### A Mini Project Report on

**COVID19 BOT**

**Submitted to**

**Jawaharlal Nehru Technological University, Hyderabad**

***in partial fulfillment of requirements for the award of the degree of* BACHELOR OF TECHNOLOGY**

## in

**COMPUTER SCIENCE AND ENGINEERING**

### By

**J.Aditi Patel(18BD1A052Q)**

**A. Bhandavi (18BD1A0522)**

**Faseeh Ahmed (18BD1A053H)**

**M. Samendu Reddy (18BD1A053J)**

**Under the guidance of**

**Chinnam Madhuri**

Assistant Professor Department of CSE



## Department of Computer Science and Engineering

## KESHAV MEMORIAL INSTITUTE OF TECHNOLOGY

Approved by AICTE, Affiliated to JNTUH

3-5-1206, Narayanaguda, Hyderabad – 500029

**2021-2022**

**KESHAV MEMORIAL INSTITUTE OF TECHNOLOGY**

***(Accredited by NBA & NAAC, Approved By A.I.C.T.E., Reg by Govt of Telangana State & Affiliated to JNTU, Hyderabad)***

**DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING**

**CERTIFICATE**

This is to certify that the project entitled **COVID19 BOT** being submitted by

|  |  |
| --- | --- |
| **J.Aditi Patel**  **A. Bhandavi**  **Faseeh Ahmed**  **M. Samendu Reddy** | **(18BD1A052Q)**  **(18BD1A0522)**  **(18BD1A053H)**  **(18BD1A053J)** |

In partial fulfilment for the award of **Bachelor of Technology** in **Computer Science and Engineering** affiliated to the **Jawaharlal Nehru Technological University, Hyderabad** during the year 2021-22**.**

**Internal Guide Head of the Department**

**(Mrs. Chinnam Madhuri) (Dr. S. Padmaja)**

**Submitted for Viva Voce Examination held on**

**External Examiner**

***Unit of Keshav Memorial Educational Society***

#: 3-5-1026 Narayanaguda Hyderabad 500029.

 040-3261407  [www.kmit.in](http://www.kmit.in/) e-mail: [principal@kmit.in](mailto:principal@kmit.in)

**Vision of KMIT**

Producing quality graduates trained in the latest technologies and related tools and striving to make India a world leader in software and hardware products and services. To achieve academic excellence by imparting in depth knowledge to the students, facilitating research activities and catering to the fast growing and ever- changing industrial demands and societal needs.

**Mission of KMIT**

* To provide a learning environment that inculcates problem solving skills, professional, ethical responsibilities, lifelong learning through multi modal platforms and prepare students to become successful professionals.
* To establish industry institute Interaction to make students ready for the industry.
* To provide exposure to students on latest hardware and software tools.
* To promote research based projects/activities in the emerging areas of technology convergence.
* To encourage and enable students to not merely seek jobs from the industry but also to create new enterprises.
* To induce a spirit of nationalism which will enable the student to develop, understand lndia's challenges and to encourage them to develop effective solutions.
* To support the faculty to accelerate their learning curve to deliver excellent service to students.

**Vision & Mission of CSE**

**Vision of the CSE**

To be among the region's premier teaching and research Computer Science and Engineering  departments producing globally competent and socially responsible graduates in the most conducive  academic environment.

**Mission of the CSE**

* To provide faculty with state of the art facilities for continuous professional development and research, both in foundational aspects and of relevance to emerging computing trends.
* To impart skills that transform students to develop technical solutions for societal needs and inculcate entrepreneurial talents.
* To inculcate an ability in students to pursue the advancement of knowledge in various specializations of Computer Science and Engineering and make them industry-ready.
* To engage in collaborative research with academia and industry and generate adequate resources for research activities for seamless transfer of knowledge resulting in sponsored projects and consultancy.
* To cultivate responsibility through sharing of knowledge and innovative computing solutions that benefit the society-at-large.
* To collaborate with academia, industry and community to set high standards in academic excellence and in fulfilling societal responsibilities.

**PROGRAM OUTCOMES (POs)**

**1. Engineering Knowledge:** Apply the knowledge of mathematics, science, engineering fundamentals and an engineering specialization to the solution of complex engineering problems.

**2. Problem analysis:** Identify formulate, review research literature, and analyse complex engineering problems reaching substantiated conclusions using first principles of mathematics, natural sciences, and engineering sciences

**3. Design/development of solutions:** Design solutions for complex engineering problem and design system component or processes that meet the specified needs with appropriate consideration for the public health and safety, and the cultural societal, and environmental considerations.

**4. Conduct investigations of complex problems:** Use research-based knowledge and research methods including design of experiments, analysis and interpretation of data, and synthesis of the information to provide valid conclusions.

**5. Modern tool usage:** Create select, and, apply appropriate techniques, resources, and modern engineering and IT tools including prediction and modelling to complex engineering activities with an understanding of the limitations.

**6. The engineer and society:** Apply reasoning informed by the contextual knowledge to societal, health, safety. legal und cultural issues and the consequent responsibilities relevant to professional engineering practice.

**7. Environment and sustainability:** Understand the impact of the professional engineering solutions in societal and environmental contexts and demonstrate the knowledge of, and need for sustainable development.

**8. Ethics:** Apply ethical principles and commit to professional ethics and responsibilities and norms of the engineering practice.

**9. Individual and team work:** Function effectively as an individual, and as a member or leader in diverse teams and in multidisciplinary settings.

**10. Communication:** Communicate effectively on complex engineering activities with the engineering community and with society at large, such as, being able to comprehend and write effective reports and design documentation make effective presentations, and give and receive clear instructions.

**11. Project management and finance:** Demonstrate knowledge and understanding of the engineering and management principles and apply these to one's own work, as a member and leader in a team, to manage projects and in multidisciplinary environments.

**12. Life-long learning:** Recognize the need for, and have the preparation and ability to engage in independent and life-long learning in the broadest context of technological change.

**PROGRAM SPECIFIC OUTCOMES (PSOs)**

**PSO1:** An ability to analyze the common business functions to design and develop appropriate Computer Science solutions for social upliftment’s.

**PSO2:** Shall have expertise on the evolving technologies like Mobile Apps, CRM, ERP, Big Data, etc.

**PROGRAM EDUCATIONAL OBJECTIVES (PEOs)**

**PEO1:** Graduates will have successful careers in computer related engineering fields or will be able to successfully pursue advanced higher education degrees.

**PEO2:** Graduates will try and provide solutions to challenging problems in their profession by applying computer engineering principles.

**PEO3:** Graduates will engage in life-long learning and professional development by rapidly adapting changing work environment.

**PEO4:** Graduates will communicate effectively, work collaboratively and exhibit high levels of professionalism and ethical responsibility.

**PROJECT OUTCOMES**

**P1: To provide a friendly environment to the user**

**P2: Accurate results of statistics are provided**

**P3: Emergency contact info is provided with respect to the location of the user**

**P4: Government and press releases are delivered**

**LOW - 1**

**MEDIUM - 2**

**HIGH - 3**

|  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **PO** | **PO1** | **PO2** | **PO3** | **PO4** | **PO5** | **PO6** | **PO7** | **PO8** | **PO9** | **PO10** | **PO11** | **PO12** |
| **P1** |  |  | 3 |  | 3 | 3 |  |  | 2 |  |  | 2 |
| **P2** | 2 | 2 |  | 1 | 2 |  |  |  |  |  |  | 2 |
| **P3** |  |  |  |  | 2 |  |  | 1 |  | 2 |  | 1 |
| **P4** |  |  |  | 1 | 2 | 1 |  |  |  | 1 |  | 1 |

**PROJECT OUTCOMES MAPPING PROGRAM OUTCOMES**

**PROJECT OUTCOMES MAPPING PROGRAM SPECIFIC OUTCOMES**

|  |  |  |
| --- | --- | --- |
| **PSO** | **PSO1** | **PSO2** |
| **P1** | 1 | 3 |
| **P2** |  | 3 |
| **P3** |  | 3 |
| **P4** |  | 3 |

**PROJECT OUTCOMES MAPPING PROGRAM EDUCATIONALOBJECTIVES**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **PEO** | **PEO1** | **PEO2** | **PEO3** | **PEO4** |
| **P1** | 1 |  |  | 2 |
| **P2** | 1 |  |  | 2 |
| **P3** | 1 |  |  | 2 |
| **P4** | 1 |  |  | 2 |

**DECLARATION**

We hereby declare that the project report entitled “**COVID19 BOT”** is done in the partial fulfillment for the award of the Degree in Bachelor of Technology in Computer Science and Engineering affiliated to Jawaharlal Nehru Technological University, Hyderabad. This project has not been submitted anywhere else.

