**ML-BASED MEDICINE RECOMMENDATION SYSTEM**

****

**Session Spring 2021 - Fall 2025**

Program

**Bachelor of Studies in Artificial Intelligence**

**Submitted by:**

Muhammad Shahab 303-211022

Irfan Haider 303-211017

Sharif Ullah 303-211080

**Supervised by:**

Dr. Faisal Bahadur

Lecturer, Department of CS & IT

**HAZARA UNIVERSITY, MANSEHRA**

**Approval Certificate**

This is certifying that we have read the project report title **“ML-Based Medicine Recommendation System.”** submitted by the following students of BSAI, Fall 2025 Hazara University Mansehra.

|  |  |
| --- | --- |
| **Name** | **Roll NO.** |
| Muhammad Shahab | 303-211022 |
| Irfan Haider | 303-211017 |
| Sharif Ullah | 303-211080 |

It is our judgment that is project report is of sufficient standard to warrant its acceptance by the Department of CS & IT, Hazara University Mansehra.

**COMMITTEE**

1. **Supervisor -----------------------**
2. **External Examiner -----------------------**
3. **Chairman -----------------------**

**DECLARATION**

It is hereby declared that this project, **“**ML-Based Medicine Recommendation System,” neither as a whole nor as a part, has been copied from any source. It is further declared that we developed this software and this report entirely based on our personal efforts, made under the sincere guidance of our project supervisor.

If any part of this software or report is found to be copied or derived from another report, we are fully responsible for the consequences. No portion of the work presented in this report has been submitted in support of any application for any other degree or qualification at any other university or institute of learning.

We further declare that this software and all associated documents, reports, and records are submitted as a partial requirement for the degree of **BS Artificial Intelligence**. We therefore understand and transfer copyrights for these materials to **Hazara University, Mansehra**.

**ACKNOWLEDGEMENTS**

We extend our heartfelt gratitude to all those who contributed to the successful completion of this final year project on "**ML-Based Medicine Recommendation System**."

First and foremost, we are deeply indebted to our supervisor, Dr. Faisal Bahadur, whose invaluable guidance, constructive feedback, and unwavering support made this project possible throughout this research endeavor. His expertise and mentorship were instrumental in shaping our understanding and approach to this complex subject matter.

We would like to acknowledge Hazara University, Mansehra, and express our sincere appreciation to the faculty members of the Department of Computer Science & Information Technology for fostering an excellent academic environment. The comprehensive curriculum, quality instruction, and access to essential learning resources provided us with the fundamental knowledge and technical skills required to undertake this project.

We are also grateful to our family members and friends whose constant encouragement, patience, and moral support sustained us throughout the challenging phases of this project development.

Finally, we acknowledge all the researchers and authors whose published works and contributions to the field of machine learning and healthcare informatics provided the theoretical foundation for our research.

**DEDICATION**

With deepest gratitude and love, we dedicate this work to the people who mean the world to us.

To our **mother and father**, whose sacrifices, prayers, and endless support have shaped our journey. Your belief in us was the light that guided us.

To our **grandfather and grandmother**, a symbol of strength and wisdom, your presence remains a blessing in our lives.

To our **brother**, who has always stood behind us with encouragement and quiet pride.

And to our **dearest friend**, for the laughter, late-night motivation, and for never letting us give up.

This achievement is as much yours as it is ours.

**PROJECT IN BRIEF**

**Project Title**: ML-Based Medicine Recommendation System

**Developed By**: Muhammad Shahab, Irfan Haider, Sharif Ullah

**Supervised By:** Dr. Faisal Bahadur

**Tools Used:** Jupyter Notebook, VSCode

**System Used:** Dell Core i5 8th Gen, 8GB RAM

**Operating System:** Microsoft Windows 10

**Starting Date:** March 2025

**Completion Date:** May 2025

**TABLE OF CONTENTS**

[**CHAPTER 1 xi**](#_Toc201259506)

[**INTRODUCTION xi**](#_Toc201259507)

[1.1 Background of Medicine Recommendation Systems 2](#_Toc201259508)

[1.2 Importance in Healthcare 2](#_Toc201259509)

[1.3 Role of AI and Machine Learning 3](#_Toc201259510)

