed into the database

GENE Information

Unlike the case of general information, the gene information is extracted from a dynamic website and thus we need to use selenium. Using selenium, we extract the data in the form of a pandas data frame and then further convert it into a json format to insert the table into MongoDB.



```
import pandas as pd
from selenium import webdriver
from webdriver_manager.chrome import ChromeDriverManager

driver = webdriver.Chrome(ChromeDriverManager().install())


url = 'https://www.genecards.org/Search/Keyword?queryString=cholera&startPage=0'
#driver = webdriver.Chrome()
driver.get(url)
page = driver.page_source

# will return a list of dataframes
tables = pd.read_html(page)

# chose the dataframe you want from the list by it's position
df = tables[0]
df.head()
```

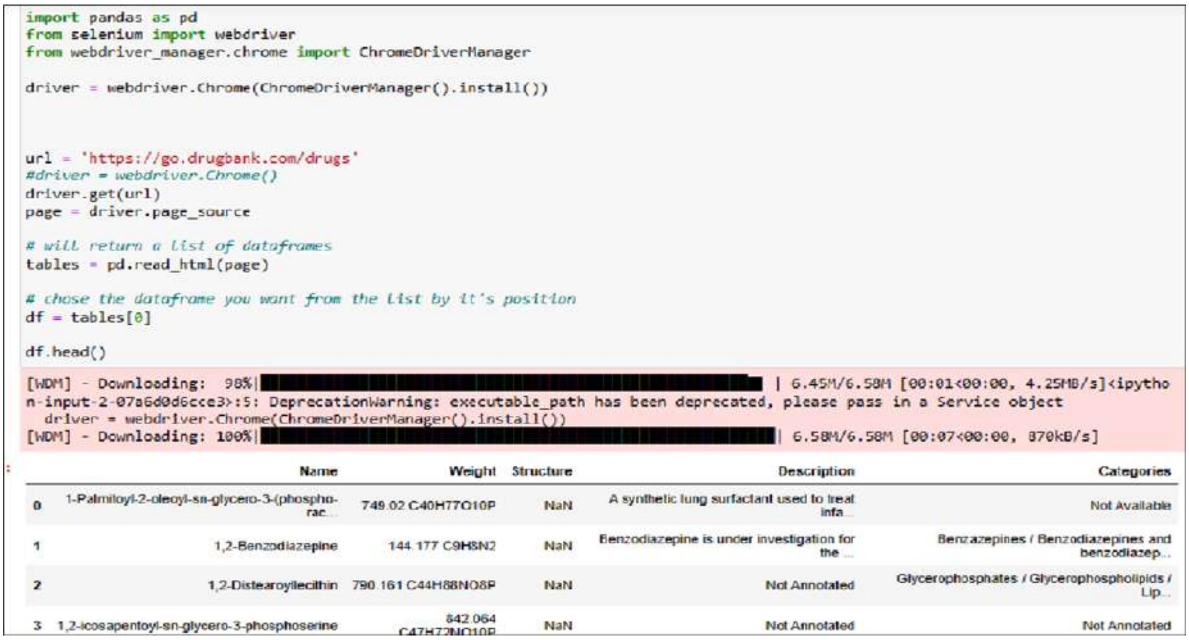
```
<ipython-input-4-a06295a09115>:5: DeprecationWarning: executable_path has been deprecated, please pass in a Service object
  driver = webdriver.Chrome(ChromeDriverManager().install())
```

```
   Unnamed: 0  Unnamed: 1  Symbol              Description     Category  GIFtS        GC id  Score ▼
0           1         NaN    ARF1  ADP Ribosylation Factor 1  Protein Coding     51  GC01P228082   15.57
1           2         NaN    ARF3  ADP Ribosylation Factor 3  Protein Coding     47  GC12M050074   14.45
2           3         NaN    ARF4  ADP Ribosylation Factor 4  Protein Coding     47  GC03M057572   14.13
3           4         NaN  ADCY10       Adenylate Cyclase 10  Protein Coding     50  GC01M167809   14.12
4           5         NaN    ARF6  ADP Ribosylation Factor 6  Protein Coding     49  GC14P049895   13.51
```

Figure 5.3 Gene Information Scrapping

Drug Information

The Drug information extraction is done in a similar way to that of gene information, we use selenium and ChromeDriver to access the website in chrome automatically and store the extracted information as a pandas data frame. While deploying the data into MongoDB, the data frame is converted into json format directly using the to_json command.



```
import pandas as pd
from selenium import webdriver
from webdriver_manager.chrome import ChromeDriverManager

driver = webdriver.Chrome(ChromeDriverManager().install())


url = 'https://go.drugbank.com/drugs'
#driver = webdriver.Chrome()
driver.get(url)
page = driver.page_source

# will return a list of dataframes
tables = pd.read_html(page)

# chose the dataframe you want from the list by it's position
df = tables[0]

df.head()
```

```
[WDM] - Downloading:  98%|████████████████████████████████████████ | 6.45M/6.58M [00:01<00:00, 4.25MB/s]<ipytho
n-input-2-07a6d0d6cce3>:5: DeprecationWarning: executable_path has been deprecated, please pass in a Service object
  driver = webdriver.Chrome(ChromeDriverManager().install())
[WDM] - Downloading: 100%|█████████████████████████████████████████| 6.58M/6.58M [00:07<00:00, 870kB/s]
```

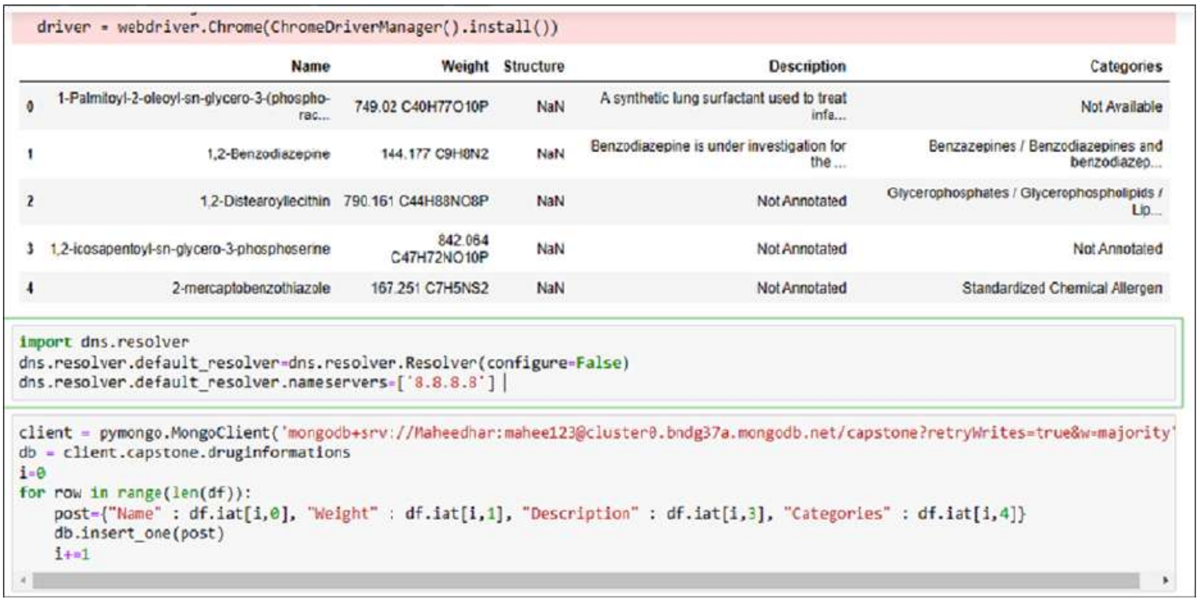
	Name	Weight	Structure	Description	Categories
0	1-Palmitoyl-2-oleoyl-sn-glycero-3-(phospho-rac...	749.02 C40H77O10P	NaN	A synthetic lung surfactant used to treat infa...	Not Available
1	1,2-Benzodiazepine	144.177 C9H8N2	NaN	Benzodiazepine is under investigation for the ...	Benzazepines / Benzodiazepines and benzodiazep...
2	1,2-Distearoyllecithin	790.161 C44H88NO8P	NaN	Not Annotated	Glycerophosphates / Glycerophospholipids / Lip...
3	1,2-icosapentoyl-sn-glycero-3-phosphoserine	842.064 C47H72NO10P	NaN	Not Annotated	Not Annotated

Figure 5.4 Drug Information Scraping

Deploying the scraped data into MongoDB:

After scraping the data as shown above, we then use a standard DNS resolver to access our Mongo client. A DNS resolver is a name server on the internet which checks if the host name is available or not and then returns either the required host ip address or other alternate name servers. We then use the pymongo library of python to access our MongoDB collection using the connection string and deploy the information into the database.