### J.ADITI PATEL (18BD1A052Q)

**A.BHANDAVI (18BD1A0522)**

**FASEEH AHMED (18BD1A053H)**

**M.SAMENDU REDDY(18BD1A053J)**

**ACKNOWLEDGMENT**

We take this opportunity to thank all the people who have rendered their full support to our project work.

We render our thanks to **Dr. Maheshwar Dutta**, B.E., M Tech., Ph.D., Principal who encouraged us to do the Project.

We are grateful to **Mr. Neil Gogte**, Director for facilitating all the amenities required for carrying out this project.

We express our sincere gratitude to **Mr. S. Nitin**, Director and **Mrs. Anuradha**, Dean Academics for providing an excellent environment in the college.

We are also thankful to **Dr. S. Padmaja**, Head of the Department for providing us with both time and amenities to make this project a success within the given schedule.

We are also thankful to our guide **Chinnam Madhuri**, for his/her valuable guidance and encouragement given to us throughout the project work.

We would like to thank the entire CSE Department faculty, who helped us directly and indirectly in the completion of the project. We sincerely thank our friends and family for their constant motivation during the project work.

### J.ADITI PATEL(18BD1A052Q)

**A.BHANDAVI(18BD1A0522)**

**FASEEH AHMED(18BD1A053H)**

**M.SAMENDU REDDY(18BD1A053J)**

# **CONTENT**

|  |  |
| --- | --- |
| **DESCRIPTION** | **PAGE NO.** |
| **ABSTRACT** | **i** |
| **LIST OF FIGURES** | **ii** |
| **LIST OF TABLES** | **iii** |
| **CHAPTERS** |  |
| **1. INTRODUCTION** | **1-6** |
| 1.1. Purpose of the Project | 2 |
| 1.2. Problem with the Existing System | 2 |
| 1.3. Proposed System | 2 |
| 1.4. Scope of Project | 3 |
| 1.5. Design Architecture | 4 |
| 1.6. Technologies used | 4 |
| **2. SOFTWARE REQUIREMENTS SPECIFICATIONS** | **7-11** |
| 2.1. Requirements Specification Document | 8 |
| 2.2. Functional Requirements | 9 |
| 2.3. Non-Functional Requirements | 9 |
| 2.4. Software Requirements | 10 |
| 2.5. Hardware Requirements | 11 |
| **3. LITERATURE SURVEY** | **12-14** |
| **4. SYSTEM DESIGN** | **15-21** |
| 4.1. Introduction to UML | 16 |
| 4.2. UML Diagrams | 16 |
| 4.2.1 Use Case diagram | 16 |
| 4.2.2 Sequence diagram | 18 |
| 4.2.3 Activity diagram | 20 |
| **5. IMPLEMENTATION** | **22-26** |
| 5.1. Code Snippets | 23 |
| **6. TESTING** | **27-34** |
| 6.1. Introduction to Testing | 28 |
| 6.2. Test Cases | 30 |
| **7. SCREENSHOTS** | **35-40** |
| **8.FURTHER ENHANCEMENTS** | **41** |
| **9.CONCLUSION** | **43** |
| **10.REFERENCES** | **45** |

**ABSTRACT**

# Chatbot is a computer program designed to simulate conversation with human users, especially over the internet. Simple chatbots work based on pre-written keywords that they understand, each of these commands must be written by the developer separately using regular expressions or other form of string analysis, if the user has asked a question without using a single keyword the bot cannot understand it and as a rule responds with default messages. Smart chatbots rely on artificial intelligence when they communicate with the users instead of pre-prepared answers the bot responds with adequate suggestions on the topic. A COVID-19 bot helps the user to get answer related all the queries/FAQ Related to Covid-19like Help Desk,Live News,Govt Announcements,Images and Videos related to Covid-19 etc.

# **LIST OF FIGURES**

|  |  |  |
| --- | --- | --- |
|  | **LIST OF FIGURES** | PAGE NO |
| 1.4 | Chatbot | 3 |
| 1.5 | Architecture of Chatbot | 4 |
| 4.2.1 | Use Case Diagram | 17 |
| 4.2.2 | Sequence Diagram | 19 |
| 4.2.3 | Activity Diagram | 21 |
| 7.1 | Working of Chatbot in telegram | 36 |
| 7.2 | Desktop View of Chatbot | 37 |
| 7.3 | Code for API request | 38 |

**LIST OF TABLES**

|  |  |  |
| --- | --- | --- |
| **S. No** | **LIST OF TABLES** | **PAGE NO** |
| 6.2.1 | Test Case 1 | 30 |
| 6.2.2 | Test Case 2 | 31 |
| 6.2.3 | Test Case 3 | 32 |
| 6.2.4 | Test Case 4 | 33 |
| 6.2.5 | Test Case 5 | 34 |

**CHAPTER -1**

**INTRODUCTION**

## 1.1 Purpose of Project

A Chatbot is a computer program that tries to conduct a natural language conversation with a human user. It lets the user ask normal questions and statements such as “How are you today?” and “How is the weather today?” and then responds in a similar natural manner . The ultimate aim for chatbots is when the user cannot tell the difference between talking to a chatbot and talking to a real human. Chatbots have been on the rise ever since machine learning and NLP have gained prominence. They have managed to penetrate into various field like healthcare, stock, customer care, entertainment and banking. NLP helps the computer develop a understanding of human behavior so that it can create a proper communication channel for the user.

Common people came together during hard times and to play a part in the world’s fight against covid they developed chatbots which were enabled on Telegram,WhatsApp for use for general public.A text is easier to make than call nowadays and during busy hours it is far more efficient as you can simultaneously communicate with multiple people and provide accurate and fast responses.

## 1.2 Problems with Existing System

Exsisting Covid19 related chatbot are not updated on a regular basis even though there main purpose is to provide statistics.Most of the chatbots relating to covid are web based and fail to provide a friendy environment for such a sensitive converstion to take place.Nearby facilties are provided on the basis of pincode rather than current location.

**1.3 Proposed System**

We are providing the user with current statistics by Country,State,City and district.Government Press releases regarding Corona are provided via the live news feature in the proposed chatbot.User is provided with Emergency helpline numbers based on city they reside in. Hospitals and pharmacy stores are provided on the basis of the user’s current location provided by the user.

## 1.4 Scope of the Project

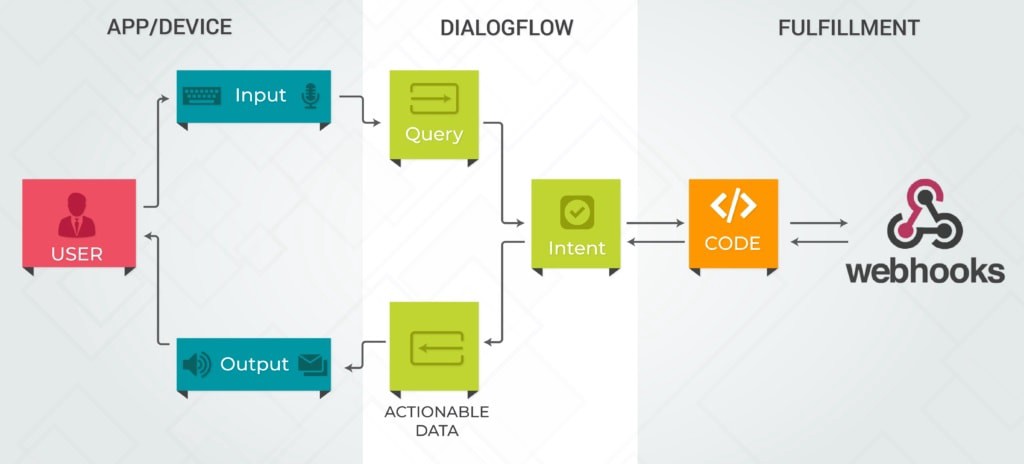
A Covid19 chat bot is going provide a person with information regarding the number of covid cases based on the area input provided and if he/she needs to be vaccinated a suitable appointment is shown depending on vaccine choice provided by the user. Additionally provides information regarding helpful resources like plasma and oxygen.

