VISVESVARAYA TECHNOLOGICAL UNIVERSITY

BELAGAVI – 590018,Karnataka
INTERNSHIP REPORT

ON

"Chatbot for Healthcare System using AI"

BACHELOR OF ENGINEERING IN
Artificial Intelligence and Machine Learning

Submitted by:

NAME: SHREYAS H S

USN:1DT21AI052

Conducted by: **Compsoft Technologies**

DAYANANDA SAGAR ACADEMY OF TECHNOLOGY AND MANAGEMENT

Department of Artificial Intelligence and Machine Learning
(Accredited by NAAC with A+ Grade)
Udayapura, Kanakapura Road, Bangalore-560082

DAYANANDA SAGAR ACADEMY OF TECHNOLOGY AND MANAGEMENT
Department of Artificial Intelligence and Machine Learning
(Accredited by NAAC with A+ Grade)
Udayapura, Kanakapura Road, Bangalore-560082

CERTIFICATE

This is to certify that the Internship titled **"Chatbot for Healthcare System Using AI"** carried out by SHREYAS H S**,** a bonafide student of Dayananda Sagar Academy of Technology and Management, in partial fulfillment for the award of **Bachelor of Engineering**, in **Artificial Intelligence and Machine Learning** under Visvesvaraya Technological University, Belagavi, during the year 2022-2023. It is certified that all corrections/suggestions indicated have been incorporated in the report.

Signature of Guide **Signature of HOD** **Signature of Principal**

External Viva:

Name of the Examiner Signature with Date

1)_____

2)_____

D E C L A R A T I O N

I,SHREYAS H S, third year student of AIML, Dayananda Sagar Academy of Technology and Management - 560082, declare that the Internship has been successfully completed, in **COMPSOFT TECHNOLOGIES**. This report is submitted in partial fulfillment of the requirements for award of Bachelor Degree in AIML, during the academic year 2022-2023.

Date:04/12/2023 :

Place:Bengaluru

USN:1DT21AI052

NAME: SHREYAS H S

OFFER LETTER

Internship Offer Letter

Date: 31st October, 2023

Name: **SHREYAS H S**
USN: **1DT21AI052**
Placement ID: **23OCTMLBONE**

Dear Student,

We would like to congratulate you on being selected for the **Machine Learning with Python (Research Based)** Internship position with **Compsoft Technologies**, effective Start Date **31st October, 2023**, All of us are excited about this opportunity provided to you!

This internship is viewed as being an educational opportunity for you, rather than a part-time job. As such, your internship will include training/orientation and focus primarily on learning and developing new skills and gaining a deeper understanding of concepts of **Machine Learning with Python (Research Based)** through hands-on application of the knowledge you learn while you train with the senior developers. You will be bound to follow the rules and regulations of the company during your internship duration.

Again, congratulations and we look forward to working with you!.

Sincerely,

Nithin K. S
Project Manager
Compsoft Technologies
No. 363, 19th main road,
1st Block Rajajinagar
Bangalore - 560010

A C K N O W L E D G E M E N T

This Internship is a result of accumulated guidance, direction and support of several important persons. We take this opportunity to express our gratitude to all who have helped us to complete the Internship.

We express our sincere thanks to our Principal, for providing us adequate facilities to undertake this Internship.

We would like to thank our Head of Dept – AIML, for providing us an opportunity to carry out Internship and for his valuable guidance and support.

We would like to thank all the faculty members of our department for the support extended during the course of Internship.

We would like to thank the non-teaching members of our dept, for helping us during the Internship.

Last but not the least, we would like to thank our parents and friends without whose constant help, the completion of Internship would have not been possible.

NAME: SHREYAS H S
USN :1DT21AI052

ABSTRACT

The design, development, and deployment of an AI-powered chatbot for the healthcare system are covered in detail in this study. It is now essential to integrate AI technologies into healthcare systems due to the growing need for affordable, accessible healthcare services. As a virtual assistant, the chatbot described in this paper facilitates user interactions, offers medical information, and makes first diagnosis depending on input from the user.

Utilizing machine learning methods, Natural Language Processing (NLP) methodologies, and a comprehensive medical knowledge base are all part of the development process. In addition to providing individualized healthcare advice, appointment scheduling, and prescription reminders, the chatbot is made to comprehend and react to user inquiries. Continuous learning is made possible by the incorporation of AI, which enables the chatbot to adjust over time to changing user preferences and medical knowledge.

Essential elements of the report encompass the training process utilized for the chatbot, the integration of ethical protocols to guarantee user confidentiality and data protection, and the assessment of the chatbot's efficacy via user feedback and interaction analysis. The paper also looks at how the chatbot can affect the healthcare system, with a focus on how it could improve accessibility, lighten the strain for medical staff, and increase patient participation.

The use of chatbots appears to be a useful way to expedite healthcare services as AI continues to change the healthcare industry. The potential of AI-powered chatbots to improve patient experiences and streamline healthcare delivery is highlighted in this paper, which advances knowledge on their application in the healthcare industry.

Table of Contents

CHAPTER -1

COMPANY PROFILE

A Brief History of Compsoft Technologies

Compsoft Technologies, founded on the principle of delivering high-quality technological solutions tailored to the unique business requirements of their clients, stands as a beacon of innovation in the software development industry. Recognizing the diverse needs of businesses, they have consistently strived to offer individualized technology solutions, emphasizing the importance of reaching the target market and automating existing processes.

The core features that clients seek in technological solutions – reaching the targeted market, automating processes into an e-client, and simplifying operations – form the cornerstone of Compsoft Technologies' design philosophy. Their expertise encompasses a wide range of technologies, including web design and development, MYSQL, PYTHON programming, HTML, CSS, ASP.NET, and LINQ.

Sarvamoola Software Services, a subsidiary specializing in ERP, connectivity, SEO services, conference management, online promotion, and custom software products, further extends their capabilities to meet the growing demands for automation.

Compsoft Technologies envisions leading the software development industry with a commitment to originality and innovation. Positioned as an unconventional software development company in Bangalore, India, they leverage well-researched skills to convert software development experience into tangible value for their clients. Collaboration with clients is a key aspect of their approach, as they believe that a deep understanding of client needs is the foundation for delivering optimal results.

The company's consulting services go beyond the conventional, occasionally leading to the complete redesign of solutions or the identification of entirely new application requirements during collaborative brainstorming sessions. This adaptive and client-centric approach reflects their dedication to staying at the forefront of technology trends and addressing evolving business challenges.

Compsoft Technologies firmly believes in the transformative power of technology to propel businesses to new heights. They assert that the right application of technology is a catalyst for growth, success, and customer satisfaction. In essence, their technological solutions are crafted not just to enhance efficiency, profitability, and dependability but ultimately to "delight customers," encapsulating their mission to create meaningful and impactful experiences through innovative technology applications.

CHAPTER 2

ABOUT THE COMPANY

Compsoft Technologies is a technology company that offers solutions for HTML, CSS, ASP.NET, LINQ, MYSQL, and Python programming, as well as all aspects of web design and development. Compsoft Technologies specializes in ERP, connectivity, SEO services, conference management, efficient online promotion, and custom software products, creating solutions that best meet the needs of clients while satisfying the ever-increasing automation requirements. The company where we can serve our clients to the best of our abilities and in accordance with industry standards since they have the correct mix of specialists as stakeholders. To develop technological advances in the fields of mobile technologies, web apps, and business and enterprise solutions, they have a youthful, passionate, creative, and energetic team of professionals. The motto of our organization is "Work together to offer our clients the best technological solutions, resulting in a positive present and future for our clients that will also have a cascading positive effect on their business structure." Offering a Full Suite of Technical Solutions is more than just our slogan; it is our goal for our clients, and we work very hard to realize it.