Here, a for loop is run on the data frame, and each row of the table is inserted separately as shown below:



```
driver = webdriver.Chrome(ChromeDriverManager().install())
```

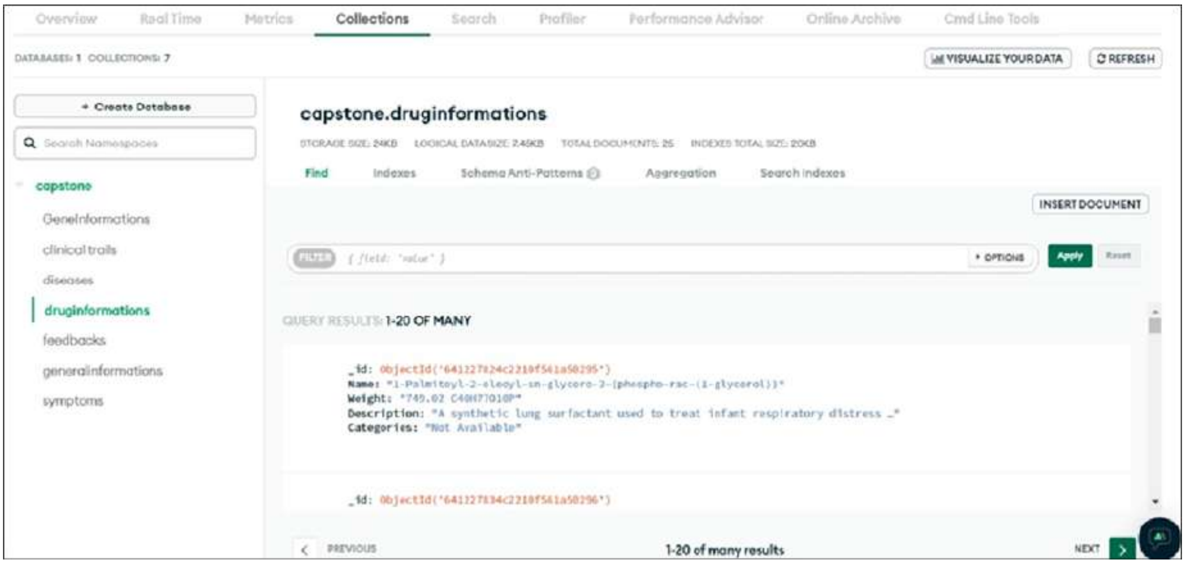
	Name	Weight	Structure	Description	Categories
0	1-Palmitoyl-2-oleoyl-sn-glycero-3-(phospho-rac...	749.02 C40H77O10P	NaN	A synthetic lung surfactant used to treat infa...	Not Available
1	1,2-Benzodiazepine	144.177 C9H8N2	NaN	Benzodiazepine is under investigation for the ...	Benzazepines / Benzodiazepines and benzodiazep...
2	1,2-Distearoyllecithin	790.161 C44H88NO8P	NaN	Not Annotated	Glycerophosphates / Glycerophospholipids / Lip...
3	1,2-icosapentoyl-sn-glycero-3-phosphoserine	842.064 C47H72NO10P	NaN	Not Annotated	Not Annotated
4	2-mercaptobenzothiazole	167.251 C7H5NS2	NaN	Not Annotated	Standardized Chemical Allergen

```
import dns.resolver
dns.resolver.default_resolver=dns.resolver.Resolver(configure=False)
dns.resolver.default_resolver.nameservers=['8.8.8.8']
```

```
client = pymongo.MongoClient('mongodb+srv://Maheedhar:mahee123@cluster0.bndg37a.mongodb.net/capstone?retryWrites=true&w=majority'
db = client.capstone.druginformations
i=0
for row in range(len(df)):
    post={"Name" : df.iat[i,0], "Weight" : df.iat[i,1], "Description" : df.iat[i,3], "Categories" : df.iat[i,4]}
    db.insert_one(post)
    i+=1
```

Figure 5.5 Data is directed to MongoDB using pymongo

Data inserted into MongoDB:



DATABASES: 1 COLLECTIONS: 7

capstone.druginformations

STORAGE SIZE: 24KB LOGICAL DATA SIZE: 2.45KB TOTAL DOCUMENTS: 25 INDEXES TOTAL SIZE: 20KB

{ field: 'value' }

QUERY RESULTS: 1-20 OF MANY

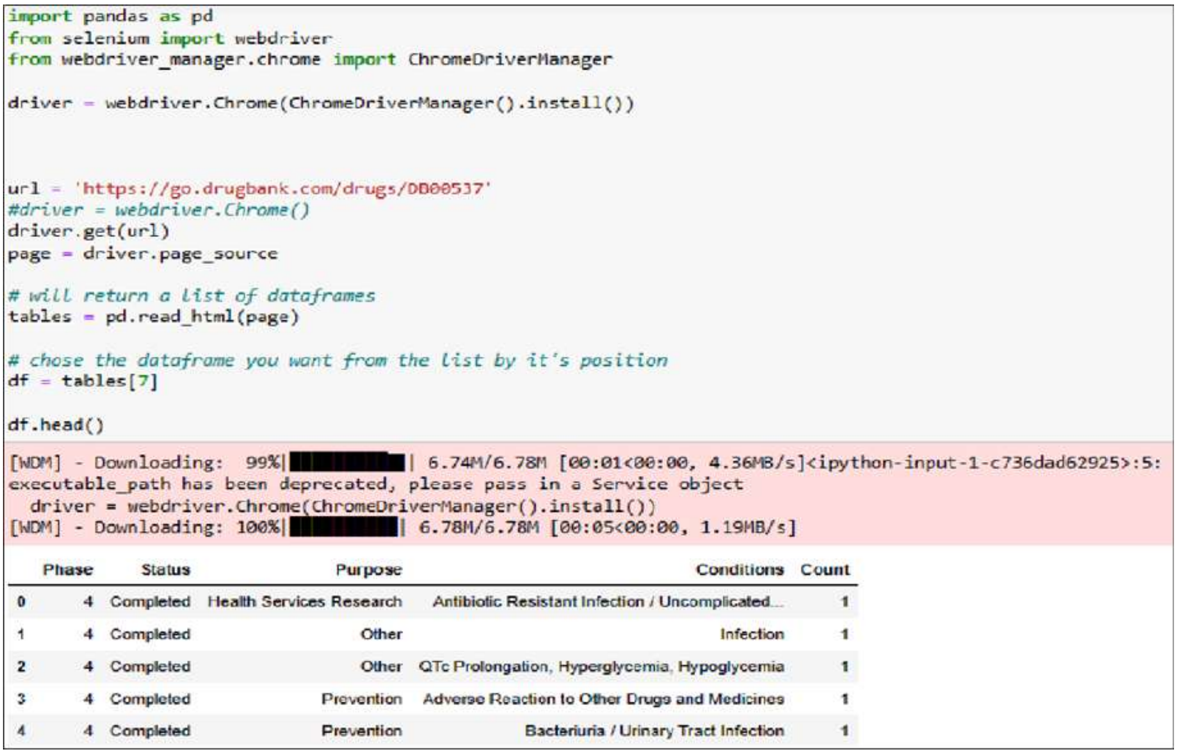
```
_id: ObjectId('641227824c2210f561a50295')
Name: "1-Palmitoyl-2-oleoyl-sn-glycero-3-(phospho-rac-(1-glycerol))"
Weight: "749.02 C40H77O10P"
Description: "A synthetic lung surfactant used to treat infant respiratory distress …"
Categories: "Not Available"

_id: ObjectId('641227834c2210f561a50296')
```

1-20 of many results

Figure 5.6 Drug Information deployed to MongoDB collection

Clinical Trials

Clinical trials is a sub-part of drug information where we gather information about the clinical trials done for each of the drugs used for each of these diseases. Thus, we use the id number of the disease and then extract the information about clinical trials in a similar sense to that of drug information extraction. The resultant extraction can be seen below:



```
import pandas as pd
from selenium import webdriver
from webdriver_manager.chrome import ChromeDriverManager

driver = webdriver.Chrome(ChromeDriverManager().install())


url = 'https://go.drugbank.com/drugs/DB00537'
#driver = webdriver.Chrome()
driver.get(url)
page = driver.page_source

# will return a list of dataframes
tables = pd.read_html(page)

# chose the dataframe you want from the list by it's position
df = tables[7]

df.head()
```

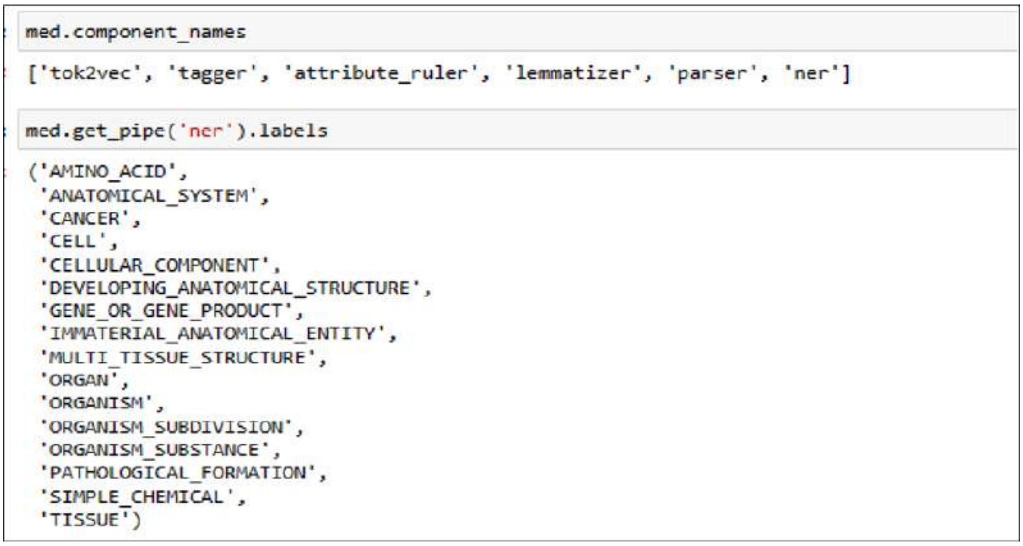
```
[WDM] - Downloading:  99%|██████████▉| 6.74M/6.78M [00:01<00:00, 4.36MB/s]<ipython-input-1-c736dad62925>:5:
executable_path has been deprecated, please pass in a Service object
  driver = webdriver.Chrome(ChromeDriverManager().install())
[WDM] - Downloading: 100%|███████████| 6.78M/6.78M [00:05<00:00, 1.19MB/s]
```

	Phase	Status	Purpose	Conditions	Count
0	4	Completed	Health Services Research	Antibiotic Resistant Infection / Uncomplicated...	1
1	4	Completed	Other	Infection	1
2	4	Completed	Other	QTc Prolongation, Hyperglycemia, Hypoglycemia	1
3	4	Completed	Prevention	Adverse Reaction to Other Drugs and Medicines	1
4	4	Completed	Prevention	Bacteriuria / Urinary Tract Infection	1

Figure 5.7 Clinical Trials Information Scraped into a pandas dataframe

Named Entity Recognition (Sci-spacy Model)