Chatbots during Covid where mostly area oriented and were mainly made in language widely used in that area, which is an another benefit provided by NLP as English is not the only language it supports.The outcome of this project is to provide a chatbot which would help with the harsh conditions provided by COVID19.

## 1_CFGHV5Kpme5hliFR3ZbGrA

**Fig 1.4 –Chatbot**

## 1.5 Architecture Diagram

****

**Fig 1.5 – Architecture of Chatbot**

Once the Dialogflow agent receives the user’s request, it will try to map it to a list of pre-defined intents. Intents are created by the bot developer as a way of telling the Dialogflow service “here is a list of tasks that the user might ask the chatbot to do”.The structured information is sent to some code running on some web host which is maintained by the bot creator – this code is called a webhook. The webhook performs the necessary business logic.It will return a response to your Dialogflow agent in a specific format.Dialogflow will parse the response, extract relevant information and use it in the response sent to the user.

**1.6.Technologies Used :**

### 1.6.1 Dialog flow

Dialogflow is a natural language understanding platform that makes it easy to design and integrate a conversational user interface into your mobile app, web application, device, bot, interactive voice response system, and so on. Using Dialogflow, you can provide new and engaging ways for users to interact with your product.

Dialogflow can analyze multiple types of input from your customers, including text or audio inputs (like from a phone or voice recording). It can also respond to your customers in a couple of ways, either through text or with synthetic speech.

### 1.6.2 Postman

Postman is an application used for API testing. It is an HTTP client that tests HTTP requests,utilizing a graphical user interface, through which we obtain different types of responses that need to be subsequently validated.

### Methods

Postman offers many endpoint interaction methods. The following are some of the most used, including their functions:

* GET: Obtain information
* POST: Add information
* PUT: Replace information
* PATCH: Update certain information
* DELETE: Delete information

### 1.6.3 RapidAPI

RapidAPI Testing is a cloud-based API testing solution that enables enterprises to create and manage comprehensive API tests from development through deployment.

It supports any API type including REST, SOAP, and GraphQL, and offers an intuitive UX that simplifies testing, monitoring, management, and integration across the development lifecycle.

RapidAPI Testing automatically integrates with your APIs on RapidAPI’s API Marketplace / Enterprise Hub, creating one central location for all API tests and eliminating the need to manually add APIs for testing.

### 1.6.4 Framework

**1.6.4.1 Flask**

Flask is a web framework, it’s a Python module that lets you develop web applications easily. It’s has a small and easy-to-extend core: it’s a microframework that doesn’t include an ORM (Object Relational Manager) or such features.It does have many cool features like url routing, template engine. It is a WSGI web app framework.

**The layout of the Python Flask Framework**

* **Module Init** - (project\_root/app\_name/admin/\_\_init\_\_.py) - required to enable the app
* **Module URL** - (project\_root/app\_name/admin/url.py) - Url definitions of each module
* **App root Init** - (project\_root/app\_name/\_\_init\_\_.py) - Not necessary to define the entire app within \_\_init\_\_.py
* **Module Views** - (project\_root/app\_ame/admin/views.py) - Defines views for each module. Separate ‘.py.’ Files as the project scale to ensure they are accessible to URLs.
* **Module Templates** - (project\_root/app\_name/admin/templates/admin/main.html) - Normal template folder.

**CHAPTER -2**

# **2.SYSTEM RERUIREMENT SPECIFICATIONS**

## 2.1 What is SRS?

Software Requirement Specification (SRS) is the starting point of the software developing activity. As system grew more complex it became evident that the goal of the entire system cannot be easily comprehended. Hence the need for the requirement phase arose. The software project is initiated by the client needs. The SRS is the means of translating the ideas of the minds of clients (the input) into a formal document (the output of the requirement phase.)

The SRS phase consists of two basic activities:

### Problem/Requirement Analysis:

The process is order and more nebulous of the two, deals with understand the problem, the goal and constraints.

### Requirement Specification:

Here, the focus is on specifying what has been found giving analysis such as representation, specification languages and tools, and checking the specifications are addressed during this activity.The Requirement phase terminates with the production of the validate SRS document. Producing the SRS document is the basic goal of this phase.

## 2.2 Role of SRS

The purpose of the Software Requirement Specification is to reduce the communication gap between the clients and the developers. Software Requirement Specification is the medium though which the client and user needs are accurately specified. It forms the basis of software development. A good SRS should satisfy all the parties involved in the system.

## 2.3 Requirements Specification Document

A Software Requirements Specification (SRS) is a document that describes the nature of a project, software or application. In simple words, SRS document is a manual of a project provided it is prepared before you kick-start a project/application. This document is also known by the names SRS report, software document. A software document is primarily prepared for a project, software or any kind of application.

There are a set of guidelines to be followed while preparing the software requirement specification document. This includes the purpose, scope, functional and non functional requirements, software and hardware requirements of the project. In addition to this, it also contains the information about environmental conditions required, safety and security requirements, software quality attributes of the project etc.

The purpose of SRS (Software Requirement Specification) document is to describe the external behaviour of the application developed or software. It defines the operations, performance and interfaces and quality assurance requirement of the application or software. The complete software requirements for the system are captured by the SRS.This section introduces the requirement specification document for Word Building Game using Alexa which enlists functional as well as non-functional requirements.

## 2.4 Functional Requirements

For documenting the functional requirements, the set of functionalities supported by the system are to be specified. A function can be specified by identifying the state at which data is to be input to the system, its input data domain, the output domain, and the type of processing to be carried on the input data to obtain the output data. Functional requirements define specific behaviour or function of the application. Following are the functional requirements:

FR1) The reply should be realistic like speaking to a human.

FR2) Should be interactive, that is it should prompt the user what it should do next.

FR3) The Proposed system should be capable to understand the query faced by the user and provide the appropriate answer.

FR4) The answers should prompt and if a invalid input is provided the user should be informed.

## 2.5 Non-Functional Requirements

A non-functional requirement is a requirement that specifies criteria that can be used to judge the operation of a system, rather than specific behaviours. Especially these are the constraints the system must work within. Following are the non-functional requirements:

NFR 1) The chatbot must be efficient with very little lag in response time for instance no longer than 5 seconds to reply to a user message.

NFR 2) The chatbot must be reliable with next to no faults or bugs.

NFR 3) The chatbot must be secure as sensitive data is being used.

NFR 4) The use of natural language used to interact with the chatbot promotes human computer interaction.

NFR 5) Provide accurate responses to input.

NFR 6) Appropriate handling of unexpected input and correctly inform the user if it cannot provide assistance .

## Performance:

The performance of the developed applications can be calculated by using following methods: Measuring enables you to identify how the performance of your application stands in relation to your defined performance goals and helps you to identify the bottlenecks that affect your application performance. It helps you identify whether your application is moving toward or away from your performance goals. Defining what you will measure, that is, your metrics, and defining the objectives for each metric is a critical part of your testing plan.

Performance objectives include the following:

Response time, Latency throughput or Resource utilization.

## 2.6 Software Requirements

Operating System : Windows 10 or MAC OS.

Platform : PyCharm,Telegram

Eclipse Programming Language : Java, JSP, Servlets, MySQL,MongoDB

## 2.7 Hardware Requirements

Processor : Intel core i5 or Above, AMD Ryzen 5 or Above

Hard Disk : 2 GB or above.

RAM : 2 GB or above.

Internet : 4 Mbps or above (Wireless).

GPU : Intel Integrated Graphics or higher

**CHAPTER -3**

# **LITERATURE SURVEY**

**1.)“Recruitment Chatbots”, International Research Journal of Engineering and Technology (IRJET), vol. 5, Issue: 08, Aug 2018[1].**

**Authors: Akash Balachandar, Anusha D Kulkarni**

In this paper, authors have explained how the chatbot behaving as a human conversational partner are designed to comprehend a conclusive human response. In today’s world, it is difficult to collect correct information easily while hiring the right candidate. Using simply a chatbot can be a solution to this problem. Recruiters can use this in day-to-day life to automate time-consuming tasks.

**SYSTEM DESIGN**

Describing the designing process of interaction between the chatbot and the user. It uses dialogues systems, and they are of two types:

1. Goal Oriented Dialogue Systems.

2) General conversation Dialogue Systems.