[1.4 Objectives of the Project 5](#_Toc201259511)

[1.5 Scope (Extended) 6](#_Toc201259512)

[1.6 Feasibility 6](#_Toc201259513)

[1.7 Report Structure 6](#_Toc201259514)

[1.8 Summary 6](#_Toc201259515)

[CHAPTER 2 7](#_Toc201259516)

[LITERATURE REVIEW 7](#_Toc201259517)

[2.2 Existing Medicine Recommendation Systems 8](#_Toc201259518)

[2.3 Use of AI/ML in Medicine Recommendations 8](#_Toc201259519)

[2.4 Gaps in Existing Research 9](#_Toc201259520)

[2.5 Relevance to the Proposed System 9](#_Toc201259521)

[2.6 Summery 10](#_Toc201259522)

[CHAPTER 3 11](#_Toc201259523)

[SYSTEM DESIGN AND METHODOLOGY 11](#_Toc201259524)

[3.1 Chat Flow Diagram 12](#_Toc201259525)

[3.2 System Architecture 13](#_Toc201259526)

[3.3 Methodology 13](#_Toc201259527)

[3.3.1 Data Collection and Preprocessing 13](#_Toc201259528)

[3.3.2 Model Development 14](#_Toc201259529)

[3.3.3 Medicine Recommendation Mapping 14](#_Toc201259530)

[3.3.4 Bi-Directional Translation 14](#_Toc201259531)

[3.4 Tools and Technologies 15](#_Toc201259532)

[3.5 System Workflow 15](#_Toc201259533)

[3.6 Summery 15](#_Toc201259534)

[CHAPTER 4 17](#_Toc201259535)

[MPLEMENTATION 17](#_Toc201259536)

[4.1 Data Preparation 18](#_Toc201259537)

[4.2 Model Training and Serialization 19](#_Toc201259538)

[4.3 Flask Integration (Front-End and Back-End) 20](#_Toc201259539)

[4.4 Front-End (templates/) 21](#_Toc201259540)

[4.5 Static Assets 22](#_Toc201259541)

[4.5 Back-End (main.py) 22](#_Toc201259542)

[4.6 Features Delivered by the System 23](#_Toc201259543)

[4.7 Summery 24](#_Toc201259544)

[CHAPTER 5 25](#_Toc201259545)

[TESTING AND EVALUATION 25](#_Toc201259546)

[5.1 Testing and Evaluation 26](#_Toc201259547)

[5.2 Test Case 1: Valid Symptom Input 26](#_Toc201259548)

[5.4 Model Evaluation Results 33](#_Toc201259549)

[5.5 Final Selected Model: Random Forest Classifier 34](#_Toc201259550)

[5.6 Summery 35](#_Toc201259551)

[CHAPTER 6 36](#_Toc201259552)

[CONCLUSION AND FUTURE WORK 36](#_Toc201259553)

[6.1 Conclusion 37](#_Toc201259554)

[6.2 Future Work 37](#_Toc201259555)

[6.3 Final Remarks 38](#_Toc201259556)

[6.4 Summery 38](#_Toc201259557)

[References 39](#_Toc201259558)

LIST OF FIGURES

[Figure 4.1 Load Dataset 18](#_Toc201259629)

[Figure 4.2 Selecting the best model 19](#_Toc201259630)

[Figure 4.3 Saving the model 20](#_Toc201259631)

[Figure 4.4 Data Processing and Extraction Workflow 20](#_Toc201259632)

[Figure 4.5 Symptom Input and Processing Logic 21](#_Toc201259633)

[Figure 4.6 Flask Route Definitions Overview 22](#_Toc201259634)

[Figure 4.7 Static Files Directory Structure 22](#_Toc201259635)

[Figure 4.8 Model Prediction Function Implementation 23](#_Toc201259636)

[Figure 4.9 Health Care Center Prediction Interface 24](#_Toc201259637)

[Figure 5.1 Symptom Input and Result Output 27](#_Toc201259638)

[Figure 5.2 Symptom Input and Allergy Results 28](#_Toc201259639)

[Figure 5.3 Empty Symptom Input Warning 29](#_Toc201259640)