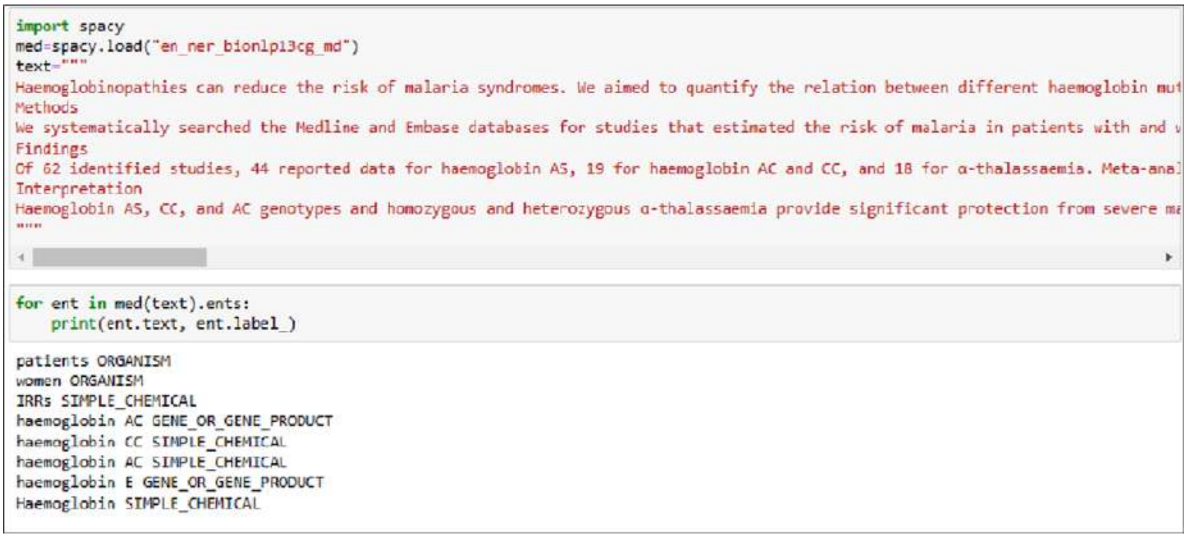
Sci-spacy model is the spacy model for processing bio-medical data that we are using for the process of Named Entity recognition. The different NER entities used by the model are shown below:



```
med.component_names

['tok2vec', 'tagger', 'attribute_ruler', 'lemmatizer', 'parser', 'ner']

med.get_pipe('ner').labels

('AMINO_ACID',
 'ANATOMICAL_SYSTEM',
 'CANCER',
 'CELL',
 'CELLULAR_COMPONENT',
 'DEVELOPING_ANATOMICAL_STRUCTURE',
 'GENE_OR_GENE_PRODUCT',
 'IMMATERIAL_ANATOMICAL_ENTITY',
 'MULTI_TISSUE_STRUCTURE',
 'ORGAN',
 'ORGANISM',
 'ORGANISM_SUBDIVISION',
 'ORGANISM_SUBSTANCE',
 'PATHOLOGICAL_FORMATION',
 'SIMPLE_CHEMICAL',
 'TISSUE')
```

Figure 5.8 Labels defined by Sci-spacy

Using the entities shown above, we use abstracts of research journals regarding diseases to classify the information into above-mentioned named entities.



```
import spacy
med=spacy.load("en_ner_bionlp13cg_md")
text="""
Haemoglobinopathies can reduce the risk of malaria syndromes. We aimed to quantify the relation between different haemoglobin mut
Methods
We systematically searched the Medline and Embase databases for studies that estimated the risk of malaria in patients with and
Findings
Of 62 identified studies, 44 reported data for haemoglobin AS, 19 for haemoglobin AC and CC, and 18 for α-thalassaemia. Meta-anal
Interpretation
Haemoglobin AS, CC, and AC genotypes and homozygous and heterozygous α-thalassaemia provide significant protection from severe ma
"""
```

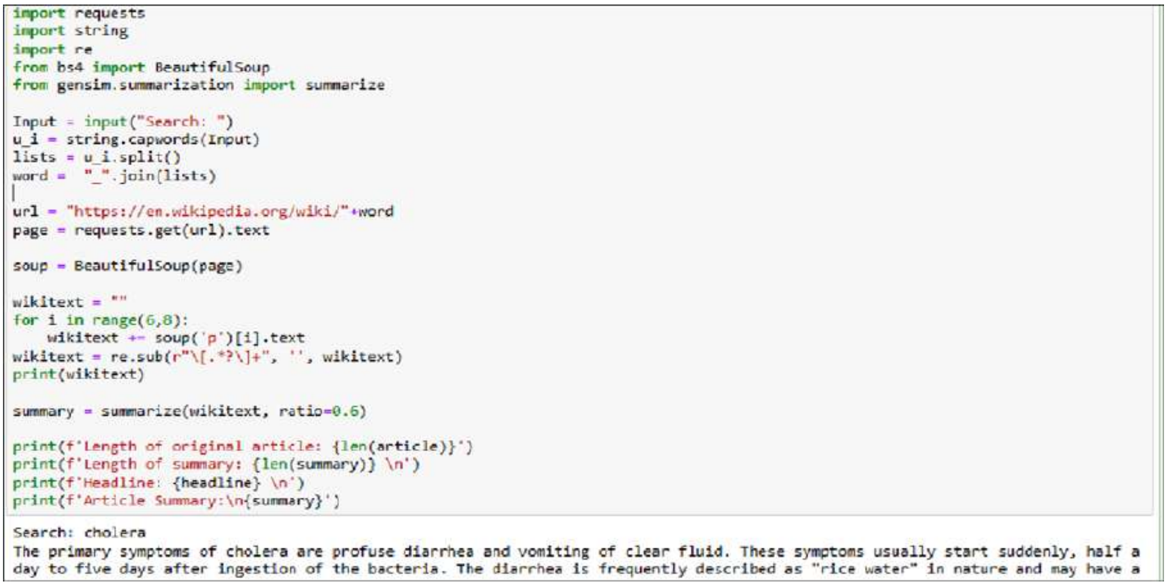
```
for ent in med(text).ents:
    print(ent.text, ent.label_)
```

```
patients ORGANISM
women ORGANISM
IRRs SIMPLE_CHEMICAL
haemoglobin AC GENE_OR_GENE_PRODUCT
haemoglobin CC SIMPLE_CHEMICAL
haemoglobin AC SIMPLE_CHEMICAL
haemoglobin E GENE_OR_GENE_PRODUCT
Haemoglobin SIMPLE_CHEMICAL
```

Figure 5.9 Using abstracts on the Sci-spacy model

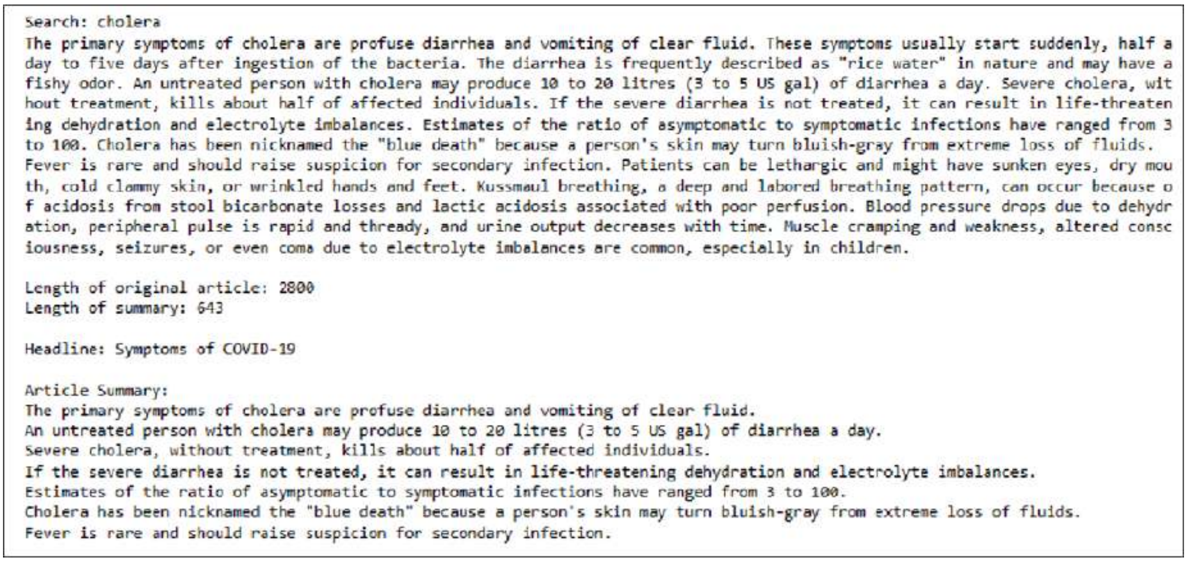
Text Summarization

We use the gensim.summarization library for the process of text summarization. The text presented to the gensim model is extracted using the procedure of web scraping and then based on the required ratio, we can get the summarized result as shown below:



```
import requests
import string
import re
from bs4 import BeautifulSoup
from gensim.summarization import summarize

Input = input("Search: ")
u_i = string.capwords(Input)
lists = u_i.split()
word =  "_".join(lists)
|
url = "https://en.wikipedia.org/wiki/"+word
page = requests.get(url).text

soup = BeautifulSoup(page)

wikitext = ""
for i in range(6,8):
    wikitext += soup('p')[i].text
wikitext = re.sub(r"\[.*?\]+", '', wikitext)
print(wikitext)

summary = summarize(wikitext, ratio=0.6)

print(f'Length of original article: {len(article)}')
print(f'Length of summary: {len(summary)} \n')
print(f'Headline: {headline} \n')
print(f'Article Summary:\n{summary}')

Search: cholera
The primary symptoms of cholera are profuse diarrhea and vomiting of clear fluid. These symptoms usually start suddenly, half a
day to five days after ingestion of the bacteria. The diarrhea is frequently described as "rice water" in nature and may have a
```