Products of Compsoft Technologies.

Android Apps

It's the procedure by which new apps for Android-powered devices are made. The Android software development kit (SDK) is typically used to develop applications in the Java (and/or Kotlin; or other such option) programming language. However, there are other development environments available; Kotlin, for example, supports the exact same Android APIs (and bytecode), while Go has restricted API access.

A full suite of development tools is included with the Android software development kit. These consist of tutorials, sample code, documentation, libraries, a QEMU-based handset emulator, and a debugger. Computers running Linux (any contemporary desktop Linux version), Mac OS X 10.5.8 or later, and Windows 7 or later are currently supported development platforms. Although the SDK is not available on Android itself as of March 2015, software development is still feasible through the use of specialized Android applications.

Web Application

It is a client-server computer programme, meaning that a web browser is used to run the client (which includes the client-side logic and user interface). Web mail, online retail sales, online auctions, wikis, instant messaging systems, and many more features are examples of common web applications. Web apps make use of web documents that are authored in common web browser-supporting formats like HTML and JavaScript.
Web applications are a specific type of client-server software in which the client programme is downloaded to the client computer over common protocols like HTTP when the user visits the relevant web page.

Every time a web page is viewed, upgrades to the client web software could take place. Throughout the session, the web browser serves as the universal client for all web applications, interpreting and displaying the pages. By streamlining the code and enabling one team to focus on the framework while another addresses a particular use case, web application frameworks can frequently minimize the amount of faults in a programme. Programme faults may give rise to security-related issues in applications that are frequently targeted by cyberattacks.

Best practises like using GET after POST can also be encouraged by frameworks. A web application is perceived as having a two-tier architecture by certain people. A "dumb" client that depends on a "smart" server, or a "smart" client that handles all the work and queries a "dumb" server, can be used in this scenario. The presentation tier would be managed by the client, the database (storage tier) by the server, and the business logic (application tier) by one or both of them. Although this makes the apps more scalable and divides the database and presentation, it still prevents true layer specialization, therefore most applications will outgrow this approach.
Giving software that was formerly given as local applications web access is a new tactic used by application software providers. Creating a completely new browser-based application or simply modifying an existing one to use a different presentation technology may be necessary, depending on the kind of application. With the use of these programmes, users can access software without needing to install it locally on their hard drive by paying a monthly or annual price. An organization that employs this tactic is referred to as an application service provider (ASP), and the software industry is currently paying close attention to ASPs.

Because private consumer information as well as company information may be compromised, security breaches on these kinds of applications are very concerning. Any online application must consider protecting these assets, and the development process needs to cover a few crucial operational areas. This covers the procedures for input, logging and auditing, asset handling, authorization, and authentication. Ensuring application security from the start can prove to be more efficient and less disruptive in the long run.

Web design

It is encompasses many different skills and disciplines in the production and maintenance of websites. The different areas of web design include web graphic design; interface design; authoring, including standardized code and proprietary software; user experience design; and Search engine optimization. The term web design is normally used to describe the design process relating to the front-end (client side) design of a website including writing mark up. Web design partially overlaps web engineering in the broader scope of web development. Web designers are expected to have an awareness of usability and if their role involves creating mark up then they are also expected to be up to date with web accessibility guidelines. Web design partially overlaps web engineering in the broader scope of web development.

Departments and services offered

Compsoft Technologies plays an essential role as an institute, the level of education, development of student's skills are based on their trainers. If you do not have a good mentor then you may lag in many things from others and that is why we at Compsoft Technologies gives you the facility of skilled employees so that you do not feel insecure about the academics. Personality development and academic status are some of those things which lie in the mentor's hands. If you are trained well then you can do well in your future and knowing its importance, Compsoft Technologies always tries to give you the best.

They have a great team of skilled mentors who are always ready to direct their trainees in the best possible way they can and to ensure the skills of mentors we held many skill development programs as well so that each and every mentor can develop their own skills with the demands of the companies so that they can prepare a complete packaged trainee.

Services provided by Compsoft Technologies.

- Core Java and Advanced Java
- Web services and development
- Dot Net Framework
- Python
- Selenium Testing
- Conference / Event Management Service
- Academic Project Guidance
- On The Job Training
- Software T

CHAPTER 3
INTRODUCTION

Introduction to ML

Machine Learning (ML) is a subfield of artificial intelligence (AI) that focuses on the development of algorithms and models that enable computers to learn and make predictions or decisions based on data. Unlike traditional programming, where explicit instructions are provided, ML systems use statistical techniques to automatically improve their performance over time.

Essential Aspects:

Learning from Data: ML algorithms enable systems to learn patterns and behaviors from data, allowing them to generalize and make predictions on new, unseen data.

Types of Machine Learning:

- **Supervised Learning**: Models are trained on labeled datasets, learning relationships between inputs and corresponding outputs.
- **Unsupervised Learning**: Algorithms identify patterns and structures within unlabeled data without predefined categories.
- **Reinforcement Learning**: Systems learn by interacting with an environment, receiving feedback in the form of rewards or penalties.

Algorithms in ML:

- **Linear Regression**: Predicts a continuous outcome based on input variables.
- **Decision Trees**: Construct tree-like structures to make decisions based on input features.
- **Neural Networks**: Deep learning models inspired by the human brain, suitable for complex tasks.
- **Support Vector Machines (SVM)**: Classify data points by finding optimal hyperplanes.
- **Feature Engineering:** The process of selecting, transforming, or creating relevant features to enhance model performance.
- **Model Evaluation:** ML models are assessed using metrics like accuracy, precision, recall, and F1 score to ensure effectiveness.

Challenges in ML:

- **Overfitting and Underfitting**: Striking a balance to avoid fitting the training data too closely or loosely.
- **Bias and Fairness**: Ensuring models are not biased and provide fair outcomes across diverse groups.

Applications of ML:

- **Healthcare**: Disease prediction, personalized treatment plans.
- **Finance**: Fraud detection, risk assessment.
- **Marketing**: Customer segmentation, recommendation systems.
- **Autonomous Vehicles**: Object recognition, decision-making.
- **Future Trends**: ML evolves with advancements in deep learning, reinforcement learning, and the integration of AI into various industries. Ethical considerations and model interpretability gain significance.

In conclusion, ML is a dynamic field shaping the future of technology, allowing systems to leverage data for intelligent decision-making and problem-solving.

Problem Statement

Creating a Health chatbot using rasa and natural language processing

One intriguing AI-powered response to medical questions is Compsoft Technologies' Chatbot for Healthcare System. The principal aim of the project is to perform a comprehensive analysis of all features that have been included, evaluate the Chatbot's overall correctness, and investigate potential areas for model enhancement.

During the testing process, a variety of scenarios, from standard medical questions to intricate user interactions, are carefully carried out. Through a comprehensive analysis of the system's reactions to various inputs, the testing endeavors to ascertain the Chatbot's resilience and effectiveness in furnishing precise and pertinent medical data. We hope to provide a thorough knowledge of the Chatbot's performance by measuring its correctness through this extensive review by comparing its responses to predetermined proper outcomes.