[Figure 5.4 Symptom Input and Malaria Results 29](#_Toc201259641)

[Figure 5.5 Symptom Input and Bronchial Asthma Results 30](#_Toc201259642)

[Figure 5.6 Model Accuracy and Confusion Matrices 32](#_Toc201259643)

[Figure 5.7 Model Training and Evaluation Code 34](#_Toc201259644)

[Figure 5.8 Best Model Selection Process 35](#_Toc201259645)

# CHAPTER 1

# INTRODUCTION

## 1.1 Background of Medicine Recommendation Systems

In the modern era of digital healthcare, the integration of artificial intelligence (AI) into medical systems has introduced innovative solutions for enhancing patient care. One such advancement is the development of medicine recommendation systems, which assist in identifying the most appropriate medications based on patient-specific data. Traditionally, prescribing medicine has been the sole responsibility of healthcare professionals who rely on their experience, diagnostic skills, and available clinical knowledge. However, the growing complexity of patient cases, especially those involving multiple comorbidities, drug interactions, and vast pharmaceutical options, has created the need for intelligent systems that can provide quick, data-driven support.

Medicine recommendation systems are AI-driven platforms designed to analyze a patient’s symptoms, medical history, and other health-related parameters to suggest relevant medications. These systems typically leverage machine learning algorithms to interpret and map input symptoms to corresponding diseases and recommend suitable drug treatments.

In our project, we aim to develop a symptom-based medicine recommendation system where users can input symptoms in natural language. The system processes this input, uses trained machine learning models to predict the most likely medical condition, and then recommends medications associated with the prediction. This approach reduces the dependency on manual lookups, supports faster decision-making, and increases accessibility, especially in remote or resource-constrained environments.

As the healthcare industry shifts toward data-driven solutions, medicine recommendation systems represent a vital step toward more efficient and intelligent clinical decision support.

## 1.2 Importance in Healthcare

The healthcare industry is under constant pressure to deliver accurate diagnoses, reduce treatment errors, and improve patient outcomes—all while operating within time and resource constraints. In this context, medicine recommendation systems play a pivotal role by introducing automation, accuracy, and efficiency into the clinical decision-making process.

One of the most critical aspects of healthcare is ensuring that patients receive the right medication at the right time. Mistakes in prescribing drugs—whether due to human error, incomplete patient history, or oversight of drug interactions—can lead to severe complications, increased hospital stays, and even fatal outcomes. A medicine recommendation system, powered by artificial intelligence and machine learning, helps minimize these risks by analyzing symptoms, medical history, and drug databases to generate reliable treatment suggestions.

In areas with limited access to medical specialists, such systems can assist frontline healthcare workers and even patients themselves in identifying suitable over-the-counter or prescription medicines, thus expanding the reach of quality healthcare. For busy clinicians, these systems act as decision support tools, saving time by quickly narrowing down treatment options based on large datasets.

Furthermore, medicine recommendation systems support the movement toward personalized medicine. By incorporating patient-specific data—such as allergies, prior conditions, and symptom patterns—these systems can deliver more tailored and safer recommendations than generic treatment guidelines.

In the context of our project, where the model predicts illnesses based on input symptoms and suggests corresponding medications, the importance lies in speed, accessibility, and support for non-experts. Whether used by patients for preliminary advice or by professionals for second opinions, such systems represent a powerful tool in enhancing both preventive and curative healthcare delivery.

## 1.3 Role of AI and Machine Learning

Artificial Intelligence (AI) and Machine Learning (ML) are transforming the landscape of healthcare by enabling systems that can learn from data, adapt to new information, and make intelligent decisions without explicit programming. In the domain of medicine recommendation systems, AI/ML plays a central role in analyzing complex, high-dimensional medical data to generate accurate, data-driven suggestions for diagnosis and treatment.

The core strength of machine learning lies in its ability to identify hidden patterns and relationships in data. In a medicine recommendation system, this means learning the associations between symptoms, diseases, and effective medications from historical medical records and pharmaceutical databases. For example, a supervised ML model can be trained on labeled datasets containing symptoms and their corresponding diagnoses, allowing it to predict potential diseases when new symptoms are input by a user.