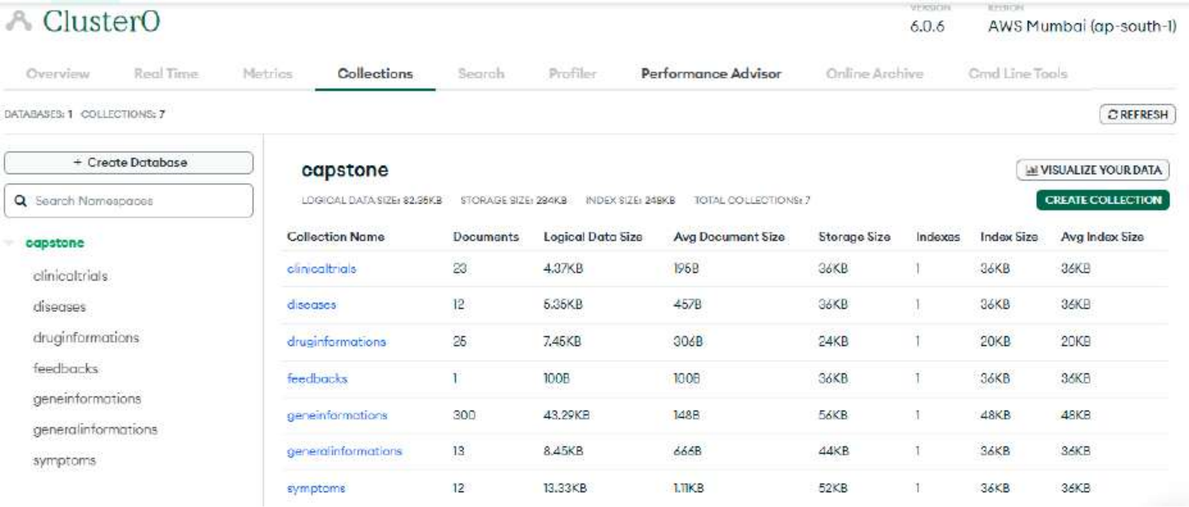
Figure 5.10 Summarizing Symptoms information of diseases



```
Search: cholera
The primary symptoms of cholera are profuse diarrhea and vomiting of clear fluid. These symptoms usually start suddenly, half a
day to five days after ingestion of the bacteria. The diarrhea is frequently described as "rice water" in nature and may have a
fishy odor. An untreated person with cholera may produce 10 to 20 litres (3 to 5 US gal) of diarrhea a day. Severe cholera, wit
hout treatment, kills about half of affected individuals. If the severe diarrhea is not treated, it can result in life-threaten
ing dehydration and electrolyte imbalances. Estimates of the ratio of asymptomatic to symptomatic infections have ranged from 3
to 100. Cholera has been nicknamed the "blue death" because a person's skin may turn bluish-gray from extreme loss of fluids.
Fever is rare and should raise suspicion for secondary infection. Patients can be lethargic and might have sunken eyes, dry mou
th, cold clammy skin, or wrinkled hands and feet. Kussmaul breathing, a deep and labored breathing pattern, can occur because o
f acidosis from stool bicarbonate losses and lactic acidosis associated with poor perfusion. Blood pressure drops due to dehydr
ation, peripheral pulse is rapid and thready, and urine output decreases with time. Muscle cramping and weakness, altered consc
iousness, seizures, or even coma due to electrolyte imbalances are common, especially in children.

Length of original article: 2800
Length of summary: 643

Headline: Symptoms of COVID-19

Article Summary:
The primary symptoms of cholera are profuse diarrhea and vomiting of clear fluid.
An untreated person with cholera may produce 10 to 20 litres (3 to 5 US gal) of diarrhea a day.
Severe cholera, without treatment, kills about half of affected individuals.
If the severe diarrhea is not treated, it can result in life-threatening dehydration and electrolyte imbalances.
Estimates of the ratio of asymptomatic to symptomatic infections have ranged from 3 to 100.
Cholera has been nicknamed the "blue death" because a person's skin may turn bluish-gray from extreme loss of fluids.
Fever is rare and should raise suspicion for secondary infection.
```

Figure 5.11 Summarized result

MongoDB Collection



capstone

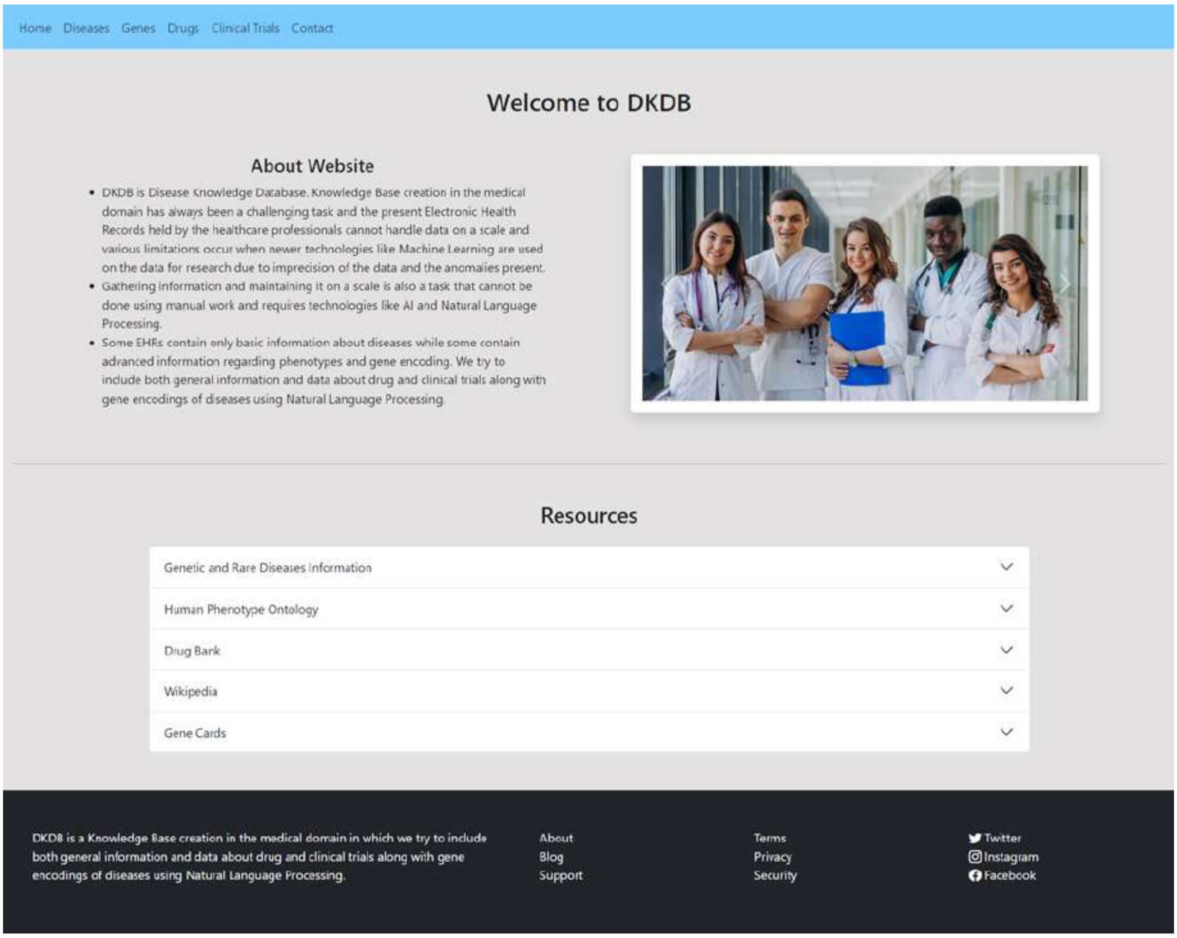
LOGICAL DATA SIZE: 82.35KB STORAGE SIZE: 284KB INDEX SIZE: 248KB TOTAL COLLECTIONS: 7

Collection Name	Documents	Logical Data Size	Avg Document Size	Storage Size	Indexes	Index Size	Avg Index Size
clinicaltrials	23	4.37KB	196B	36KB	1	36KB	36KB
diseases	12	5.35KB	457B	36KB	1	36KB	36KB
druginformations	25	7.45KB	306B	24KB	1	20KB	20KB
feedbacks	1	100B	100B	36KB	1	36KB	36KB
geneinformations	300	43.29KB	148B	56KB	1	48KB	48KB
generalinformations	13	8.45KB	666B	44KB	1	36KB	36KB
symptoms	12	13.33KB	1.11KB	52KB	1	36KB	36KB

Figure 5.12 MongoDB collection

Website Home Page

The homepage of our website contains some information about what the website provides and the need for such a website and also the resources that were used while building the website as shown below. Along with that, the navigation bar contains direct hyperlinks to the disease list page, the gene information page, the drugs information page, the clinical trials page and finally the contact us page. The footer contains some basic information about copyright and links to social handles.



Welcome to DKDB

About Website

- DKDB is Disease Knowledge Database. Knowledge Base creation in the medical domain has always been a challenging task and the present Electronic Health Records held by the healthcare professionals cannot handle data on a scale and various limitations occur when newer technologies like Machine Learning are used on the data for research due to imprecision of the data and the anomalies present.
- Gathering information and maintaining it on a scale is also a task that cannot be done using manual work and requires technologies like AI and Natural Language Processing.
- Some EHRs contain only basic information about diseases while some contain advanced information regarding phenotypes and gene encoding. We try to include both general information and data about drug and clinical trials along with gene encodings of diseases using Natural Language Processing.