We use Generative and Selective approaches in recruitment chatbot which needs a general conversational dialog system. The Machine Learning principle is a core philosophy for both these approaches: Build it, Train it, and Test it. By using bot characteristics, constraints, dialogue dataset, access flow, and Sequence tokens this model is built.

**2.) “Classification Technique of Interviewer-Bot Result using Naïve Bayes an Phrase Reinforcement Algorithms,” International Journal of Emerging Technologies in Learning (iJET), 13(02), 33-47, 2018[2].**

Authors: Sarosa, M., Junus, M., Hoesny, M. U., Sari, Z., & Fatnuriyah, M. In this paper authors have classified the outcomes of a job interview among the the interviewer-bot and user by using Naïve Bayes algorithm.

Advantages of Naive Bayes Algorithm 

* It is easy to understand and implement. 
* Any kind of complex optimization is not required. 
* It is easily updated if new training data is received. . 
* Sometimes independence assumption may seem unreasonable, but its performance is usually good.

Advantages of Phrase Reinforcement (PR) Algorithm 

* Maximizes Performance. 
* For a long period of time changes can be sustained.

1. **) “Task-based Interaction Chatbot”, EEE521 final year project Report school of computing, Engineering & Intelligent System[4].**

**Authors: Dr. Kevin Curran, Dr. Daniel Kelly**

Chatbot Architecture It comprises of four parts first is front-end second is knowledge-base third is back-end and corpus which are training data. The communication with the user is done on front end part. NLU (natural language understanding) is used to understand the context and intent of the user input. An appropriate response is generated from user. The knowledge base determines the chatbots knowledge, which is done with the NLU and supported at the back-end. The back-end produces the knowledge base by making use of the domains corpus. Input is given to the chatbot in the form of speech or text. The input is given to the dialog management system which defines an appropriate response and asks the chatbots to perform the required action. The responses are produced in the form of text and speech both.

**CHAPTER -4**

## 4.SYSTEM DESIGN

## 4.1 Introduction to UML

The Unified Modeling Language allows the software engineer to express an analysis model using the modeling notation that is governed by a set of syntactic, semantic and pragmatic rules. A UML system is represented using five different views that describe the system from distinctly different perspective. Each view is defined by a set of diagram, which is as follows:

1. User Model View

This view represents the system from the users’ perspective. The analysis representation describes a usage scenario from the end-users’ perspective.

1. Structural Model View

In this model, the data and functionality are arrived from inside the system. This model view models the static structures.

1. Behavioural Model View

It represents the dynamic of behavioural as parts of the system, depicting he interactions of collection between various structural elements described in the user model and structural model view.

1. Implementation Model View

In this view, the structural and behavioural as parts of the system are represented as they are to be built.

1. Environmental Model View

In this view, the structural and behavioural aspects of the environment in which the system is to be implemented are represented.

**4.2 UML Diagrams**

**4.2.1 Use Case Diagram**

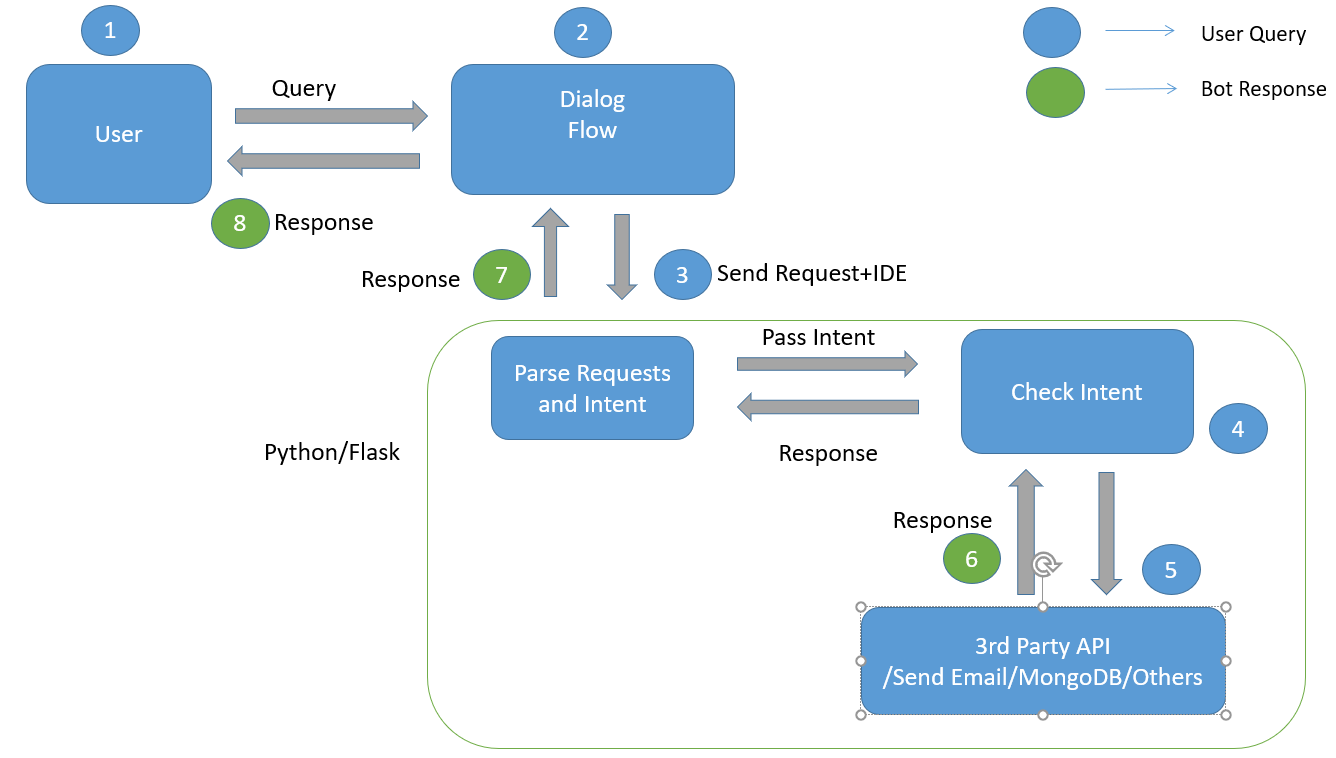
To model a system, the most important aspect is to capture the dynamic behaviour. To clarify a bit in details, dynamic behaviour means the behaviour of the system when it is running/operating.

So only static behaviour is not sufficient to model a system rather dynamic behaviour is more important than static behaviour. In UML there are five diagrams available to model dynamic nature and use case diagram is one of them. Now as we have to discuss that the use case diagram is dynamic in nature there should be some internal or external factors for making the interaction.

These internal and external agents are known as actors. So use case diagrams are consisting of actors, use cases and their relationships. The diagram is used to model the system/subsystem of an application. A single use case diagram captures a particular functionality of a system. So to model the entire system numbers of use case diagrams are used.

Use case diagrams are used to gather the requirements of a system including internal and external influences. These requirements are mostly design requirements. So when a system is analysed to gather its functionalities use cases are prepared and actors are identified. In brief, the purposes of use case diagrams can be as follows:

* 1. Used to gather requirements of a system.
  2. Used to get an outside view of a system.
  3. Identify external and internal factors influencing the system.
  4. Show the interacting among the requirements are actors.



**Fig 4.2.1 – Use Case Diagram**

## 4.2.2 Sequence Diagram

Sequence diagrams describe interactions among classes in terms of an exchange of messages over time. They're also called event diagrams. A sequence diagram is a good way to visualize and validate various runtime scenarios. These can help to predict how a system will behave and to discover responsibilities a class may need to have in the process of modelling a new system.

The aim of a sequence diagram is to define event sequences, which would have a desired outcome. The focus is more on the order in which messages occur than on the message per se. However, the majority of sequence diagrams will communicate what messages are sent and the order in which they tend to occur.

**Basic Sequence Diagram Notations**

**Class Roles or Participants**

Class roles describe the way an object will behave in context. Use the UML object symbol to illustrate class roles, but don't list object attributes.

**Activation or Execution Occurrence**

Activation boxes represent the time an object needs to complete a task. When an object is busy executing a process or waiting for a reply message, use a thin grey rectangle placed vertically on its lifeline.

**Messages**

Messages are arrows that represent communication between objects. Use half-arrowed lines to represent asynchronous messages.

Asynchronous messages are sent from an object that will not wait for a response from the receiver before continuing its tasks.