In our project, machine learning is used to build a predictive model that receives symptoms as input and outputs the most likely condition. Once the condition is identified, the system retrieves a list of recommended medicines that are known to treat that specific illness. This pipeline automates what would otherwise require manual expertise, providing quick and reliable assistance.

AI techniques, such as Natural Language Processing (NLP), also contribute by enabling the system to understand user input written in natural language and translate it into structured data for further processing. Additionally, by incorporating feedback loops, AI can improve over time, making the system smarter with each interaction.

By leveraging AI/ML, our system offers several advantages:

* Accuracy in matching symptoms to the correct disease and medication.
* Scalability, allowing the system to handle large volumes of diverse patient data.
* Adaptability, enabling updates as new drugs and medical knowledge become available.
* Accessibility, especially for under-resourced settings lacking medical professionals.

Overall, AI and ML serve as the intelligent core of the medicine recommendation system, enabling it to perform complex tasks efficiently and reliably—thus contributing to safer, faster, and more informed healthcare delivery.

## 1.4 Objectives of the Project

The primary objective of this project is to develop an intelligent, AI-powered Medicine Recommendation System that can predict possible diseases based on user-provided symptoms and suggest appropriate medications accordingly. This system is designed to assist both healthcare professionals and patients by delivering fast, accurate, and accessible recommendations.

**The key objectives of the project are:**

1. To develop a machine learning-based model that can analyze symptoms input by the user and predict the most likely medical condition using trained algorithms.
2. To build a medicine recommendation engine that provides appropriate drug suggestions based on the predicted condition, using a curated database of medicines, diseases, and treatment protocols.
3. To implement a user-friendly, text-based interface that allows users to enter symptoms in natural language, enhancing accessibility for both professionals and laypersons.
4. To enable bi-directional translation functionality, allowing conversion between standard medical terms/symbols and plain text to support clinical usability and improve understanding.
5. To evaluate the system’s accuracy and performance through simulation using real-world patient data or synthetic datasets, focusing on factors such as precision, recall, and user satisfaction.
6. To identify and address limitations in existing medicine recommendation systems, such as lack of personalization and limited adaptability, by integrating AI techniques that improve over time with more data.
7. To support better clinical decision-making and reduce the time required for diagnosis and prescription, particularly in remote or resource-constrained environments.

## 1.5 Scope (Extended)

This project is highly relevant for Pakistan, where access to immediate and accurate medical advice is limited in rural and underdeveloped districts. An AI-powered medicine recommendation system can bridge the healthcare accessibility gap, particularly in regions like Khyber Pakhtunkhwa and Punjab.

## 1.6 Feasibility

* **Technical Feasibility**: All required tools, such as Python, Flask, and Scikit-learn, are open-source and well-supported.

## 1.7 Report Structure

Chapter 2 presents the requirements specification. Chapter 3 describes the system design. Chapter 4 covers implementation and testing. Chapter 5 presents results. Chapter 6 concludes with future directions.

## 1.8 Summary

This chapter introduces the concept of medicine recommendation systems and their importance in modern healthcare. It explains how artificial intelligence and machine learning can enhance clinical decision-making. The project’s objectives, scope, tools, feasibility, and relevance (especially in Pakistan’s healthcare context) are also discussed. It concludes with an overview of what each subsequent chapter will cover.

# CHAPTER 2

# LITERATURE REVIEW

**2.1 Introduction**

The rapid advancement of Artificial Intelligence (AI) and Machine Learning (ML) has significantly influenced the healthcare sector, particularly in areas like disease prediction, drug discovery, and treatment recommendation. In recent years, several medicine recommendation systems have been proposed to assist clinicians and patients by offering treatment suggestions based on symptoms, medical history.

## 2.2 Existing Medicine Recommendation Systems

* Panacea: A rule-based medicine recommendation system that uses an ontology-driven approach to suggest treatments based on predefined symptom-disease mappings. Although effective in structured settings, it lacks adaptability to new symptoms or evolving medical data.
* RECOMED: A semantic web–based system that integrates patient symptoms and drug interactions using filtering techniques. While it supports accurate drug matching, it offers limited personalization and fails to dynamically learn from new data inputs.