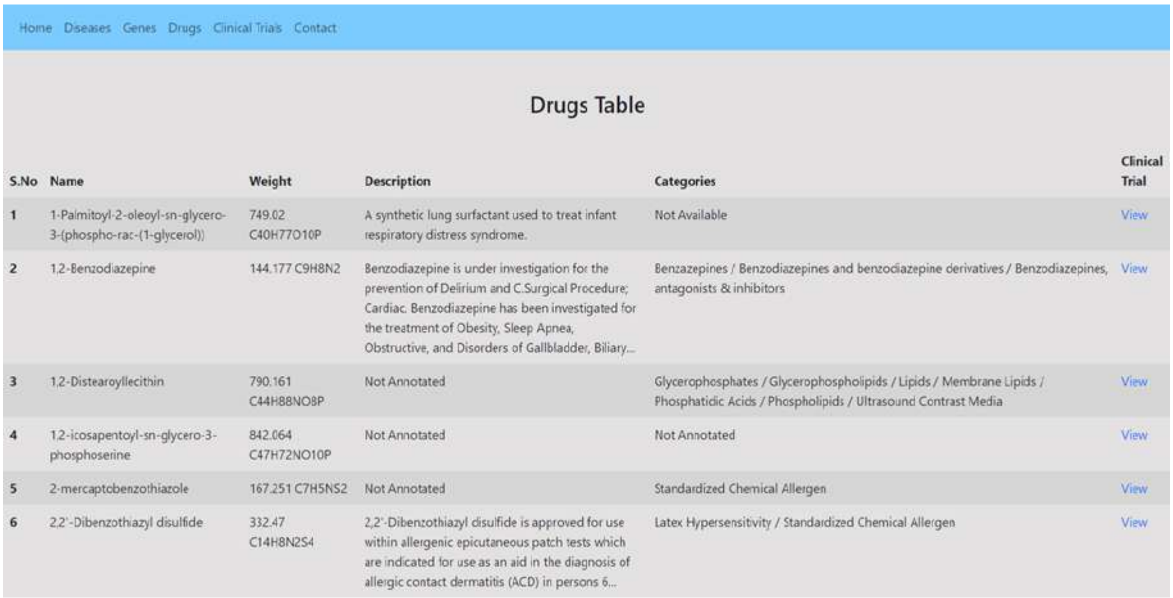
Resources

DKDB is a Knowledge Base creation in the medical domain in which we try to include both general information and data about drug and clinical trials along with gene encodings of diseases using Natural Language Processing.

Figure 5.13 Website Home Page

Drugs Information Webpage

Once we click Drugs in the navigation bar, we move to this page where we can see the list of drugs present on the website currently and we can see information regarding the weight, basic description and category the drug belongs to along with hyperlinks to find more information regarding the clinical trials used for that particular drug.



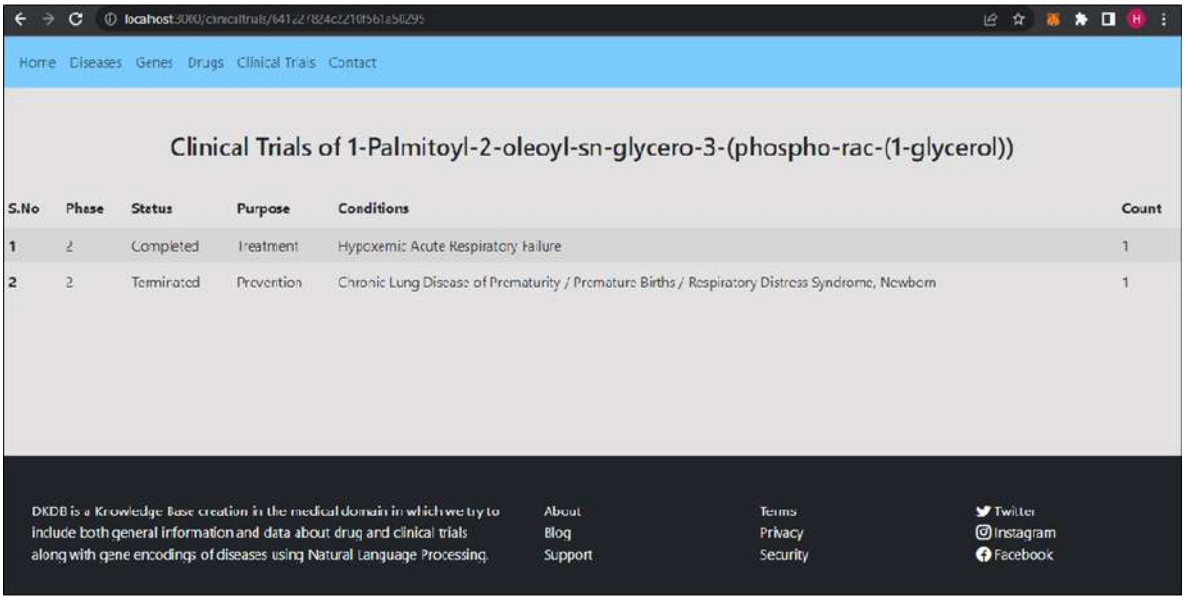
Drugs Table

S.No	Name	Weight	Description	Categories	Clinical Trial
1	1-Palmitoyl-2-oleoyl-sn-glycero-3-(phospho-rac-(1-glycerol))	749.02 C40H77O10P	A synthetic lung surfactant used to treat infant respiratory distress syndrome.	Not Available	View
2	1,2-Benzodiazepine	144.177 C9H8N2	Benzodiazepine is under investigation for the prevention of Delirium and C.Surgical Procedure; Cardiac. Benzodiazepine has been investigated for the treatment of Obesity, Sleep Apnea, Obstructive, and Disorders of Gallbladder, Biliary...	Benzazepines / Benzodiazepines and benzodiazepine derivatives / Benzodiazepines, antagonists & inhibitors	View
3	1,2-Distearoyllecithin	790.161 C44H88NO8P	Not Annotated	Glycerophosphates / Glycerophospholipids / Lipids / Membrane Lipids / Phosphatidic Acids / Phospholipids / Ultrasound Contrast Media	View
4	1,2-icosapentoyl-sn-glycero-3-phosphoserine	842.064 C47H72NO10P	Not Annotated	Not Annotated	View
5	2-mercaptobenzothiazole	167.251 C7H5NS2	Not Annotated	Standardized Chemical Allergen	View
6	2,2'-Dibenzothiazyl disulfide	332.47 C14H8N2S4	2,2'-Dibenzothiazyl disulfide is approved for use within allergenic epicutaneous patch tests which are indicated for use as an aid in the diagnosis of allergic contact dermatitis (ACD) in persons 6...	Latex Hypersensitivity / Standardized Chemical Allergen	View

Figure 5.14 Drugs Information Page

Clinical Trials information of a particular drug

As discussed earlier, once we click on clinical trial information on any particular drug, we will move to a page where the clinical trials done for that drug would be visible as shown below:



Clinical Trials of 1-Palmitoyl-2-oleoyl-sn-glycero-3-(phospho-rac-(1-glycerol))

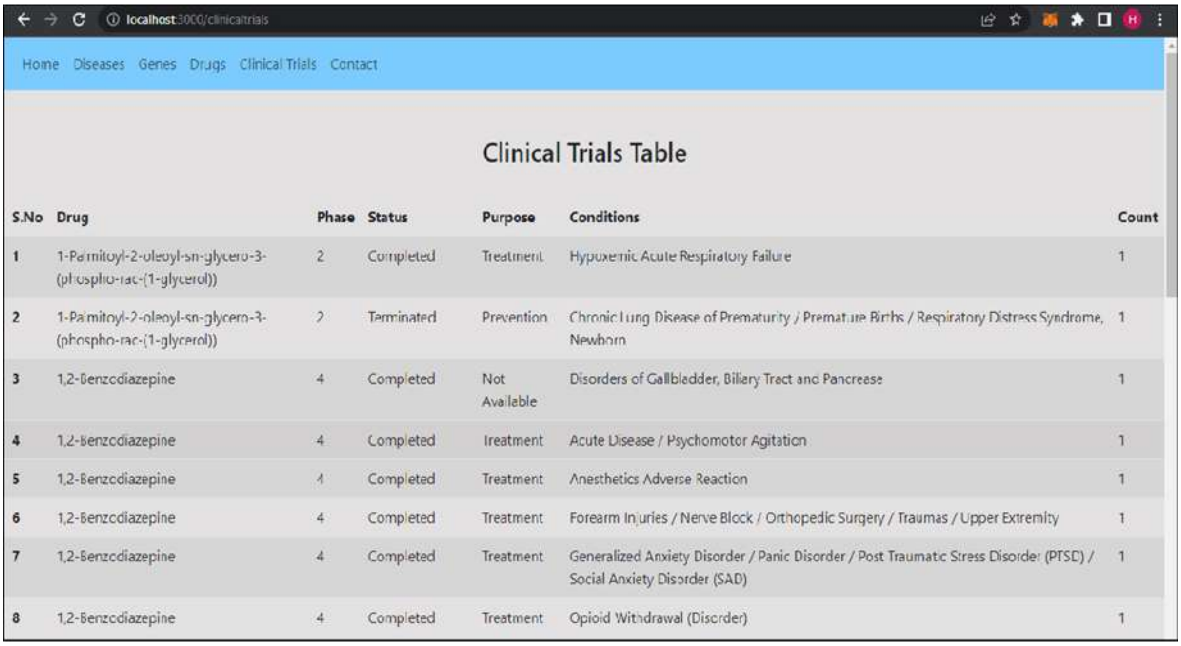
S.No	Phase	Status	Purpose	Conditions	Count
1	2	Completed	Treatment	Hypoxemic Acute Respiratory Failure	1
2	2	Terminated	Prevention	Chronic Lung Disease of Prematurity / Premature Births / Respiratory Distress Syndrome, Newborn	1

DKDB is a Knowledge Base creation in the medical domain in which we try to include both general information and data about drug and clinical trials along with gene encodings of diseases using Natural Language Processing.