Feedback from users and stakeholders will also be critical in determining how satisfied users are in the actual world and pinpointing issues that the existing model may not be able to sufficiently solve. Through this feedback loop, we can add insightful information to the process of improvement.The assessment procedure will examine the advantages and disadvantages of the current model. Through the identification of error

patterns, restrictions when interpreting certain medical terminology, and difficulties answering intricate questions, our goal is to provide focused improvement plans. The training dataset could be enriched, the settings could be adjusted, or sophisticated natural language processing methods could be used.

Delivering a chatbot that thrives at user interactions in the healthcare industry is the initiative's ultimate goal. The testing process, findings, and suggested improvements will all be thoroughly described in the documentation that goes along with this project. We believe that by methodically addressing the areas that have been identified for improvement, we will be able to improve not only the accuracy and performance of the Chatbot but also its capacity to offer users in the healthcare ecosystem more dependable and relevant information.

CHAPTER 4

SYSTEM ANALYSIS

Existing System-

Manual Interaction:

In the existing healthcare system, patients typically interact with healthcare providers through manual means, such as phone calls or in-person visits

Limited Accessibility:

Access to healthcare information and assistance is often limited to the availability of healthcare professionals during working hours.

Overwhelmed Healthcare Systems:

Healthcare systems may be overwhelmed with routine queries, leading to delays in responses and suboptimal patient experience.

Lack of Proactive Care:

The existing system lacks proactive health monitoring and support for preventive care.

Proposed System-

Automated Chatbot Interaction:

Implementing a healthcare chatbot powered by AI and Rasa to automate and enhance patient interactions

24/7 Availability:

The proposed system ensures round-the-clock accessibility, allowing patients to seek information and assistance at any time.

Efficient Query Handling:

The chatbot efficiently handles routine queries, freeing up healthcare professionals to focus on more complex issues.

Proactive Health Monitoring:

Introducing proactive health monitoring features to track patient health indicators and provide timely interventions or recommendation

Personalized Healthcare Assistance:

Implementing a system that offers personalized assistance based on the user's medical history, preferences, and symptoms.

Objective of the System

lmprove Access to Healthcare:

The primary objective is to improve access to healthcare information and support by leveraging AI-driven chatbot technology.

Enhance Patient Engagement:

Increase patient engagement by providing a responsive and interactive platform for queries, guidance, and health monitoring.

Optimize Healthcare Resources:

Optimize the utilization of healthcare resources by automating routine tasks, allowing healthcare professionals to focus on critical patient care.

Provide Timely Health Interventions:

Enable timely interventions by monitoring patient health indicators and providing proactive recommendations or alerts.

CHAPTER 5

REQUIREMENT ANALYSIS

Hardware requirement specification

The Hardware Requirement Specification (HRS) outlines the necessary physical components and infrastructure to support the deployment and operation of the AI-powered health chatbot developed using Rasa.

1. Server Infrastructure:

Processor: Dual-core or higher processors for efficient handling of concurrent user requests.

RAM: Minimum 8 GB RAM to ensure smooth execution of Rasa and other auxiliary services.

Storage: Adequate storage capacity for storing the chatbot's models, datasets, and logs. SSD storage is recommended for faster data retrieval.

2. Networking:

Internet Connectivity: High-speed and reliable internet connectivity to facilitate real-time interactions and data exchange.

Firewall Configuration: Implementation of a firewall to enhance security and control incoming and outgoing network traffic.

3. Deployment Environment:

Cloud Services: Utilization of cloud platforms (e.g., AWS, Azure) for scalability, flexibility, and efficient resource management.

Load Balancer: If applicable, deploy a load balancer to evenly distribute incoming chatbot requests across multiple servers, ensuring optimal performance.

4. Backup and Redundancy:

Regular Backups: Scheduled backups of chatbot models, databases, and system configurations to prevent data loss.

Redundant Systems: Implementation of redundancy measures to ensure continuous chatbot availability in case of hardware failures.

Software requirement specification

The Software Requirement Specification (SRS) outlines the software components and tools required for the development, deployment, and maintenance of the AI-powered health chatbot.

1. Development Environment:

Rasa Framework: The core framework for developing natural language understanding (NLU) and dialogue management components of the chatbot.

Python: Programming language for implementing custom functionalities, integrations, and extensions within the Rasa framework.

Version Control: Git for version control to track changes and collaborate on the development of the chatbot.

2. Database Management:

Database System: A relational database management system (e.g., PostgreSQL, MySQL) to store user data, chat logs, and other relevant information.

Data Encryption: Implementation of encryption techniques to secure sensitive healthcare data stored in the database.

3. Server-Side Technologies:

Web Server: A web server (e.g., Nginx, Apache) to handle HTTP requests and facilitate communication between the chatbot and users.

Containerization: Docker for containerization, ensuring consistent deployment across different environments.

4. External APIs:

Integration APIs: If applicable, integration with external APIs for accessing additional medical knowledge, appointment scheduling systems, or other relevant services.

5. User Interface:

Frontend Framework: If a web-based interface is required, the selection of a frontend framework (e.g., React, Angular) for designing an intuitive and responsive user interface.

6. Testing and Quality Assurance:

Testing Frameworks: Implementation of testing frameworks (e.g., Pytest) for unit testing, integration testing, and ensuring the robustness of the chatbot.

Continuous Integration: CI/CD pipelines for automating the testing and deployment processes.

Authentication Mechanism: Implementation of secure user authentication mechanisms to protect sensitive healthcare information.

CHAPTER 6

DESIGN ANALYSIS

1. Overview of Design Philosophy:

In this section, we aim to provide a comprehensive understanding of the design philosophy that underlies the development of our healthcare chatbot. The design philosophy encapsulates the core goals and objectives guiding the creation of the chatbot, shaping its functionality and user interaction.

2. User Persona and Needs:

Defining the target user persona is crucial to tailoring the chatbot's capabilities to specific healthcare needs. We'll delve into the characteristics of our intended users and the healthcare challenges our chatbot is designed to address.

3. Conversation Flow:

An in-depth analysis of the conversation flow will uncover the intricacies of how user inputs are processed, how the chatbot generates responses, and the logic governing conversation branching points. This section aims to illuminate the decision-making process within the chatbot.

4. Intent Recognition and NLP Techniques:

The heart of the chatbot lies in its ability to understand user intent. We'll discuss the techniques employed for intent recognition, shedding light on the natural language processing (NLP) methods used to comprehend and interpret user queries.

5. Entity Recognition:

This section explores how our chatbot identifies and extracts entities from user inputs, emphasizing the significance of entity recognition in the healthcare context.

6. Context Management:

Seamless conversation requires effective context management. We'll explain how our chatbot handles and maintains context throughout a dialogue, including strategies for remembering user preferences and ensuring a coherent user experience.

7. Integration with Healthcare Systems:

If applicable, we'll outline the integration points with existing healthcare systems, emphasizing data privacy and security measures in place to safeguard sensitive information.

8. Multimodal Capabilities (if applicable):

For chatbots supporting multimedia inputs, this section discusses how images, voice, or other forms of data are processed and integrated into the conversational experience.

9. Personalization and Adaptability:

This part delves into how our chatbot personalizes responses based on user history or preferences and adapts to evolving user needs.