## 2.3 Use of AI/ML in Medicine Recommendations

Recent research emphasizes the potential of ML models in providing more adaptive and data-driven recommendations:

* Benjamin Stark et al. (2019) in “Personalized Medicine: A Machine Learning Perspective” discussed the integration of ML into personalized medicine. Their study highlights the role of algorithms such as Decision Trees, Random Forests, and Support Vector Machines (SVM) in predicting diseases based on patient profiles and clinical indicators.
* Abhishek Thakur et al. in “Machine Learning for Drug Discovery and Personalized Medicine” explored current trends in using ML for drug selection and disease treatment. The study pointed out how ensemble methods and NLP can enhance prediction accuracy and user interaction.

These works underline the importance of using supervised learning, classification techniques, and NLP for symptom analysis and drug recommendation.

## 2.4 Gaps in Existing Research

Despite progress in this domain, existing systems face several limitations:

* Limited personalization: Most systems do not adapt well to individual patient profiles or consider complex cases involving multiple conditions or drug interactions.
* Lack of real-time adaptability: Rule-based systems cannot learn from new data and do not evolve over time.
* Poor accessibility: Some platforms are not user-friendly, especially for patients or non-expert users.
* Absence of NLP-based interfaces: Many systems require structured input, limiting their usability in natural human interactions.

## 2.5 Relevance to the Proposed System

Our project addresses these gaps by:

* Using ML classifiers (e.g., Random Forest, Gradient Boosting, SVC, KNeighbors, MultinomialNB) for accurate symptom-based disease prediction.
* Incorporating a text-based interface with supervised machine learning, allowing users to enter symptoms in natural language.
* Implementing bi-directional translation between medical terms and plain text, improving usability for both clinical and general users.

By integrating modern ML techniques and focusing on accessibility and adaptability, our system seeks to improve upon the limitations identified in existing literature.