**Lifelines**

Lifelines are vertical dashed lines that indicate the object's presence over time.

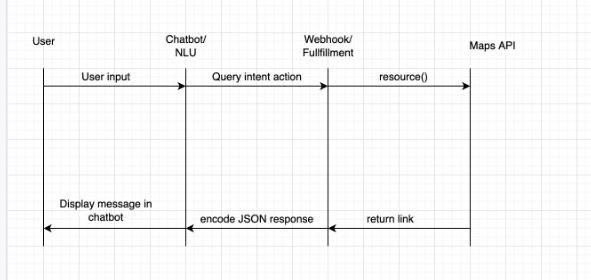
**Destroying Objects**

Objects can be terminated early using an arrow labelled "<< destroy >>" that points to an X. This object is removed from memory. When that object's lifeline ends, you can place an X at the end of its lifeline to denote a destruction occurrence.

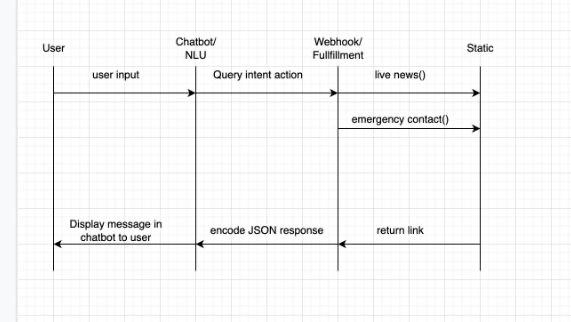
**Loops**

A repetition or loop within a sequence diagram is depicted as a rectangle. Place the condition for exiting the loop at the bottom left corner in square brackets []. Guards

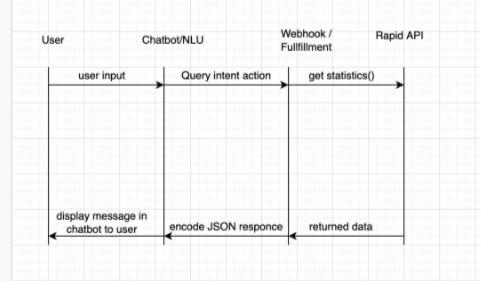
When modelling object interactions, there will be times when a condition must be met for a message to be sent to an object. Guards are conditions that need to be used throughout UML diagrams to control flow.



**Fig 4.2.2.1-Sequence diagram**



**Fig 4.2.2.2-Sequence diagram**

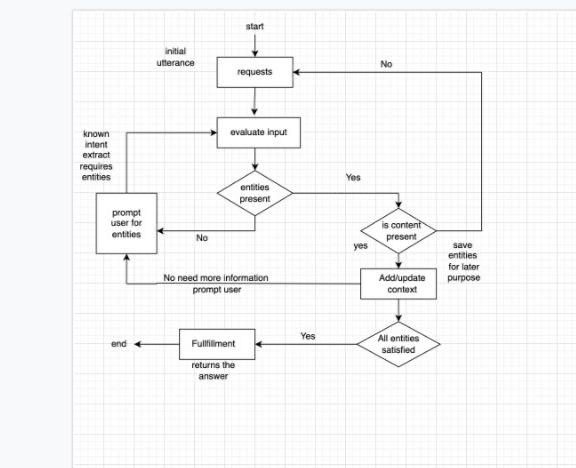


**Fig 4.2.2.3 – Sequence Diagram**

## 4.2.3 Acitvity Diagram

Activity Diagrams describe how activities are coordinated to provide a service which can be at different levels of abstraction. Typically, an event needs to be achieved by some operations, particularly where the operation is intended to achieve a number of different things that require coordination, or how the events in a single use case relate to one another, in particular, use cases where activities may overlap and require coordination. It is also suitable for modeling how a collection of use cases coordinate to represent business workflows

1. Identify candidate use cases, through the examination of business workflows
2. Identify pre- and post-conditions (the context) for use cases
3. Model workflows between/within use cases
4. Model complex workflows in operations on objects
5. Model in detail complex activities in a high level activity Diagram



**Fig 4.2.3 – Activity Diagram**

**CHAPTER -5**

# **IMPLEMENTATION**

## 5.1 Code Snippets

#''

@app.route('/webhook', methods=['POST'])

@cross\_origin()

def webhook():

req = request.get\_json(silent=True, force=True)

res = processRequest(req)

res = json.dumps(res, indent=4)

print(res)

r = make\_response(res)

r.headers['Content-Type'] = 'application/json'

return r

# processing the request from dialogflow

def processRequest(req):

# dbConn = pymongo.MongoClient("mongodb://localhost:27017/") # opening a connection to Mongo

#log = Conversations.Log()

sessionID = req.get('responseId')

result = req.get("queryResult")

intent = result.get("intent").get('displayName')

query\_text = result.get("queryText")

parameters = result.get("parameters")

cust\_name = parameters.get("cust\_name")

cust\_contact = parameters.get("cust\_contact")

cust\_email = parameters.get("cust\_email")

#db = configureDataBase()

if intent == 'covid\_searchcountry':

cust\_country = parameters.get("geo-country")

if(cust\_country=="United States"):

cust\_country = "USA"

fulfillmentText, deaths\_data, testsdone\_data = makeAPIRequest(cust\_country)

webhookresponse = "\*\*Covid Report\*\* \n\n" + " New cases :" + str(fulfillmentText.get('new')) + \

"\n" + " Active cases : " + str(

fulfillmentText.get('active')) + "\n" + " Critical cases : " + str(fulfillmentText.get('critical')) + \

"\n" + " Recovered cases : " + str(

fulfillmentText.get('recovered')) + "\n" + " Total cases : " + str(fulfillmentText.get('total')) + \

"\n" + " Total Deaths : " + str(deaths\_data.get('total')) + "\n" + " New Deaths : " + str(

deaths\_data.get('new')) + \

"\n" + " Total Test Done : " + str(deaths\_data.get('total')) + "\n\n\*\*\*\*END\*\*\*\* \n "

print(webhookresponse)

#log.saveConversations(sessionID, cust\_country, webhookresponse, intent, db)

#log.saveCases( "country", fulfillmentText, db)

return {

"fulfillmentMessages": [

{

"text": {

"text": [

webhookresponse

]

}

},

{

"text": {

"text": [

"Do you want me to send the detailed report to your e-mail address? Type.. \n 1. Sure \n 2. Not now "

# "We have sent the detailed report of {} Covid-19 to your given mail address.Do you have any other Query?".format(cust\_country)

]

}

}

]

}

**API REQUEST**

import requests

import json

class Api:

def \_init\_(self):

pass

def makeApiRequestForCounrty(self, country\_name):

url = "https://covid-193.p.rapidapi.com/statistics"

querystring = {"country": country\_name}

headers = {

'x-rapidapi-host': "covid-193.p.rapidapi.com",

'x-rapidapi-key': "a642464e88msh49d25f5f2bedc8bp159e46jsn91cfe8734c45"

}

response = requests.request("GET", url, headers=headers, params=querystring)

print(response.text)

js = json.loads(response.text)

print("\*\*", js)

result = js.get('response')[0]

print(result.get('cases'))

print("\*" \* 20)

return result.get('cases'), result.get('deaths'), result.get('tests')

def makeApiWorldwide(self):

url = "https://coronavirus-map.p.rapidapi.com/v1/summary/latest"

headers = {

'x-rapidapi-host': "coronavirus-map.p.rapidapi.com",

'x-rapidapi-key': "a642464e88msh49d25f5f2bedc8bp159e46jsn91cfe8734c45"

}

response = requests.request("GET", url, headers=headers)

print(response.text)

js = json.loads(response.text)

print("\*\*", js)

result = js['data']['summary']

return result

**CHAPTER -6**

**TESTING**

## 6.1 Introduction to Testing

Testing is the process of evaluating a system or its component(s) with the intent to find whether it satisfies the specified requirements or not. Testing is executing a system in order to identify any gaps, errors, or missing requirements in contrary to the actual requirements.

According to ANSI/IEEE 1059 standard, Testing can be defined as - A process of analyzing a software item to detect the differences between existing and required conditions (that is defects/errors/bugs) and to evaluate the features of the software item.

Who does Testing?