Figure 5.15 Clinical Trials information for a particular drug

Complete Clinical trials Information

Other than clinical trials information for a particular drug, the users can also see information regarding all the clinical trials done for various drugs once they click on 'Clinical Trials' in the navigation bar as shown below:



Clinical Trials Table

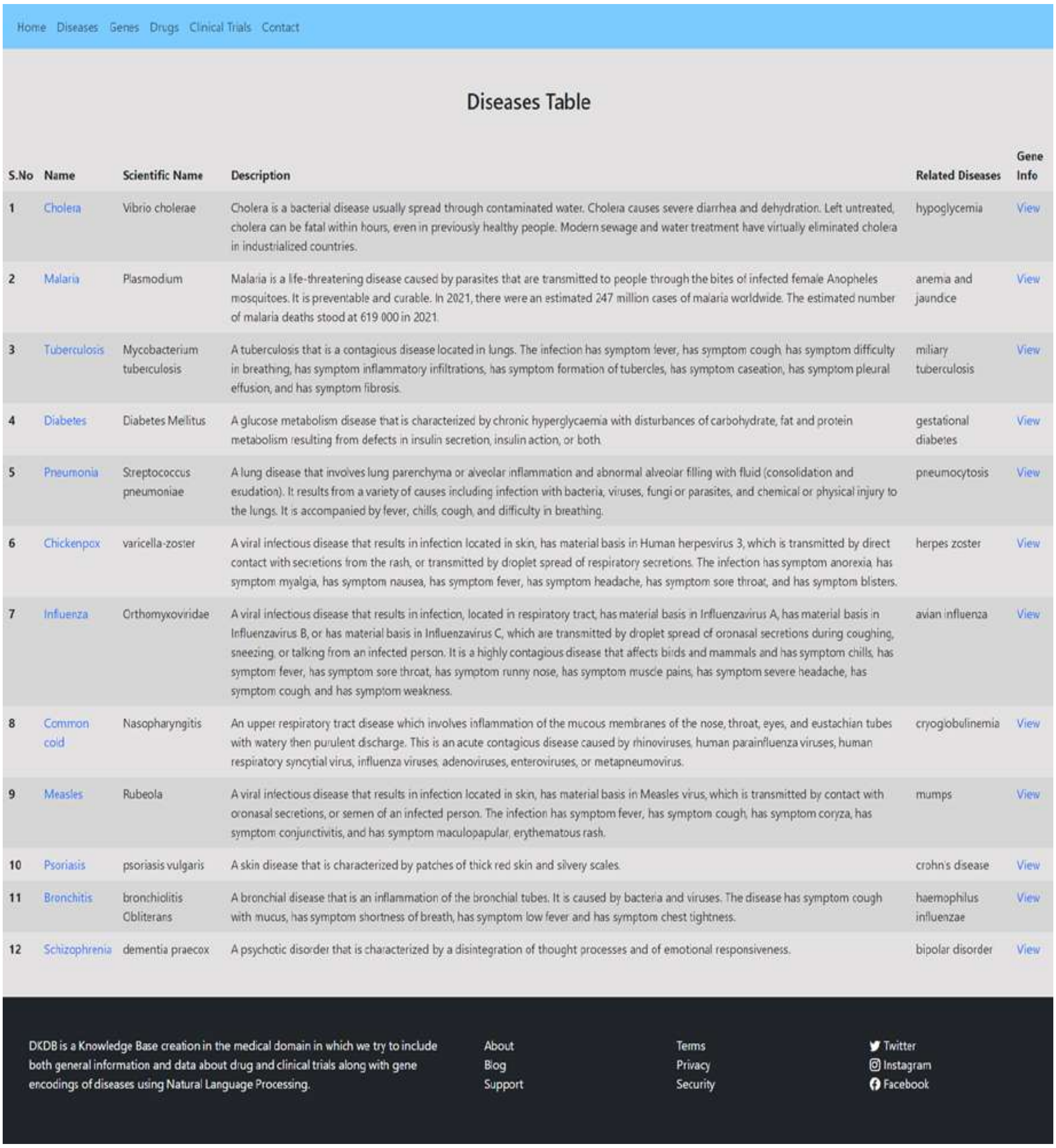
S.No	Drug	Phase	Status	Purpose	Conditions	Count
1	1-Palmitoyl-2-oleoyl-sn-glycero-3-(phospho-rac-(1-glycerol))	2	Completed	Treatment	Hypoxemic Acute Respiratory Failure	1
2	1-Palmitoyl-2-oleoyl-sn-glycero-3-(phospho-rac-(1-glycerol))	2	Terminated	Prevention	Chronic Lung Disease of Prematurity / Premature Births / Respiratory Distress Syndrome, Newborn	1
3	1,2-Benzodiazepine	4	Completed	Not Available	Disorders of Gallbladder, Biliary Tract and Pancrease	1
4	1,2-Benzodiazepine	4	Completed	Treatment	Acute Disease / Psychomotor Agitation	1
5	1,2-Benzodiazepine	4	Completed	Treatment	Anesthetics Adverse Reaction	1
6	1,2-Benzodiazepine	4	Completed	Treatment	Forearm Injuries / Nerve Block / Orthopedic Surgery / Traumas / Upper Extremity	1
7	1,2-Benzodiazepine	4	Completed	Treatment	Generalized Anxiety Disorder / Panic Disorder / Post Traumatic Stress Disorder (PTSD) / Social Anxiety Disorder (SAD)	1
8	1,2-Benzodiazepine	4	Completed	Treatment	Opioid Withdrawal (Disorder)	1

Figure 5.16 Clinical Trials information of all the drugs

The clinical trials page would contain information regarding the drug on which these trials were done, the phase in which they were done, completion status, purpose of the trial, lab conditions under which the trial was undergone and also the number of times the trial was done

Diseases Page

From the navigation bar, once we click on diseases, we move to the disease table page wherein we can see the list of all the diseases currently present on the website with some basic information regarding the disease name, scientific name and some basic description. Further, once the user clicks on any particular disease name, they would be moved to the general information page of that particular drug which will be discussed later in the document. The users can also view gene related information for any particular disease by using the link provided in the table for any corresponding disease.



Diseases Table

S.No	Name	Scientific Name	Description	Related Diseases	Gene Info
1	Cholera	Vibrio cholerae	Cholera is a bacterial disease usually spread through contaminated water. Cholera causes severe diarrhea and dehydration. Left untreated, cholera can be fatal within hours, even in previously healthy people. Modern sewage and water treatment have virtually eliminated cholera in industrialized countries.	hypoglycemia	View
2	Malaria	Plasmodium	Malaria is a life-threatening disease caused by parasites that are transmitted to people through the bites of infected female Anopheles mosquitoes. It is preventable and curable. In 2021, there were an estimated 247 million cases of malaria worldwide. The estimated number of malaria deaths stood at 619 000 in 2021.	anemia and jaundice	View
3	Tuberculosis	Mycobacterium tuberculosis	A tuberculosis that is a contagious disease located in lungs. The infection has symptom fever, has symptom cough, has symptom difficulty in breathing, has symptom inflammatory infiltrations, has symptom formation of tubercles, has symptom caseation, has symptom pleural effusion, and has symptom fibrosis.	miliary tuberculosis	View
4	Diabetes	Diabetes Mellitus	A glucose metabolism disease that is characterized by chronic hyperglycaemia with disturbances of carbohydrate, fat and protein metabolism resulting from defects in insulin secretion, insulin action, or both.	gestational diabetes	View
5	Pneumonia	Streptococcus pneumoniae	A lung disease that involves lung parenchyma or alveolar inflammation and abnormal alveolar filling with fluid (consolidation and exudation). It results from a variety of causes including infection with bacteria, viruses, fungi or parasites, and chemical or physical injury to the lungs. It is accompanied by fever, chills, cough, and difficulty in breathing.	pneumocytosis	View
6	Chickenpox	varicella-zoster	A viral infectious disease that results in infection located in skin, has material basis in Human herpesvirus 3, which is transmitted by direct contact with secretions from the rash, or transmitted by droplet spread of respiratory secretions. The infection has symptom anorexia, has symptom myalgia, has symptom nausea, has symptom fever, has symptom headache, has symptom sore throat, and has symptom blisters.	herpes zoster	View
7	Influenza	Orthomyxoviridae	A viral infectious disease that results in infection, located in respiratory tract, has material basis in Influenzavirus A, has material basis in Influenzavirus B, or has material basis in Influenzavirus C, which are transmitted by droplet spread of oronasal secretions during coughing, sneezing, or talking from an infected person. It is a highly contagious disease that affects birds and mammals and has symptom chills, has symptom fever, has symptom sore throat, has symptom runny nose, has symptom muscle pains, has symptom severe headache, has symptom cough, and has symptom weakness.	avian influenza	View
8	Common cold	Nasopharyngitis	An upper respiratory tract disease which involves inflammation of the mucous membranes of the nose, throat, eyes, and eustachian tubes with watery then purulent discharge. This is an acute contagious disease caused by rhinoviruses, human parainfluenza viruses, human respiratory syncytial virus, influenza viruses, adenoviruses, enteroviruses, or metapneumovirus.	cryoglobulinemia	View
9	Measles	Rubeola	A viral infectious disease that results in infection located in skin, has material basis in Measles virus, which is transmitted by contact with oronasal secretions, or semen of an infected person. The infection has symptom fever, has symptom cough, has symptom coryza, has symptom conjunctivitis, and has symptom maculopapular, erythematous rash.	mumps	View
10	Psoriasis	psoriasis vulgaris	A skin disease that is characterized by patches of thick red skin and silvery scales.	crohn's disease	View
11	Bronchitis	bronchiolitis Obliterans	A bronchial disease that is an inflammation of the bronchial tubes. It is caused by bacteria and viruses. The disease has symptom cough with mucus, has symptom shortness of breath, has symptom low fever and has symptom chest tightness.	haemophilus influenzae	View
12	Schizophrenia	dementia praecox	A psychotic disorder that is characterized by a disintegration of thought processes and of emotional responsiveness.	bipolar disorder	View