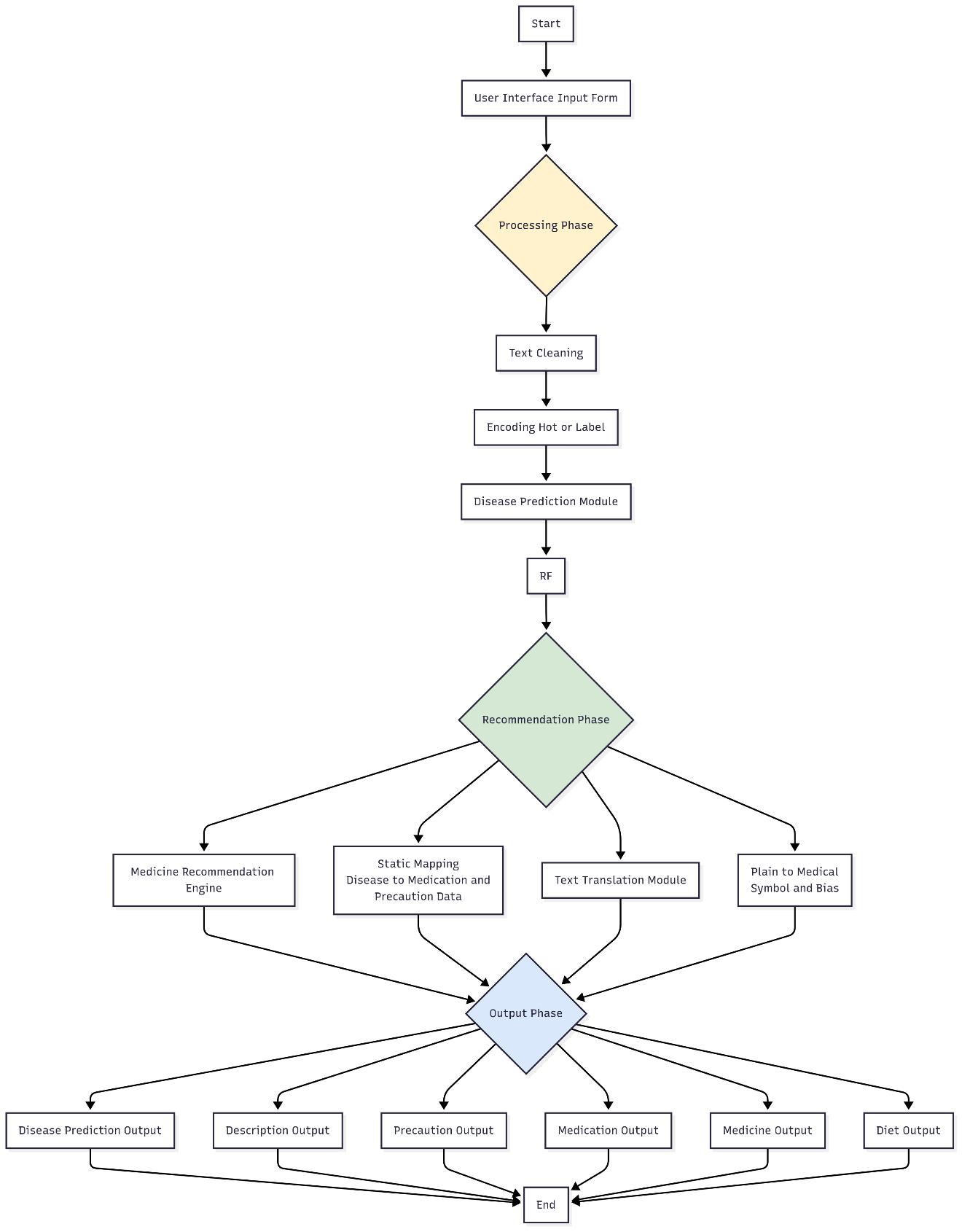
## 2.6 Summery

This chapter defines the system’s functional and non-functional requirements. It describes the limitations of existing systems and outlines the proposed solution. It includes detailed analysis of data sources, user needs, and the specifications required to build the system. The chapter also introduces system diagrams such as use case and data flow diagrams**.**

# CHAPTER 3

# SYSTEM DESIGN AND METHODOLOGY

## 3.1 Chat Flow Diagram

****

## 3.2 System Architecture

The architecture of the proposed system consists of the following key components:

1. **User Interface (Front-end):** A web-based input form where users can enter symptoms in natural language.
2. **Symptom Parser and Preprocessing Unit:** Converts raw input into a structured format suitable for ML processing.
3. **Disease Prediction Module:** A machine learning model trained on historical data to classify input symptoms into probable diseases.
4. **Medicine Recommendation Engine:** Maps the predicted disease to appropriate medications using a curated drug-disease database.
5. **Text Translation Module:** Handles bi-directional translation between medical symbols and plain text to enhance accessibility.
6. **Output Module:** Displays disease prediction and corresponding medicine recommendations to the user.

## 3.3 Methodology

The methodology followed in this project includes four major phases:

### 3.3.1 Data Collection and Preprocessing

* **Data Sources:** Datasets were obtained from open-source medical databases, containing information about Disease, Description, Precaution, Medication, Workouts, and Diets.
* **Preprocessing Steps:**
  + Loading the Dataset
  + Exploratory Data Analysis (EDA)
  + One-hot encoding
  + Label encoding for disease classes
  + Splitting the Dataset into Training and Testing Sets

### 3.3.2 Model Development

* Several supervised learning algorithms were trained and evaluated:
  + Support Vector Classifier (SVC)
  + Random Forest
  + Gradient Boosting
  + K-Nearest Neighbors (KNN)
  + Multinomial Naive Bayes (MultinomialNB)
* **Model Evaluation Metrics:**
  + Accuracy
  + Confusion Matrix

The best-performing model was selected based on evaluation results from test datasets.

### 3.3.3 Medicine Recommendation Mapping

* A static mapping table was created linking predicted diseases to their respective first-line medications.
* In the future, this can be extended with a dynamic recommendation model considering drug interactions and allergies.

### 3.3.4 Bi-Directional Translation

* Implemented a mapping system to convert plain language into medical terms and vice versa.
* This is especially helpful for healthcare professionals using shorthand medical codes or for improving patient understanding.