10. Feedback Mechanisms and Continuous Improvement:

User feedback is invaluable for enhancing the chatbot's performance. We'll explore the mechanisms in place for collecting user feedback and how this input is utilized for continuous improvement.

11. Scalability and Performance:

To ensure the chatbot can handle varying levels of demand, we'll discuss considerations for scalability, performance metrics, and optimization strategies implemented.

12. User Interface Design (if applicable):

If our chatbot has a user interface, this section covers its design principles, focusing on simplicity, accessibility, and user-friendly features.

CHAPTER 7

IMPLEMENTATION-

1. Technology Stack:

Our implementation is grounded in a robust technology stack that forms the backbone of the healthcare chatbot. Detail the programming languages, frameworks, and tools used in the development process.

2. Rasa Framework:

Rasa serves as the cornerstone of our chatbot development, offering a flexible and open-source platform for conversational AI. Discuss specific components of Rasa utilized and their roles in our implementation.

3. NLP Libraries:

Dive into the natural language processing (NLP) libraries integrated into the implementation. This could include libraries for tokenization, named entity recognition, sentiment analysis, and other language-related tasks.

4. Machine Learning Models:

Specify the machine learning models employed, such as those for intent recognition, entity recognition, and any other models contributing to the chatbot's intelligence.

5. Data Collection and Preprocessing:

Outline the procedures for data collection, emphasizing the relevance of high-quality training data. Discuss any preprocessing steps applied to clean and enhance the dataset.

6. Intent and Entity Annotation:

Elaborate on the annotation process for intents and entities in the training data. This step is crucial for teaching the chatbot to understand user input accurately.

7. Training Process:

Provide insights into the training process, explaining how the machine learning models are trained on the annotated data. Discuss the iterations and adjustments made to refine the model's performance.

8. Integration with External Systems:

If applicable, describe the integration points with external healthcare systems. Highlight any challenges faced and solutions implemented to ensure seamless communication.

9. User Interface Development (if applicable):

For chatbots with a user interface, detail the development process, including design principles, user experience considerations, and accessibility features.

10. Testing and Validation:

Discuss the rigorous testing procedures implemented to ensure the chatbot's reliability. Cover unit testing, integration testing, and user acceptance testing to validate the chatbot's performance.

11. Deployment Strategy:

Explain the deployment strategy adopted, whether it's cloud-based, on-premises, or a combination. Discuss any considerations for scalability and redundancy.

12. Monitoring and Analytics:

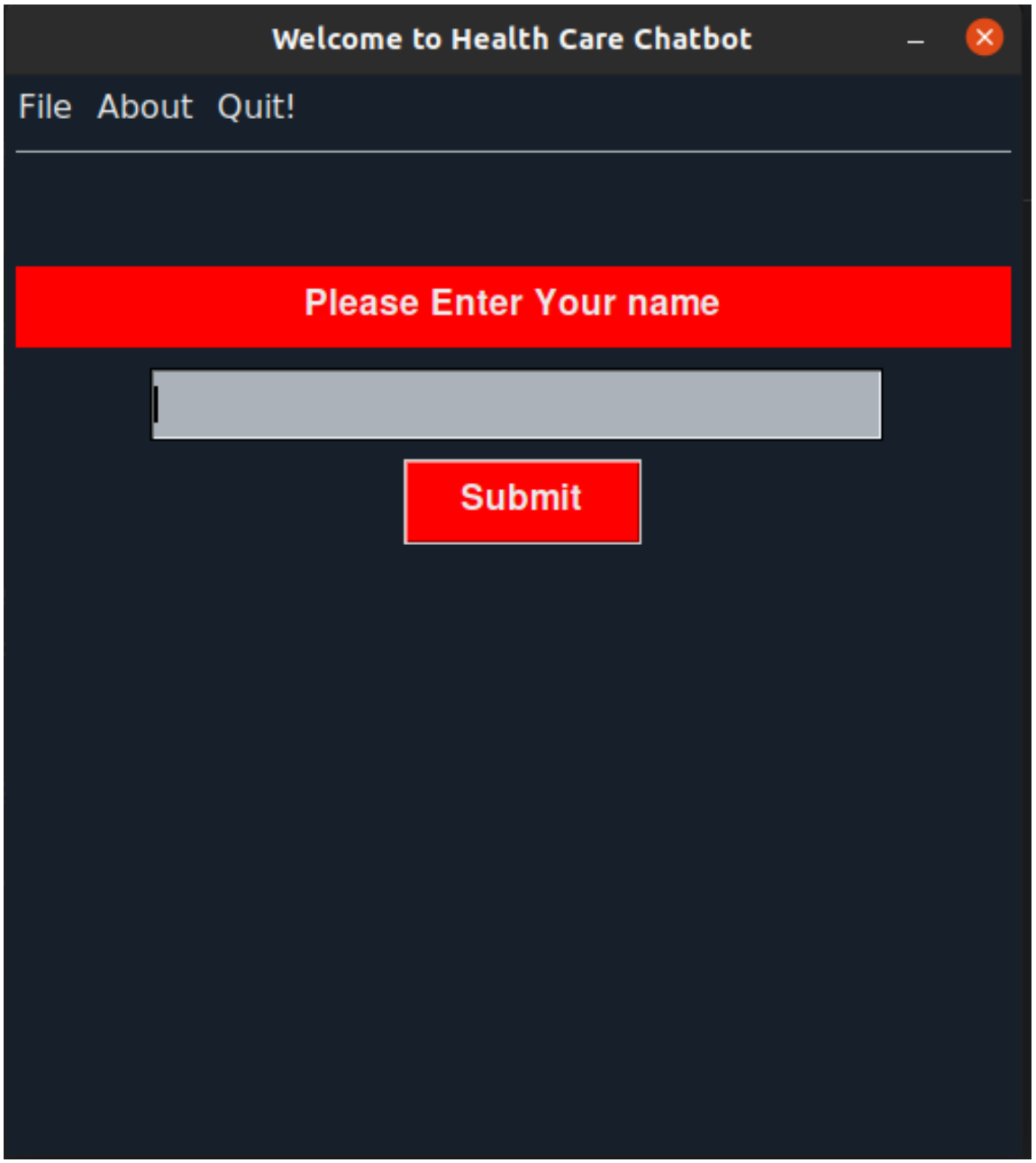
Highlight the implementation of monitoring tools and analytics to track the chatbot's performance in real-time. Discuss metrics such as response time, user satisfaction, and error rates.