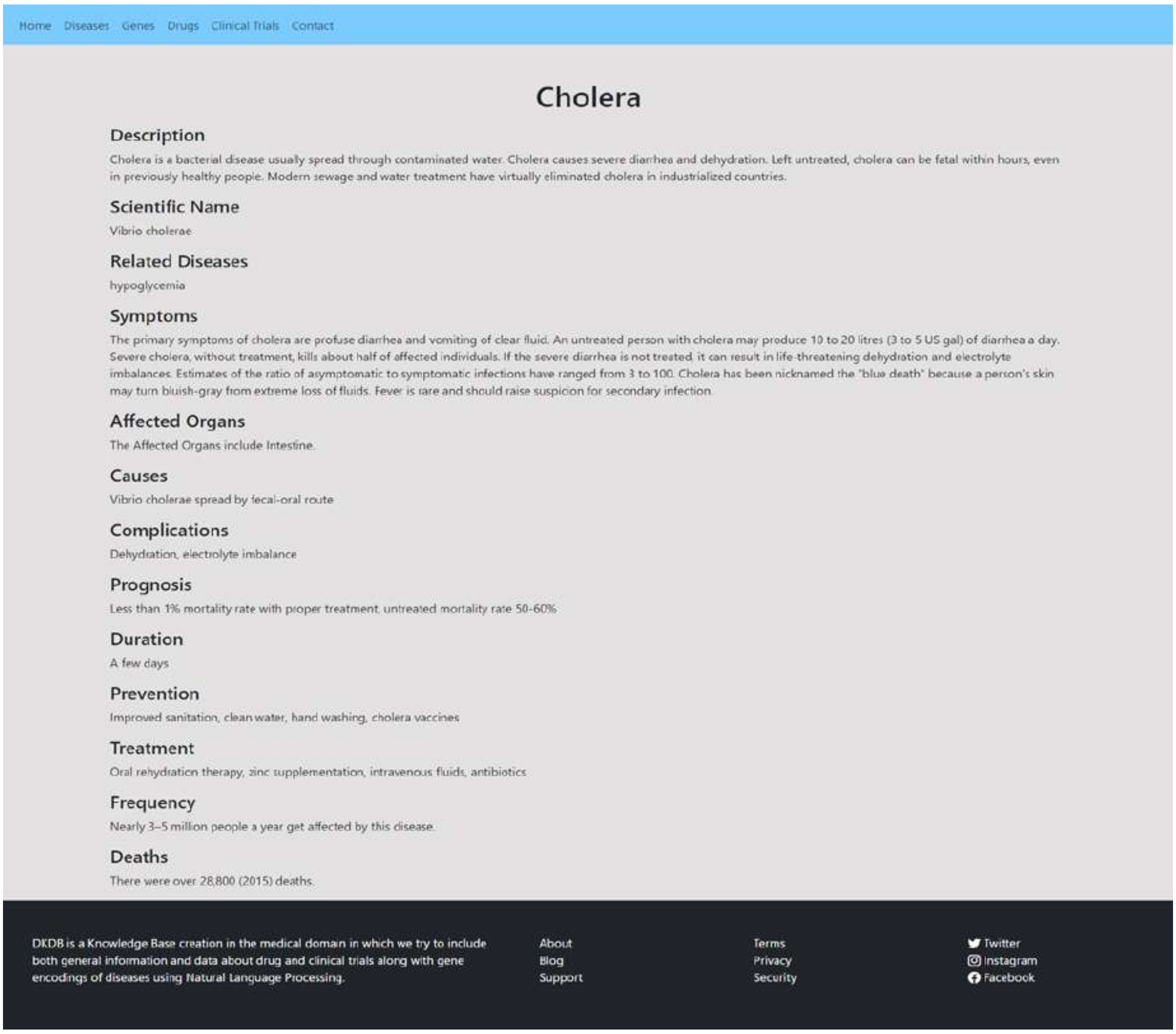
DKDB is a Knowledge Base creation in the medical domain in which we try to include both general information and data about drug and clinical trials along with gene encodings of diseases using Natural Language Processing.

Figure 5.17 Disease Table

General Information of disease

Once the user clicks on the name of any disease, they will be moved to the general information page of that particular disease. Here we can find information about various entities of that disease like the symptoms, affected organs, causes, complications, duration of the disease and finally prevention and treatments methods used against the disease.

The symptoms for the diseases were derived using text summarization while the affected organs information was gathered using Named Entity recognition. The general information for cholera can be seen below:



Cholera

Description

Cholera is a bacterial disease usualy spread through contaminated water. Cholera causes severe diarrhea and dehydration. Left untreated, cholera can be fatal within hours, even in previously healthy people. Modern sewage and water treatment have virtually eliminated cholera in industrialized countries.

Scientific Name

Vibrio cholerae

Related Diseases

hypoglycemia

Symptoms

The primary symptoms of cholera are profuse diarrhea and vomiting of clear fluid. An untreated person with cholera may produce 10 to 20 litres (3 to 5 US gal) of diarrhea a day. Severe cholera, without treatment, kills about half of affected individuals. If the severe diarrhea is not treated, it can result in life-threatening dehydration and electrolyte imbalances. Estimates of the ratio of asymptomatic to symptomatic infections have ranged from 3 to 100. Cholera has been nicknamed the "blue death" because a person's skin may turn bluish-gray from extreme loss of fluids. Fever is rare and should raise suspicion for secondary infection.

Affected Organs

The Affected Organs include Intestine.

Causes

Vibrio cholerae spread by fecal-oral route

Complications

Dehydration, electrolyte imbalance

Prognosis

Less than 1% mortality rate with proper treatment, untreated mortality rate 50-60%

Duration

A few days

Prevention

Improved sanitation, clean water, hand washing, cholera vaccines

Treatment

Oral rehydration therapy, zinc supplementation, intravenous fluids, antibiotics

Frequency

Nearly 3–5 million people a year get affected by this disease.

Deaths

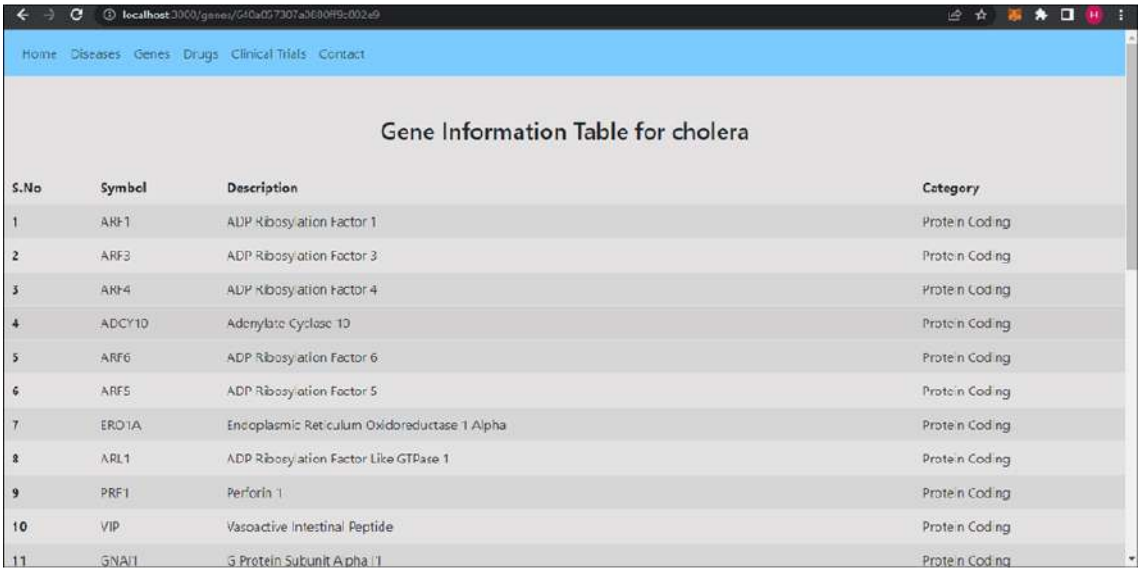
There were over 28,800 (2015) deaths.

DKDB is a Knowledge Base creation in the medical domain in which we try to include both general information and data about drug and clinical trials along with gene encodings of diseases using Natural Language Processing.

Figure 5.18 General Information Page

Gene Information of a disease

The gene information regarding the genes related to a disease like the symbol of the gene encoding, description and category of the genes are displayed in this webpage as shown.



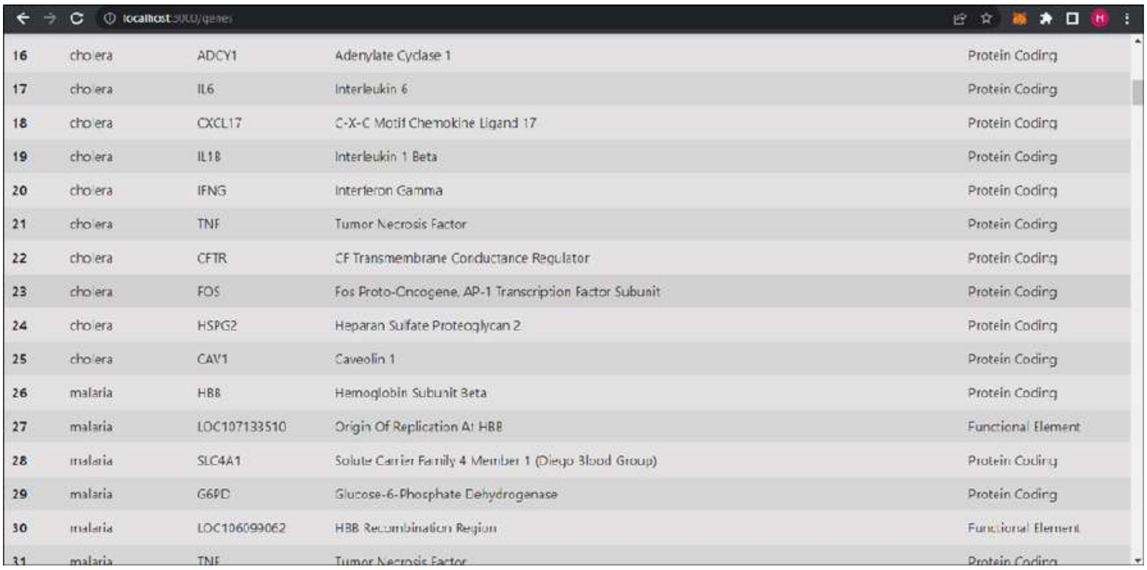
Gene Information Table for cholera

S.No	Symbol	Description	Category
1	ARF1	ADP Ribosylation Factor 1	Protein Coding
2	ARF3	ADP Ribosylation Factor 3	Protein Coding
3	ARF4	ADP Ribosylation Factor 4	Protein Coding
4	ADCY10	Adenylate Cyclase 10	Protein Coding
5	ARF6	ADP Ribosylation Factor 6	Protein Coding
6	ARF5	ADP Ribosylation Factor 5	Protein Coding
7	ERO1A	Endoplasmic Reticulum Oxidoreductase 1 Alpha	Protein Coding
8	ARL1	ADP Ribosylation Factor Like GTPase 1	Protein Coding
9	PRF1	Perforin 1	Protein Coding
10	VIP	Vasoactive Intestinal Peptide	Protein Coding
11	GNAI1	G Protein Subunit Alpha I1	Protein Coding

Figure 5.19 Gene Information of a particular disease

Complete Gene Information

The gene information for all the genes related to all the diseases can be seen in this page and the disease to which the genes are related can also be seen as shown. This page can directly be accessed through the navigation bar.