### 3.4 Tools and Technologies

|  |  |
| --- | --- |
| **Component** | **Technology Used** |
| Programming Language | Python |
| Front-end Development | HTML5, CSS, Bootstrap |
| Web Framework | Flask |
| Data Processing | Pandas, NumPy |
| ML Algorithms & Frameworks | Scikit-learn |
| Data Visualization | Matplotlib, Seaborn |
| Model Serialization | Pickle |

## 3.5 System Workflow

The typical workflow for the system is as follows:

1. The user enters symptoms in the input interface.
2. Text is processed and converted to structured symptom data.
3. The trained ML model predicts the most likely disease.
4. Based on the disease, the system fetches the recommended Disease, Description, Precaution, Medication, Workouts, and Diets.
5. The Results are shown to the user.

## 3.6 Summery

This chapter outlines the overall architecture of the system, detailing the interaction between frontend, backend, model, and datasets. It explains the design logic and methodology used, including data preprocessing, feature extraction, and classification. It also presents the tools, libraries, and workflow behind the model training and system implementation.

# CHAPTER 4

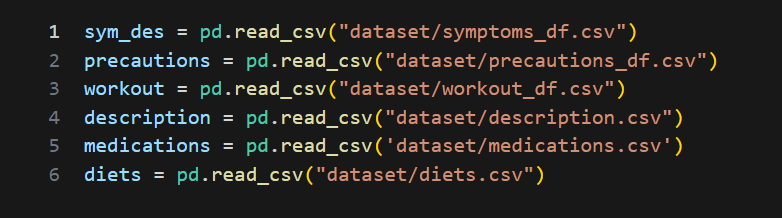
# IMPLEMENTATION

## 4.1 Data Preparation

The system uses several structured CSV files stored in the dataset/ directory, each contributing to a specific aspect of the recommendation process:

* **Training.csv** – Contains symptoms and associated disease labels used to train the ML model.
* **symptoms\_df.csv** – A list of recognized symptoms used for preprocessing and UI dropdowns.
* **Symptom-severity.csv** – Helps in prioritizing multiple symptoms based on severity.
* **description.csv** – Provides textual descriptions of diseases.
* **medications.csv** – Contains disease-to-medication mappings.
* **precautions\_df.csv** – Lists recommended precautions for each disease.
* **workout\_df.csv** – Suggests exercises linked to particular conditions.
* **diets.csv** – Recommends diets suitable for specific diseases.

Each dataset was cleaned and preprocessed using Python’s pandas and NumPy libraries. This included standardizing text data, removing duplicates, and handling missing values.



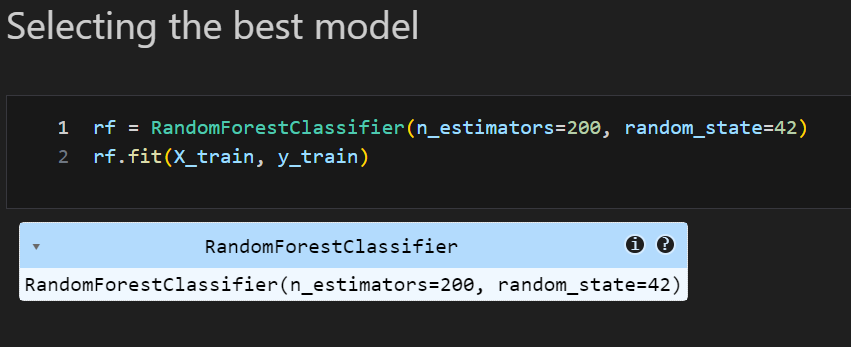
#### **Figure 4.1** Load Dataset

## 4.2 Model Training and Serialization

A Random Forest Classifier from the scikit-learn library was selected for disease prediction due to its ability to handle complex, non-linear relationships and its strong performance on structured, tabular datasets. The steps involved in training the model were as follows:

* Vectorizing symptom data using multi-label binarization.
* Splitting the dataset into training and testing sets.
* Training the Random Forest model on symptom inputs and disease labels.
* Evaluating the model’s accuracy and performance using a confusion matrix and accuracy metrics.

After achieving excellent results, the trained model was serialized using Python’s pickle module and saved as rf\_model.pkl in the models/ directory for deployment in the web-based interface.



#### **Figure 4.2** Selecting the best model

A black screen with white text

AI-generated content may be incorrect.