13. Security Measures:

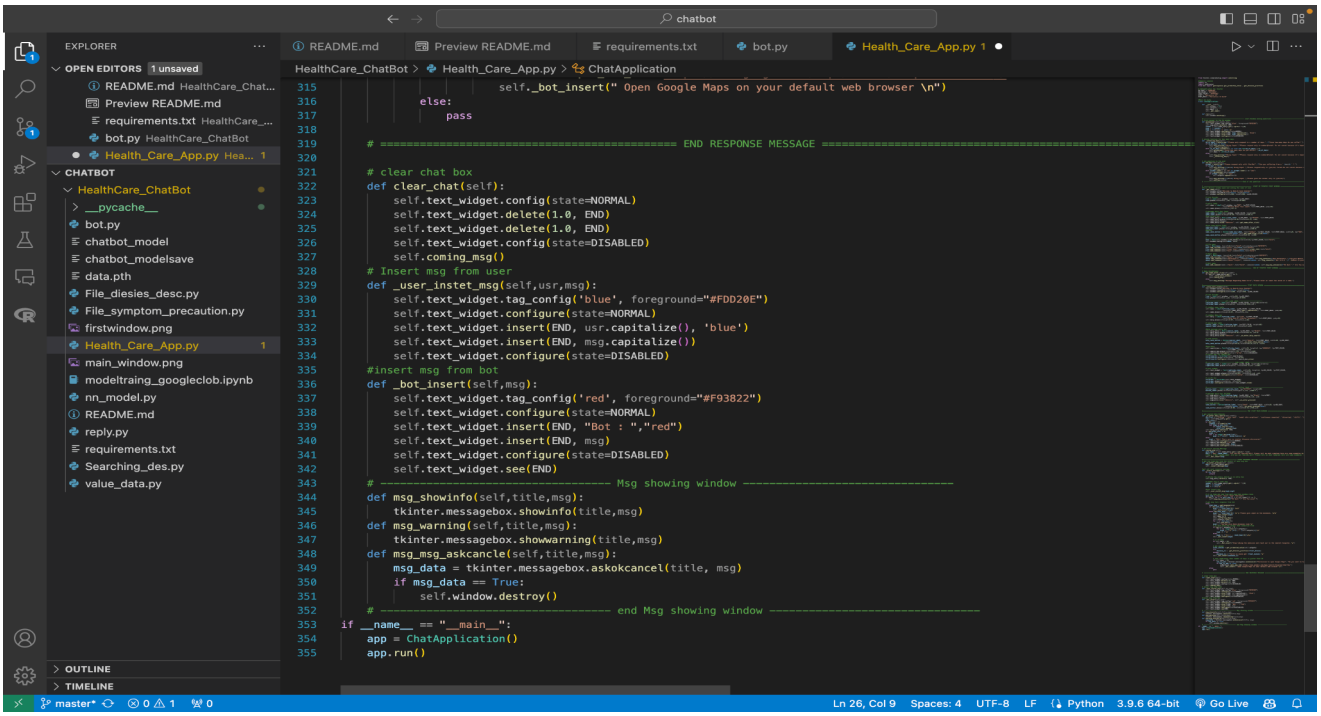
Address the security measures implemented to safeguard sensitive healthcare data. Discuss encryption, access controls, and any compliance standards adhered to.