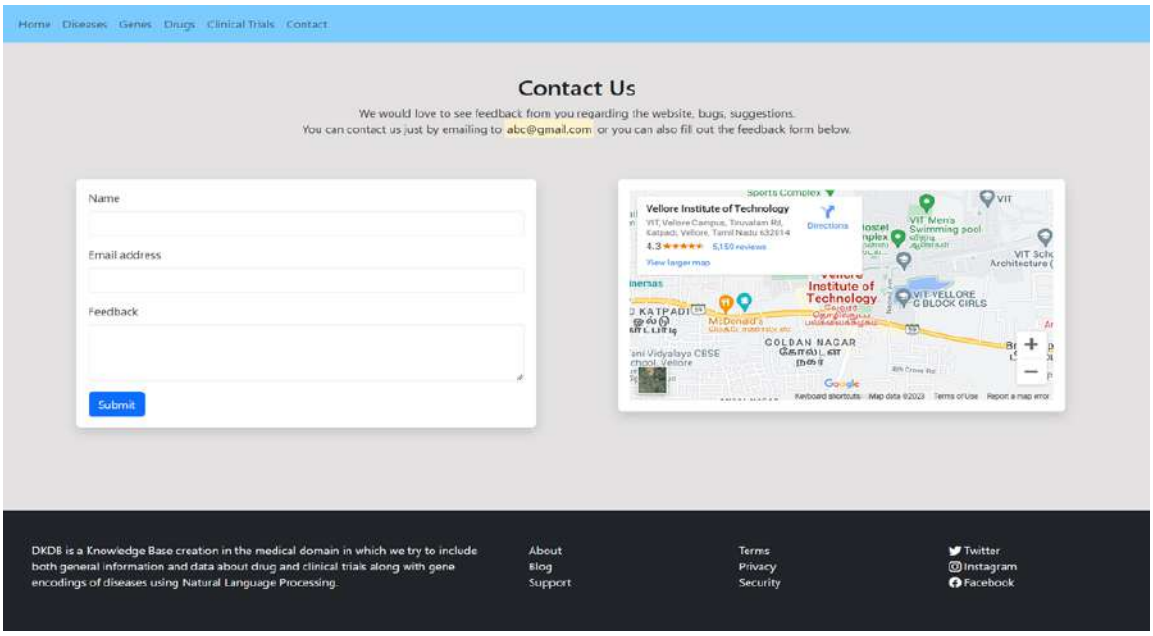


16	cholera	ADCY1	Adenylate Cyclase 1	Protein Coding
17	cholera	IL6	Interleukin 6	Protein Coding
18	cholera	CXCL17	C-X-C Motif Chemokine Ligand 17	Protein Coding
19	cholera	IL1B	Interleukin 1 Beta	Protein Coding
20	cholera	IFNG	Interferon Gamma	Protein Coding
21	cholera	TNF	Tumor Necrosis Factor	Protein Coding
22	cholera	CFTR	CF Transmembrane Conductance Regulator	Protein Coding
23	cholera	FOS	Fos Proto-Oncogene, AP-1 Transcription Factor Subunit	Protein Coding
24	cholera	HSPG2	Heparan Sulfate Proteoglycan 2	Protein Coding
25	cholera	CAV1	Caveolin 1	Protein Coding
26	malaria	HBB	Hemoglobin Subunit Beta	Protein Coding
27	malaria	LOC107133510	Origin Of Replication At HBB	Functional Element
28	malaria	SLC4A1	Solute Carrier Family 4 Member 1 (Diego Blood Group)	Protein Coding
29	malaria	G6PD	Glucose-6-Phosphate Dehydrogenase	Protein Coding
30	malaria	LOC106099062	HBB Recombination Region	Functional Element
31	malaria	TNF	Tumor Necrosis Factor	Protein Coding

Figure 5.20 Complete Gene Information Page

Contact Us Page

The final page in this website is the contact us page, where the users can send any kind of feedback that can be related to the performance or bugs related to the website or they can request addition of entities/diseases or any modifications to the existing information present on the website. This information will be stored in the database and the developers can access the information and take action according to need.



Contact Us

We would love to see feedback from you regarding the website, bugs, suggestions.
You can contact us just by emailing to abc@gmail.com or you can also fill out the feedback form below.

Name

Email address

Feedback

Submit

DKDB is a Knowledge Base creation in the medical domain in which we try to include both general information and data about drug and clinical trials along with gene encodings of diseases using Natural Language Processing.

About
Blog
Support

Terms
Privacy
Security

Twitter
Instagram
Facebook

Figure 5.21 Contact Us page

Conclusion and Future Work

An Online Database/Knowledge Base was created using Natural Language Processing Techniques like Web Scraping, Text Summarization and Named Entity Recognition in the Disease Information domain. The data collected was stored in a NoSQL database and NodeJS was used as our backend server. A simple user-friendly interface was created which displays both basic and advanced information regarding diseases, their gene composition and the drugs used, and all the resources used were cited in the same. The interface is quick and open to all without any need for user login and provides accurate information to the best of our knowledge. This website can now be used by the public who want to get some generic information about diseases and by researchers/analysts who need the data to verify past trends and advanced gene compositions of pathogens.

Although the information displayed was deemed accurate, our website relies on the accuracy of the sources used. A domain expert verification before publishing the data can be useful. The inclusion of a domain expert in the project can be done in two ways, one is that the expert will validate and approve the data before the data is stored in the database while working hand in hand with the data collection team and other method is to have an expert login feature where the experts can see all the data collected but not yet published and then can approve the data that needs to be published into the website. This way the accuracy of the information displayed on the website is further improved. Further, many other entities related to diseases which provide knowledge in depth in the field of biology like pathological formations, their structure photos can be added into the website.

6. REFERENCES

Web links

1. https://allenai.github.io/scispacy/ (accessed in March 2023).
2. https://radimrehurek.com/gensim_3.8.3/summarization/summariser.html (accessed in March 2023).
3. https://pypi.org/project/selenium/ (accessed in March 2023).
4. https://www.crummy.com/software/BeautifulSoup/bs4/doc/ (accessed in March 2023).

Journal

1. Jinhyuk Lee, Wonjin Yoon 1, Sungdong Kim, Donghyeon Kim, Sunkyu Kim, Chan Ho So and Jaewoo Kang, “BioBERT: a pre-trained biomedical language representation model for biomedical text mining” Bioinformatics, 2019, 1–7 doi: 10.1093/bioinformatics/btz682.
2. Chaojie Wen, Tao Chen, Xudong Jia, Jiang Zhu, “Medical Named Entity Recognition from Un-labelled Medical Records based on Pre-trained Language Models and Domain Dictionary” Data Intelligence 3(3), 402-417 (2021). doi: 10.1162/dint_a_00105.
3. R. Ramachandran, K. Arutchelvan, “Named entity recognition on bio-medical literature documents using hybrid based approach” J Ambient Intell Humaniz Comput. 2021 Mar 11:1-10. doi: 10.1007/s12652-021-03078-z.
4. Yang Liu and Mirella Lapata, “Text Summarization with Pre Trained Encoders” arXiv:1908.08345v2 [cs.CL] 5 Sep 2019.
5. Jingqing Zhang, Yao Zhao, Mohammad Saleh, Peter Liu, “PEGASUS: Pre-training with Extracted Gap-sentences for Abstractive Summarization” arXiv:1912.08777v3 [cs.CL]
6. Cornelia A. Győrödi, Diana V. Dumşe-Burescu, Doina R. Zmaranda and Robert Ş. Győrödi, “A Comparative Study of MongoDB and Document-Based MySQL for Big Data Application Data Management” 10 Jul 2020 Big Data and Cognitive Computing. 2022; 6(2):49. https://doi.org/10.3390/bdcc6020049.
7. M. Praveen Kumar, S.P.Santhoshkumar, T. Gowdhaman, A. Wasim Raja, “Handling Big Data using NoSQL Database” 2015 IEEE 29th International Conference on

Advanced Information Networking and Applications Workshops, Gwangju, Korea (South), 2015, pp. 393-398, doi: 10.1109/WAINA.2015.19.

8. Muhammad Iqbal Habibie, Taufiq Widiaputra, Yulianingsani, “Web Scraping of Disease information from social media Twitter” Jurnal Teknoinfo Volume 16, Nomor 2, Juli 2022, Page 246-251 ISSN: 1693-0010(Print), ISSN: 2615-224X(Online).
9. R. S. Chaulagain, S. Pandey, S. R. Basnet and S. Shakya, "Cloud Based Web Scraping for Big Data Applications," 2017 IEEE International Conference on Smart Cloud (SmartCloud), New York, NY, USA, 2017, pp. 138-143, doi: 10.1109/SmartCloud.2017.28.
10. Dobreva, J., Jofche, N., Jovanovik, M., Trajanov, D. (2020). Improving NER Performance by Applying Text Summarization on Pharmaceutical Articles. In: Dimitrova, V., Dimitrovski, I. (eds) ICT Innovations 2020. Machine Learning and Applications. ICT Innovations 2020. Communications in Computer and Information Science, vol 1316. Springer, Cham. https://doi.org/10.1007/978-3-030-62098-1_8.
11. Demian Gholipour Ghalandari, Chris Hokamp, Nghia The Pham, John Glover, Georgiana Ifrim, “A Large-Scale Multi-Document Summarization Dataset from the Wikipedia Current Events Portal” arXiv:2005.10070v1 [cs.CL] 20 May 2020.
12. Hyejin Cho, Hyunju Lee, “Biomedical named entity recognition using deep neural networks with contextual information” December 2019 BMC Bioinformatics 20(1) DOI:10.1186/s12859-019-3321-4.