It depends on the process and the associated stakeholders of the project(s). In the IT industry, large companies have a team with responsibilities to evaluate the developed software in context of the given requirements. Moreover, developers also conduct testing which is called Unit Testing. In most cases, the following professionals are involved in testing a system within their respective capacities:

* + Software Tester
  + Software Developer
  + Project Lead/Manager
  + End User

Levels of testing include different methodologies that can be used while conducting software testing. The main levels of software testing are:

* + Functional Testing
  + Non-functional Testing

**Functional Testing**

This is a type of black-box testing that is based on the specifications of the software that is to be tested. The application is tested by providing input and then the results are examined that need to conform to the functionality it was intended for. Functional testing

of a software is conducted on a complete, integrated system to evaluate the system's compliance with its specified requirements.

## Software Testing Life Cycle

The process of testing a software in a well planned and systematic way is known as software testing lifecycle (STLC).

Different organizations have different phases in STLC however generic Software Test Life Cycle (STLC) for waterfall development model consists of the following phases.

1. Requirements Analysis
2. Test Planning
3. Test Analysis
4. Test Design

**Requirements Analysis**

In this phase testers analyze the customer requirements and work with developers during the design phase to see which requirements are testable and how they are going to test those requirements.

It is very important to start testing activities from the requirements phase itself because the cost of fixing defect is very less if it is found in requirements phase rather than in future phases.

**Test Planning**

In this phase all the planning about testing is done like what needs to be tested, how the testing will be done, test strategy to be followed, what will be the test environment, what test methodologies will be followed, hardware and software availability, resources, risks etc. A highlevel test plan document is created which includes all the planning inputs mentioned above and circulated to the stakeholders.

**Test Analysis**

After test planning phase is over test analysis phase starts, in this phase we need to dig deeper into project and figure out what testing needs to be carried out in each SDLC phase. Automation activities are also decided in this phase, if automation needs to be done

for software product, how will the automation be done, how much time will it take to automate and which features need to be automated. Non functional testing areas (Stress and performance testing) are also analyzed and defined in this phase.

**Test Design**

In this phase various black-box and white-box test design techniques are used to design the test cases for testing, testers start writing test cases by following those design techniques, if automation testing needs to be done then automation scripts also needs to written in this phase.

## 6.2 Test Cases

## 6.2.1 Test Case 1

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Test Scenario  ID** | | Statistics | | **Test Case  ID** | | Statistics-1A | | |
| **Test Case  Description** | | Retrieve Statistics | | **Test Priority** | | High | | |
| **Pre-Requisite** | | NA | | **Post Requisite** | | NA | | |
| Test Execution Steps: | | | | | | | | |
| **S. No** | **Action** | | **Inputs** | **Expected**  **Output** | **Actual**  **Output** | **Test**  **Browser** | **Test**  **Result** | **Test**  **Comments** |
| 1 | Given a Query | | Worldwidre/  Country/  State/District | Stats based on the given input | Stats based on the  given input | Telegram | Pass | NA |

## 6.2.2 Test Case 2

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Test Scenario  ID** | | | Live News | **Test Case  ID** | | Live News-1A | | |
| **Test Case  Description** | | | Get Live News | **Test Priority** | | High | | |
| **Pre-Requisite** | | | NA | **Post Requisite** | | NA | | |
| Test Execution Steps: | | | | | | | | |
| **S. No** | **Action** | **Inputs** | | **Expected**  **Output** | **Actual**  **Output** | **Test**  **Browser** | **Test**  **Result** | **Test**  **Comments** |
| 1 | [https: //twitter.com/i /events/ 1240662](https://twitter.com/i/events/1240662)  0462804 8646?lang=en | Press LiveNews Button | | News is displayed via chatbot | News is displayed via chatbot | Telegram | Pass | NA |

## 6.2.3 Test Case 3

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Test Scenario  ID** | | Emergency Number | | **Test Case  ID** | | EmNo-1A | | |
| **Test Case  Description** | | Emergency  Number | | **Test Priority** | | High | | |
| **Pre-Requisite** | | NA | | **Post Requisite** | | NA | | |
| Test Execution Steps: | | | | | | | | |
| **S. No** | **Action** | | **Inputs** | **Expected**  **Output** | **Actual**  **Output** | **Test**  **Browser** | **Test**  **Result** | **Test**  **Comments** |
| 1 | Helpline Numbers are provided | | CityName | Emergency Numbers are provided based on  City name | Emergency Numbers are provided based on  City name | Telegram | Pass | NA |

## 6.2.3 Test Case 4

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Test Scenario  ID** | | Fallback Intent | | **Test Case  ID** | | Intent-1A | | |
| **Test Case  Description** | | Fallback Intent | | **Test Priority** | | High | | |
| **Pre-Requisite** | | NA | | **Post Requisite** | | NA | | |
| Test Execution Steps: | | | | | | | | |
| **S. No** | **Action** | | **Inputs** | **Expected**  **Output** | **Actual**  **Output** | **Test**  **Browser** | **Test**  **Result** | **Test**  **Comments** |
| 1 | Fallback Intent | | Not Suitable Input | Sorry, I didn't get that. Can you rephrase? | Sorry, I didn't get that. Can you rephrase? | Telegram | Fail | NA |

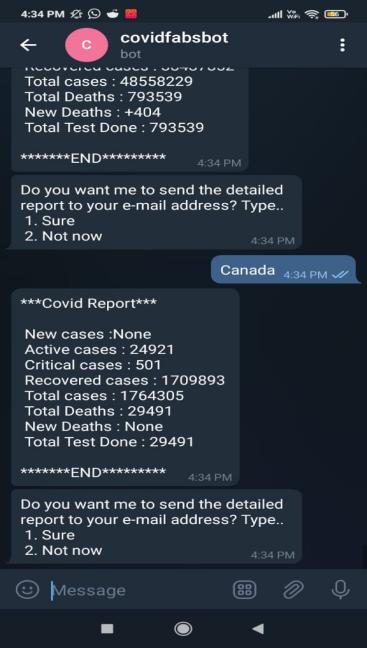
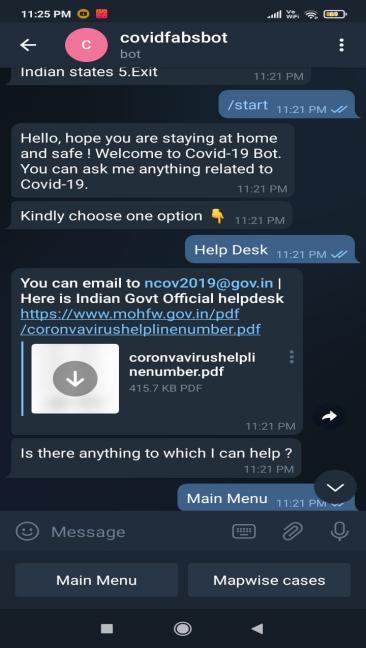
## 6.2.3 Test Case 5

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Test Scenario  ID** | | API Integration | | **Test Case  ID** | | Statistics-2A | | |
| **Test Case  Description** | | API Integration | | **Test Priority** | | High | | |
| **Pre-Requisite** | | NA | | **Post Requisite** | | NA | | |
| Test Execution Steps: | | | | | | | | |
| **S. No** | **Action** | | **Inputs** | **Expected**  **Output** | **Actual**  **Output** | **Test**  **Browser** | **Test**  **Result** | **Test**  **Comments** |
| 1 | https://covid-193.p.rapidapi.com/statistics | | Country Name | Country Statistics | Country Statistics | Telegram | Pass | NA |