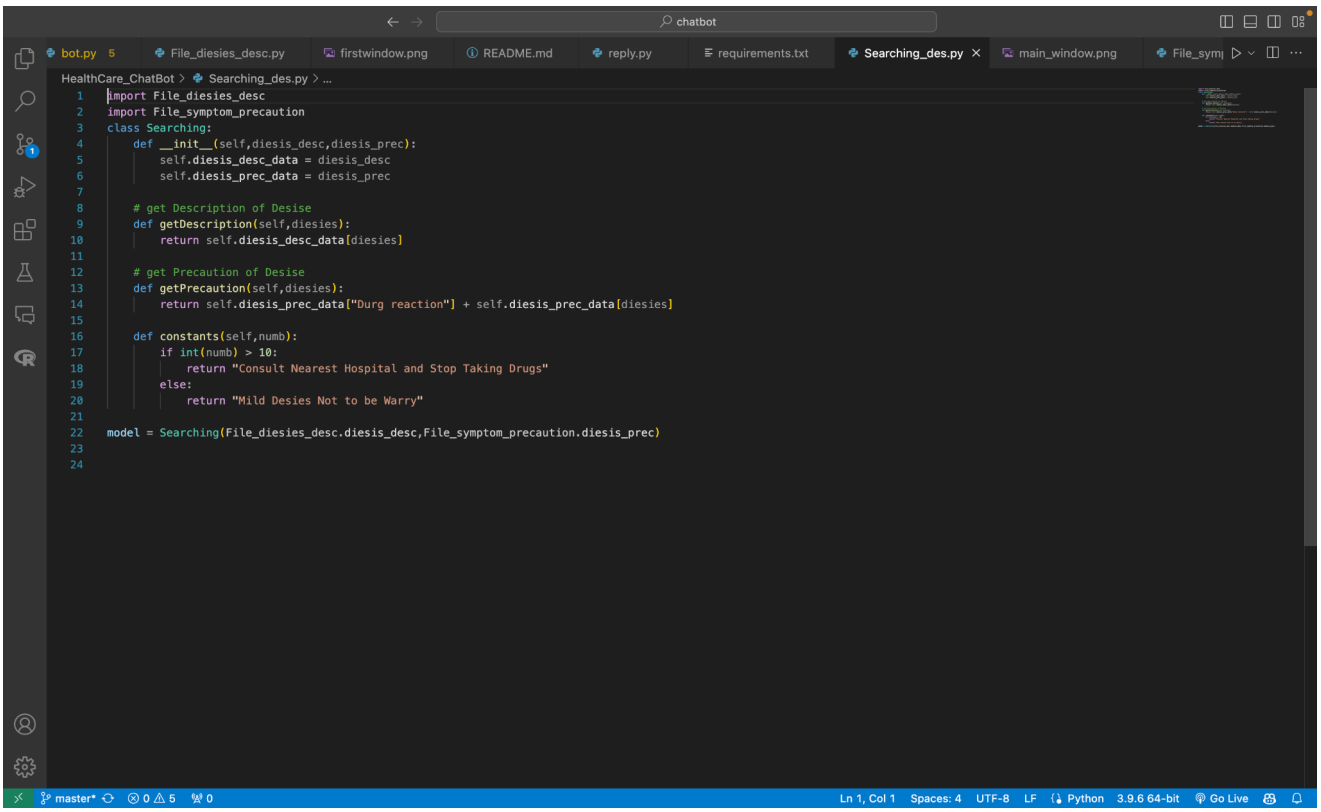
CHAPTER 8

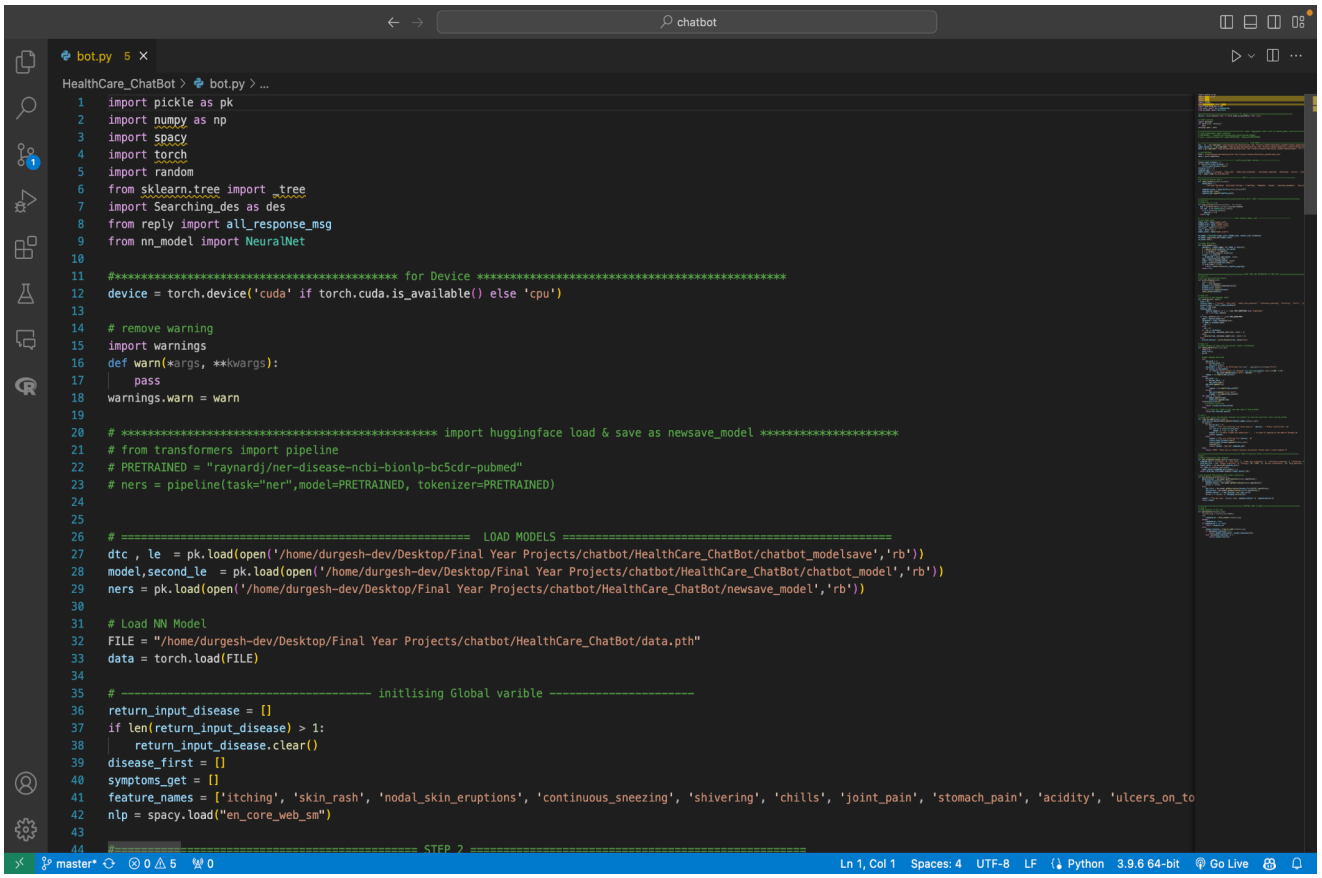
SNAPSHOTS

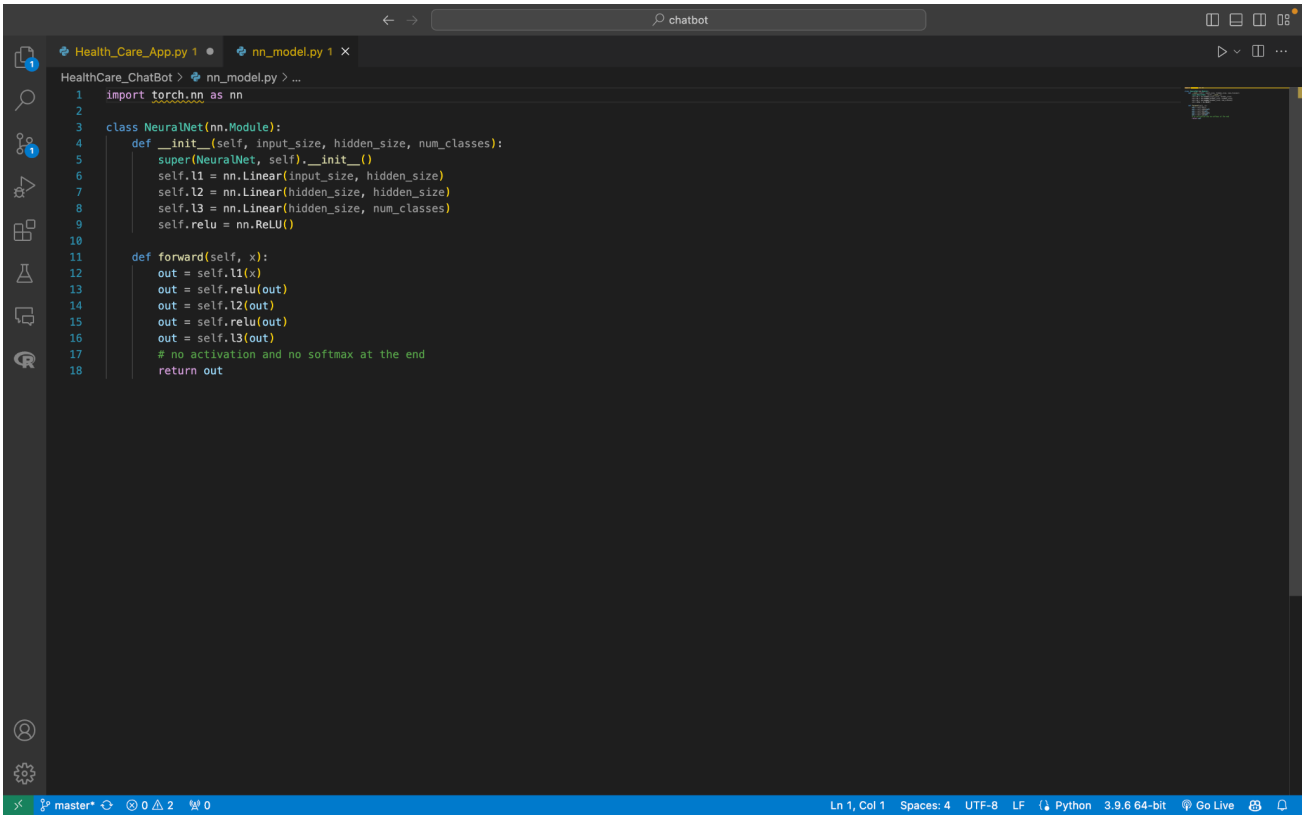


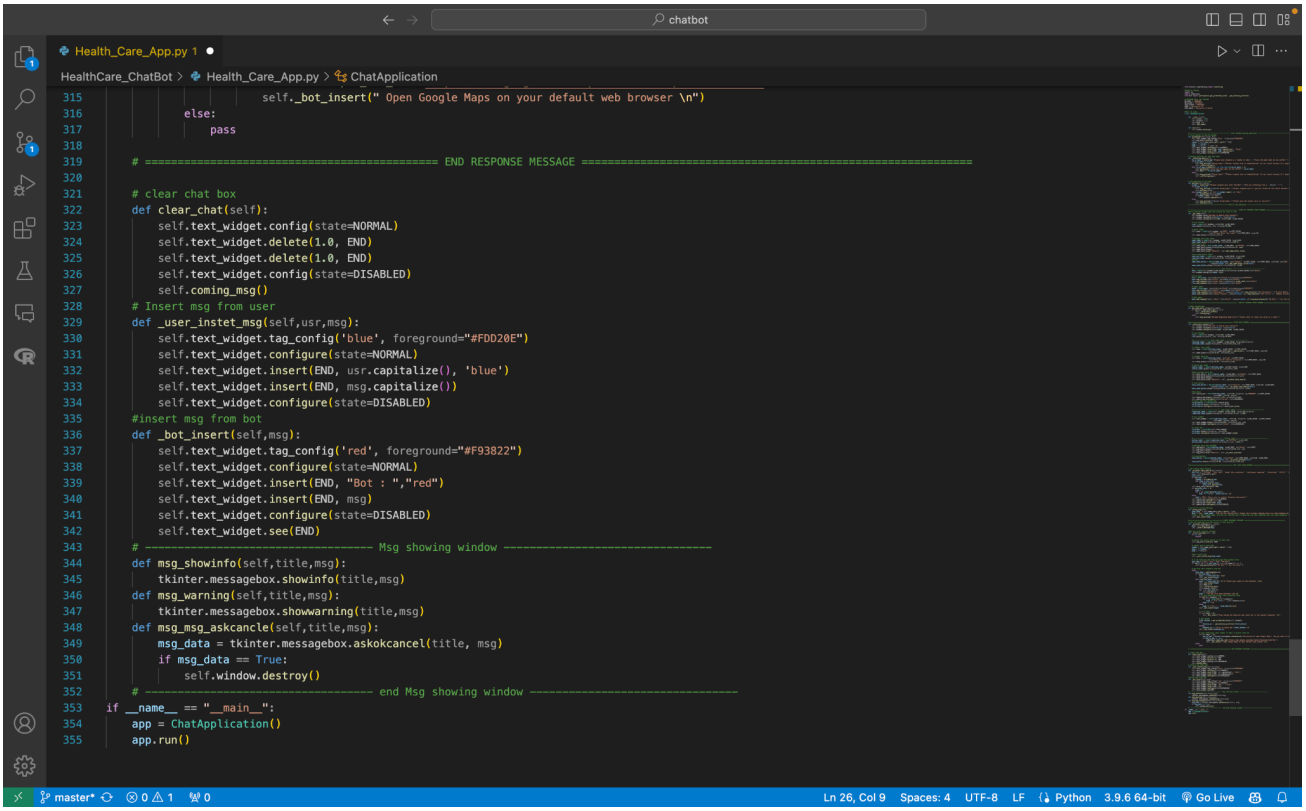

Welcome to Health Care Chatbot
File About Quit!
Help Search (Similar diseases)
Search
Bot : Hey! person , Are you not feeling well? Please tell me what symptoms here are some examples:
--> fever
--> cold
--> cough
--> headache
--> stomach_pain
--> abdominal_pain
--> dehydration
--> swelling
--> acidity
--> internal_itching
--> sneezing
--> vomiting
--> anxiety , etc
Note: Please use underscore (_) in place of spacing in the name of disease.
Send


chatbot
EXPLORER
README.md
Preview README.md
requirements.txt
bot.py
Health_Care_App.py 1
HealthCare_ChatBot > Health_Care_App.py > ChatApplication
OPEN EDITORS 1 unsaved
CHATBOT
HealthCare_ChatBot
__pycache__
bot.py
chatbot_model
chatbot_modelsave
data.pth
File_diesies_desc.py
File_symptom_precaution.py
firstwindow.png
Health_Care_App.py
main_window.png
modeltraing_googleclob.ipynb
nn_model.py
README.md
reply.py
requirements.txt
Searching_des.py
value_data.py
OUTLINE
TIMELINE
315 self._bot_insert(" Open Google Maps on your default web browser \n")
316 else:
317 pass
318
319 # ========== END RESPONSE MESSAGE ==========
320
321 # clear chat box
322 def clear_chat(self):
323 self.text_widget.config(state=NORMAL)
324 self.text_widget.delete(1.0, END)
325 self.text_widget.delete(1.0, END)
326 self.text_widget.config(state=DISABLED)
327 self.coming_msg()
328 # Insert msg from user
329 def _user_instet_msg(self,usr,msg):
330 self.text_widget.tag_config('blue', foreground="#FDD20E")
331 self.text_widget.configure(state=NORMAL)
332 self.text_widget.insert(END, usr.capitalize(), 'blue')
333 self.text_widget.insert(END, msg.capitalize())
334 self.text_widget.configure(state=DISABLED)
335 #insert msg from bot
336 def _bot_insert(self,msg):
337 self.text_widget.tag_config('red', foreground="#F93822")
338 self.text_widget.configure(state=NORMAL)
339 self.text_widget.insert(END, "Bot : ","red")
340 self.text_widget.insert(END, msg)
341 self.text_widget.configure(state=DISABLED)
342 self.text_widget.see(END)
343 # ---------- Msg showing window ----------
344 def msg_showinfo(self,title,msg):
345 tkinter.messagebox.showinfo(title,msg)
346 def msg_warning(self,title,msg):
347 tkinter.messagebox.showwarning(title,msg)
348 def msg_msg_askcancle(self,title,msg):
349 msg_data = tkinter.messagebox.askokcancel(title, msg)
350 if msg_data == True:
351 self.window.destroy()
352 # ---------- end Msg showing window ----------
353 if __name__ == "__main__":
354 app = ChatApplication()
355 app.run()
master* 0 1 0
Ln 26, Col 9 Spaces: 4 UTF-8 LF Python 3.9.6 64-bit Go Live