**CHAPTER -7**

# **7. SCREENSHOTS**

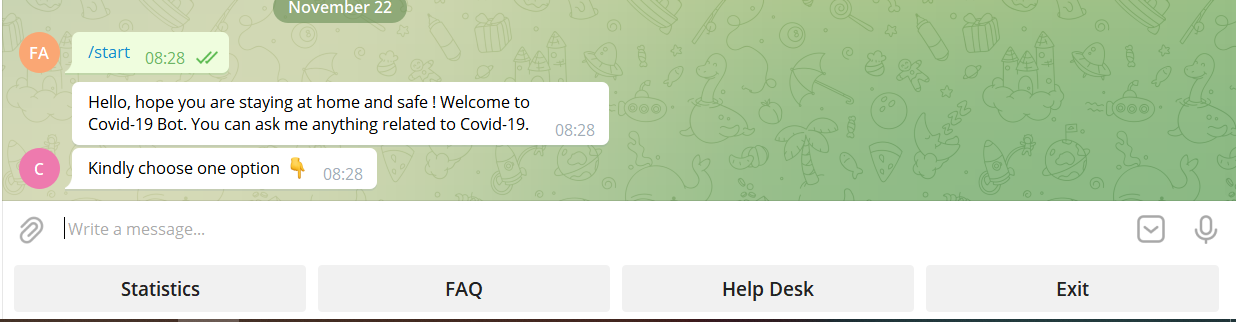
## 7.1 Mobile View of Chatbot

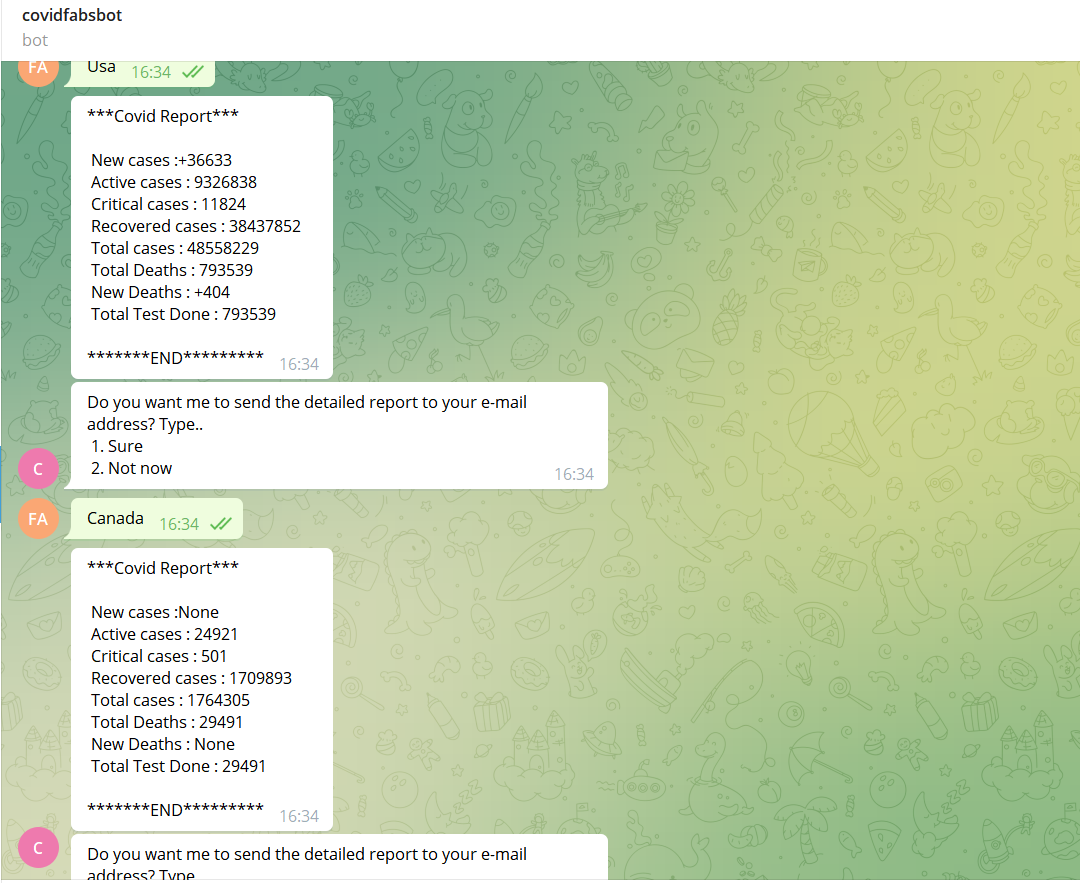
  

## govt_announcments faq cases_on_maps

**Fig 7.1-Working of chatbot in telegram**

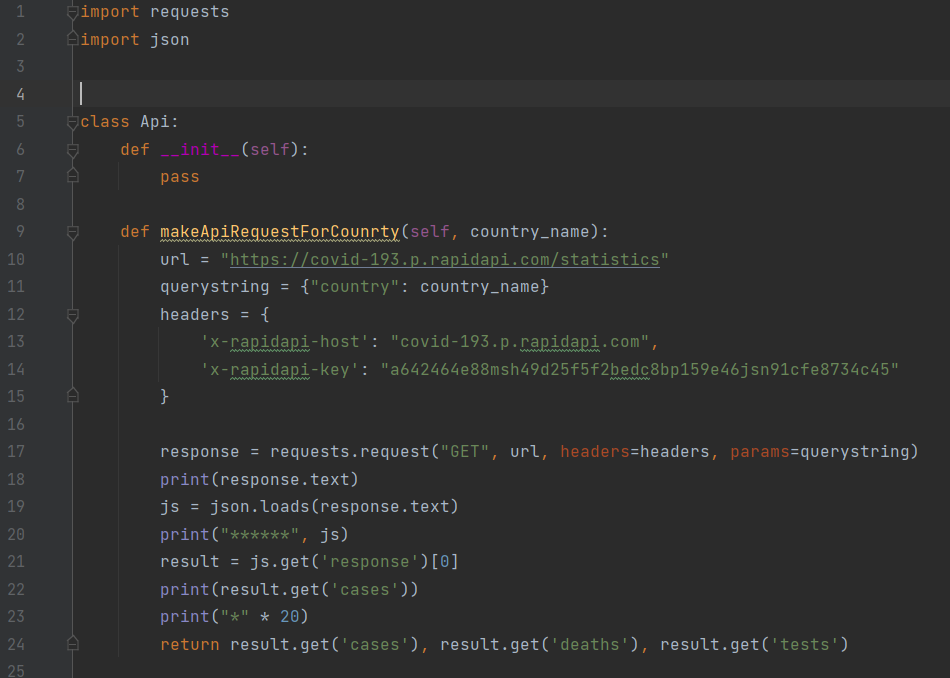
## 7.2 Desktop View of Chatbot





**Fig 7.2 –Desktop View of Chatbot (Cont.)**

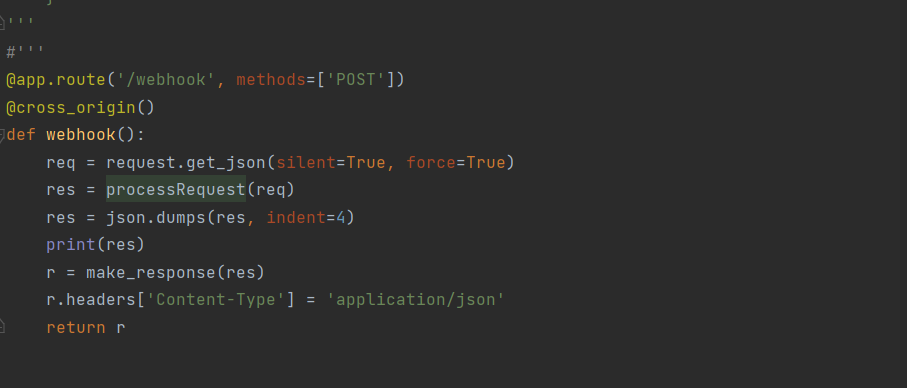
**7.3 Code snippets**

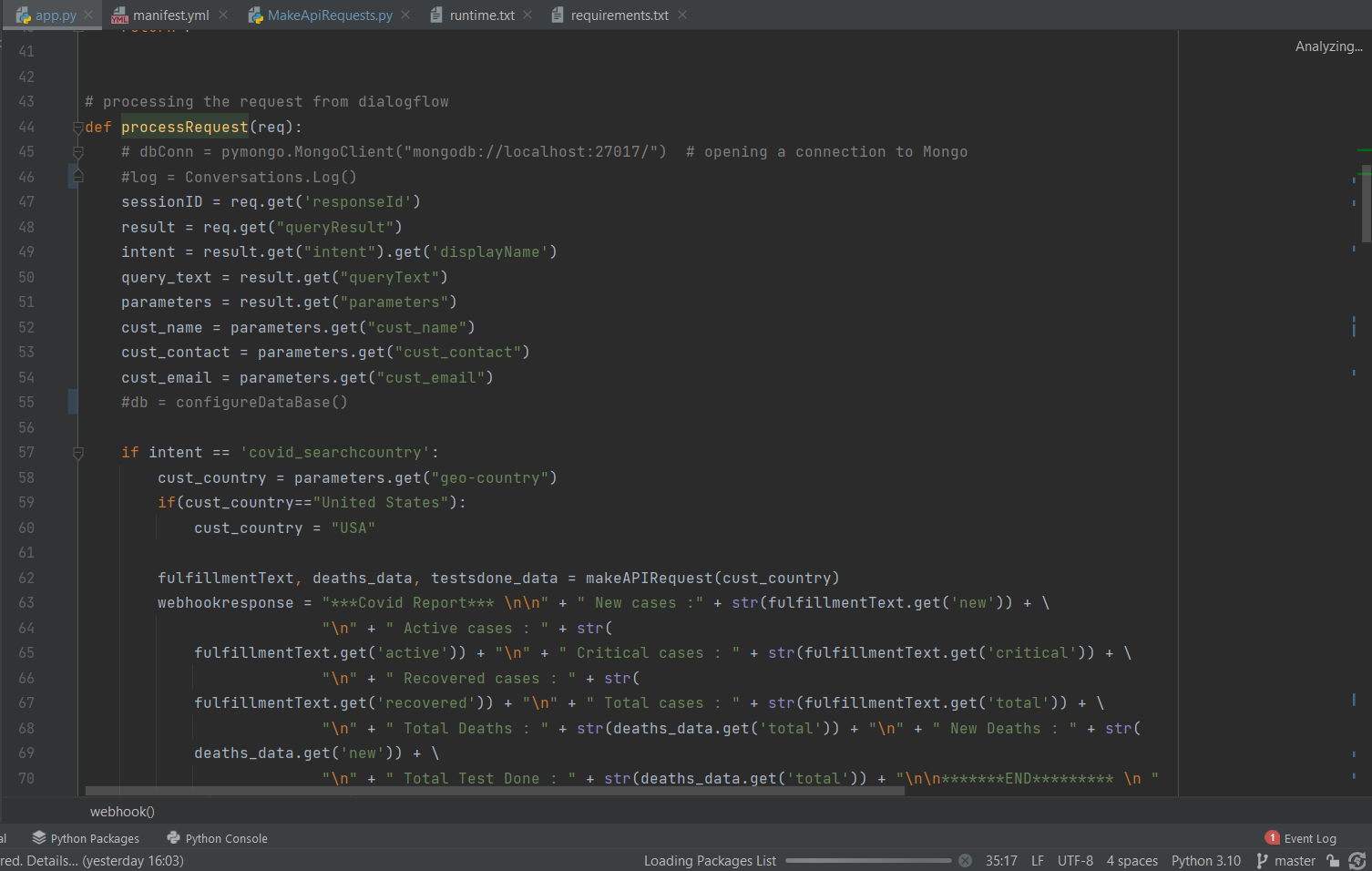




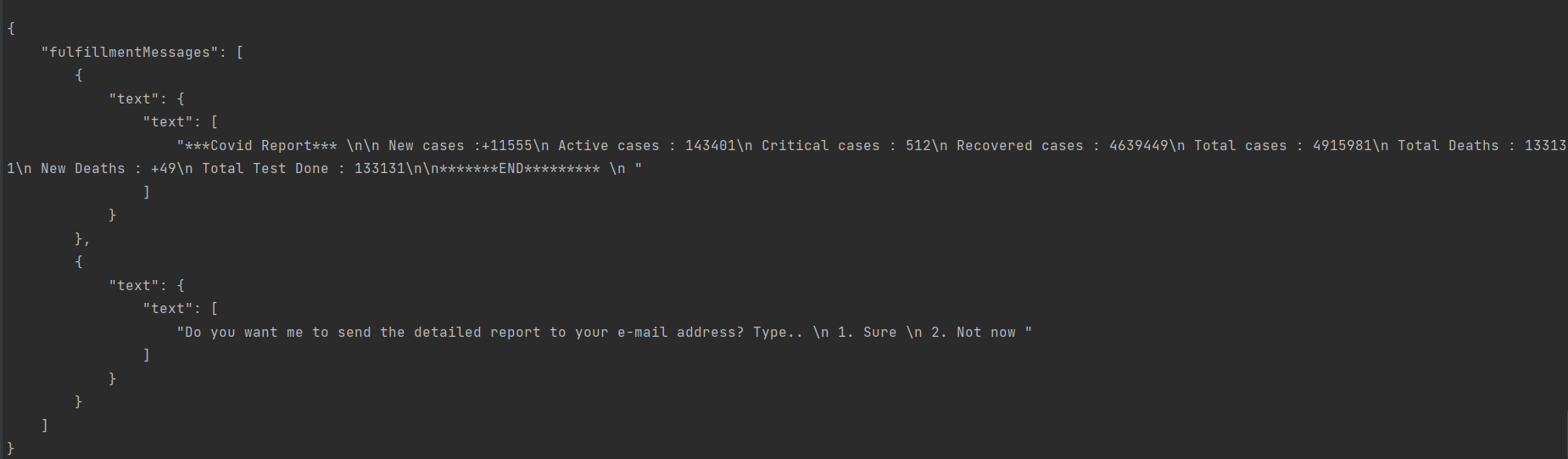
**Fig7.3-Code for API request**

**7.3 Code snippets(Cont.)**





**7.3 Code snippets(Cont.)**



**Fig 7.3-Code snippets(Cont.)**

**CHAPTER -8**

# **8.FUTURE ENHANCEMENTS**

We can include voice-based queries. The users will have to give voice input and the system will give the text output. Also, after successful execution of chatbot in health domain, we can implement it in other domains like education, forensic, sports, etc. It will be beneficial in all the fields as without spending much time, we are accessing the relevant information and that too without any sorting.

The chatbot could also be developed to be multilingual. Supporting multiple languages would help medical orgranisations in developing countries and enhance the overall accessibility of technology within the health industry whilst targeting a larger user group.

**CHAPTER -9**

# **9.CONCLUSION**

According to the scientific community, chatbots are user-friendly and any person who has an awareness of typing in their language on the desktop version and in the mobile application can use these chatbots very easily. The new development in artificial intelligence and the new wave of thinking have the potential to entirely change the experience of customers to provide the best services in such a way that echoes with the modern customers.

It is often impossible to get all the data on a single interface without the complications of going through multiple forms and windows. The covid19 vaccine chatbot aims to remove this difficulty by providing a common and user-friendly interface to solve queries of people.