chatbot
bot.py 5
File_diesies_desc.py
firstwindow.png
README.md
reply.py
requirements.txt
Searching_des.py
main_window.png
HealthCare_ChatBot > Searching_des.py > ...
1 import File_diesies_desc
2 import File_symptom_precaution
3 class Searching:
4 def __init__(self,diesis_desc,diesis_prec):
5 self.diesis_desc_data = diesis_desc
6 self.diesis_prec_data = diesis_prec
7
8 # get Description of Desise
9 def getDescription(self,diesies):
10 return self.diesis_desc_data[diesies]
11
12 # get Precaution of Desise
13 def getPrecaution(self,diesies):
14 return self.diesis_prec_data["Durg reaction"] + self.diesis_prec_data[diesies]
15
16 def constants(self,numb):
17 if int(numb) > 10:
18 return "Consult Nearest Hospital and Stop Taking Drugs"
19 else:
20 return "Mild Desies Not to be Warry"
21
22 model = Searching(File_diesies_desc.diesis_desc,File_symptom_precaution.diesis_prec)
23
24
master* 0 5 0
Ln 1, Col 1 Spaces: 4 UTF-8 LF Python 3.9.6 64-bit Go Live


chatbot
bot.py 5
HealthCare_ChatBot > bot.py > ...
import pickle as pk
import numpy as np
import spacy
import torch
import random
from sklearn.tree import _tree
import Searching_des as des
from reply import all_response_msg
from nn_model import NeuralNet
#************************************** for Device **
device = torch.device('cuda' if torch.cuda.is_available() else 'cpu')
# remove warning
import warnings
def warn(*args, **kwargs):
pass
warnings.warn = warn
# ** import huggingface load & save as newsave_model ********************
# from transformers import pipeline
# PRETRAINED = "raynardj/ner-disease-ncbi-bionlp-bc5cdr-pubmed"
# ners = pipeline(task="ner",model=PRETRAINED, tokenizer=PRETRAINED)
# == LOAD MODELS ==
dtc , le = pk.load(open('/home/durgesh-dev/Desktop/Final Year Projects/chatbot/HealthCare_ChatBot/chatbot_modelsave','rb'))
model,second_le = pk.load(open('/home/durgesh-dev/Desktop/Final Year Projects/chatbot/HealthCare_ChatBot/chatbot_model','rb'))
ners = pk.load(open('/home/durgesh-dev/Desktop/Final Year Projects/chatbot/HealthCare_ChatBot/newsave_model','rb'))
# Load NN Model
FILE = "/home/durgesh-dev/Desktop/Final Year Projects/chatbot/HealthCare_ChatBot/data.pth"
data = torch.load(FILE)
# ------------------------------------ initlising Global varible ---------------------
return_input_disease = []
if len(return_input_disease) > 1:
return_input_disease.clear()
disease_first = []
symptoms_get = []
feature_names = ['itching', 'skin_rash', 'nodal_skin_eruptions', 'continuous_sneezing', 'shivering', 'chills', 'joint_pain', 'stomach_pain', 'acidity', 'ulcers_on_to
nlp = spacy.load("en_core_web_sm")
#== STEP 2 ==
master* 0 5 0
Ln 1, Col 1 Spaces: 4 UTF-8 LF Python 3.9.6 64-bit Go Live


chatbot
Health_Care_App.py 1
nn_model.py 1
HealthCare_ChatBot > nn_model.py > ...
import torch.nn as nn
class NeuralNet(nn.Module):
def __init__(self, input_size, hidden_size, num_classes):
super(NeuralNet, self).__init__()
self.l1 = nn.Linear(input_size, hidden_size)
self.l2 = nn.Linear(hidden_size, hidden_size)
self.l3 = nn.Linear(hidden_size, num_classes)
self.relu = nn.ReLU()
def forward(self, x):
out = self.l1(x)
out = self.relu(out)
out = self.l2(out)
out = self.relu(out)
out = self.l3(out)
# no activation and no softmax at the end
return out
master* 0 2 0
Ln 1, Col 1 Spaces: 4 UTF-8 LF Python 3.9.6 64-bit Go Live


chatbot
Health_Care_App.py 1
HealthCare_ChatBot > Health_Care_App.py > ChatApplication
self._bot_insert(" Open Google Maps on your default web browser \n")
else:
pass
# ========== END RESPONSE MESSAGE ==========
# clear chat box
def clear_chat(self):
self.text_widget.config(state=NORMAL)
self.text_widget.delete(1.0, END)
self.text_widget.delete(1.0, END)
self.text_widget.config(state=DISABLED)
self.coming_msg()
# Insert msg from user
def _user_instet_msg(self,usr,msg):
self.text_widget.tag_config('blue', foreground="#FDD20E")
self.text_widget.configure(state=NORMAL)
self.text_widget.insert(END, usr.capitalize(), 'blue')
self.text_widget.insert(END, msg.capitalize())
self.text_widget.configure(state=DISABLED)
#insert msg from bot
def _bot_insert(self,msg):
self.text_widget.tag_config('red', foreground="#F93822")
self.text_widget.configure(state=NORMAL)
self.text_widget.insert(END, "Bot : ","red")
self.text_widget.insert(END, msg)
self.text_widget.configure(state=DISABLED)
self.text_widget.see(END)
# ---------- Msg showing window ----------
def msg_showinfo(self,title,msg):
tkinter.messagebox.showinfo(title,msg)
def msg_warning(self,title,msg):
tkinter.messagebox.showwarning(title,msg)
def msg_msg_askcancle(self,title,msg):
msg_data = tkinter.messagebox.askokcancel(title, msg)
if msg_data == True:
self.window.destroy()
# ---------- end Msg showing window ----------
if __name__ == "__main__":
app = ChatApplication()
app.run()
master*
Ln 26, Col 9
Spaces: 4
UTF-8
LF
Python
3.9.6 64-bit
Go Live

CONCLUSION

In conclusion, the development of the Healthcare Chatbot System using AI by Compsoft Technologies reflects a strategic and forward-thinking approach. The project was meticulously designed to accommodate future modifications seamlessly, ensuring its adaptability and relevance in the dynamic landscape of healthcare technology. The project's conclusions highlight several key benefits:

Firstly, the automation of the entire system emerges as a pivotal factor in enhancing operational efficiency. By streamlining processes and minimizing manual interventions, the Chatbot system contributes to a more streamlined and responsive healthcare environment.

The implementation of a friendly graphical user interface stands out as a notable improvement over existing systems. This user-centric design not only fosters ease of use but also contributes to a more positive and engaging user experience.

The system's capability to provide appropriate access to authorized users based on their permissions addresses the critical aspect of data security and confidentiality in healthcare operations.

Efficient communication is a cornerstone of effective healthcare, and the Chatbot system effectively overcomes delays in communication, ensuring timely access to information and services.

The system's capacity to facilitate easy updating of information aligns with the dynamic nature of healthcare data. This feature ensures that the system stays current and relevant, reflecting the latest medical knowledge and advancements.

Furthermore, the striking features of system security, data security, and reliability underscore the commitment to maintaining the integrity and privacy of healthcare information.

Lastly, the project's foresight in allowing for future modifications ensures its longevity and relevance. This adaptability positions the Healthcare Chatbot System as a scalable solution that can evolve with the changing needs and advancements in the healthcare domain. In essence, the conclusions drawn from the project underscore its effectiveness in addressing critical healthcare challenges and laying a foundation for ongoing innovation and improvement.

The automation of the entire system not only enhances operational efficiency but also reduces the likelihood of errors associated with manual processes. By leveraging AI,

repetitive tasks are automated, freeing up valuable human resources to focus on more complex and critical aspects of healthcare delivery.

The implementation of a user-friendly graphical interface not only improves ease of use but also contributes to patient engagement. A positive user experience can foster better patient-provider interactions, leading to increased trust and satisfaction. This user-centric approach aligns with the broader trend in healthcare towards patient-centered care.

The system's capability to provide appropriate access to authorized users plays a crucial role in safeguarding sensitive healthcare information. This granular control ensures that only individuals with the necessary permissions can access specific data, addressing compliance requirements and bolstering patient privacy.

Efficient communication, as facilitated by the Chatbot system, is paramount in a healthcare setting where timely access to information can significantly impact patient outcomes. The system's ability to overcome delays in communication is particularly beneficial in emergency situations and time-sensitive healthcare decisions.

The ease of updating information is vital in healthcare, where knowledge is continually evolving. The system's capacity to facilitate easy updates ensures that healthcare professionals have access to the latest medical guidelines, research findings, and treatment protocols, ultimately enhancing the quality of patient care.

The emphasis on system security, data security, and reliability speaks to the commitment to maintaining the highest standards of confidentiality and integrity in healthcare operations. This is particularly critical in an era where data breaches and cybersecurity threats are prevalent concerns in the healthcare industry.

Lastly, the foresight in allowing for future modifications positions the Healthcare Chatbot System as a solution capable of evolving with the rapidly changing landscape of healthcare technology. This adaptability is vital in addressing emerging healthcare challenges, integrating new technologies, and staying ahead of industry trends.

In essence, the conclusions not only underscore the immediate benefits of the Healthcare Chatbot System but also highlight its broader implications for shaping the future of healthcare delivery, emphasizing efficiency, patient-centered care, and ongoing innovation.

REFERENCE

Healthcare-chatbot-master.zip - Google Drive

https://github.com/robinsingh051/Healthcare-chatbot/blob/master/README.md

https://medium.com/@gangulyritwik2003/healthcare-chatbot-using-rasa-59e9ae96d4ef

https://github.com/topics/rasa-chatbot?l=html

https://zenodo.org/records/6395568