The purpose of a chatbot system is to simulate a human conversation. Its architecture integrates a language model and computational algorithm to emulate information online communication between a human and a computer using natural language.The users can freely upload their queries. The chatbot provides fast and efficient search for answers to the queries and gets the relevant links to their question.

**CHAPTER -10**

**10.REFERENCES**

1. C. J. N. J. S. S. Divya Madhu, "A novel approach for medical assistance using trained chatbot," in International Conference on Inventive Communication and Computational Technologies (ICICCT), 2017.
2. S. A. N. H. Hameedullah Kazi, "Effect of Chatbot Systems on Student’s Learning Outcomes," SYLWAN, 2019.
3. Saurav Kumar Mishra, DhirendraBharti, Nidhi Mishra," Dr.Vdoc: A Medical Chatbot that Acts as a virtual Doctor", Journal of Medical Science and Technology, Volume: 6, Issue 3,2017.
4. Abdul-Kader Sameera, John Woods, "Survey on Chatbot Design Techniques in Speech Conversation Systems", (IJACSA) International Journal of Advanced Computer Science and Applications, vol. 6, no. 7, 2015.
5. S. Divya, V. Indumathi, S. Ishwarya, M. Priyasankari, S. Kalpana Devi, "A Self-Diagnosis Medical Chatbot Using Artificial Intelligence", J. Web Dev. Web Des., vol. 3, no. 1, pp. 1-7, 2018.
6. Ryuichiro Higashinaka et al. Towards an open domain conversational system fully based on natural language processing. In COLING’14, pages 928–939, 2014.
7. Bayu Setiaji, Ferry Wahyu Wibowo, "Chatbot Using A Knowledge in Database", 7th International Conference on Intelligent Systems Modelling and Simulation, 2016.
8. H. N. I/o, C. B. Lee, "Chatbots and Conversational Agents: A Bibliometric Analysis", IEEE Int. Conf. Ind. Eng. Eng. Manag., pp. 215-219, 2017.
9. Bayan Abu Shawar, Eric Atwell, "Chatbots: are they really useful?", LDVForum, 2007.