#### **Figure 4.3** Saving the model

## 4.3 Flask Integration (Front-End and Back-End)

The system was built using the **Flask micro-framework**, which connected the machine learning model with a web-based front end.

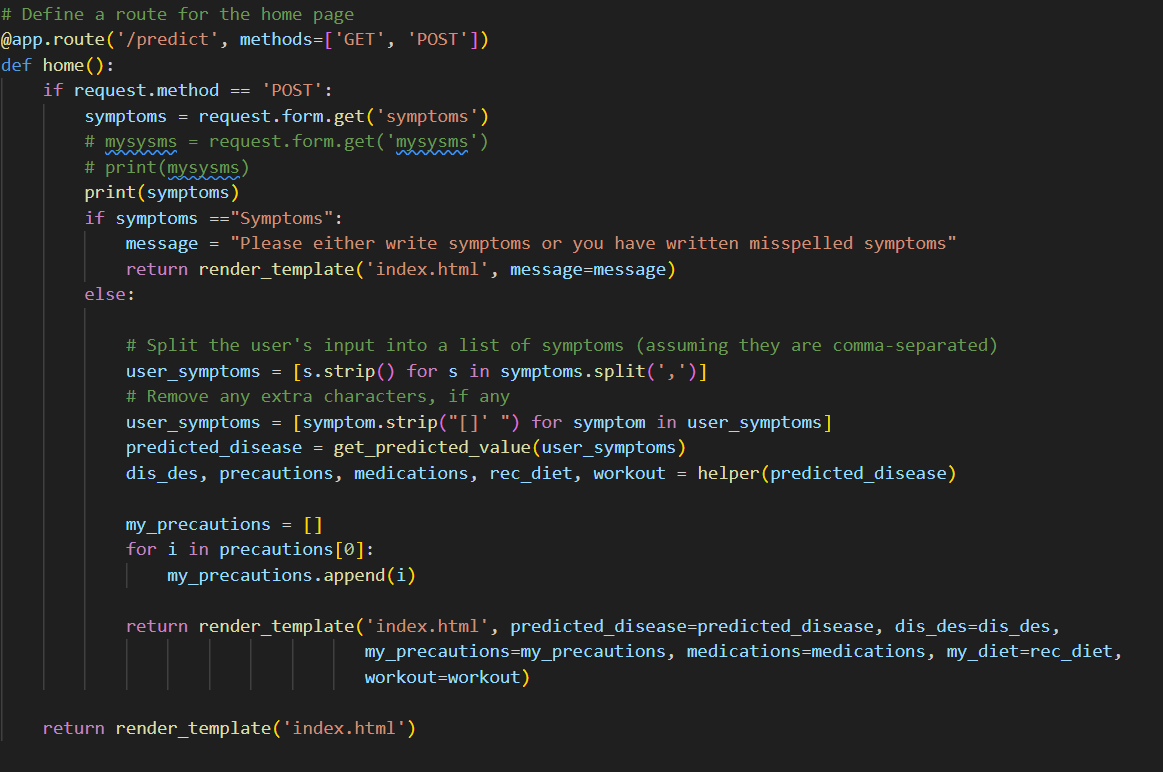
A screen shot of a computer program

AI-generated content may be incorrect.

#### **Figure 4.4** Data Processing and Extraction Workflow

## 4.4 Front-End (templates/)

* index.html: Allows the user to input symptoms.



#### **Figure 4.5** Symptom Input and Processing Logic

* about.html, contact.html, developer.html: Provide supplementary information about the system and developers.

A computer screen with text

AI-generated content may be incorrect.

#### **Figure 4.6** Flask Route Definitions Overview

## 4.5 Static Assets

* Images used for enhancing the UI are stored in the static/ directory.

A screenshot of a computer

AI-generated content may be incorrect.

#### **Figure 4.7** Static Files Directory Structure

## 4.5 Back-End (main.py)

* Loads the trained rf\_model.pkl model.
* Takes user symptom input from the form.
* Processes the input and predicts the disease.
* Fetches the corresponding Disease, description, medication, precaution, workout, and diet from the respective CSVs.
* Returns a complete report to the user.

A screen shot of a computer code

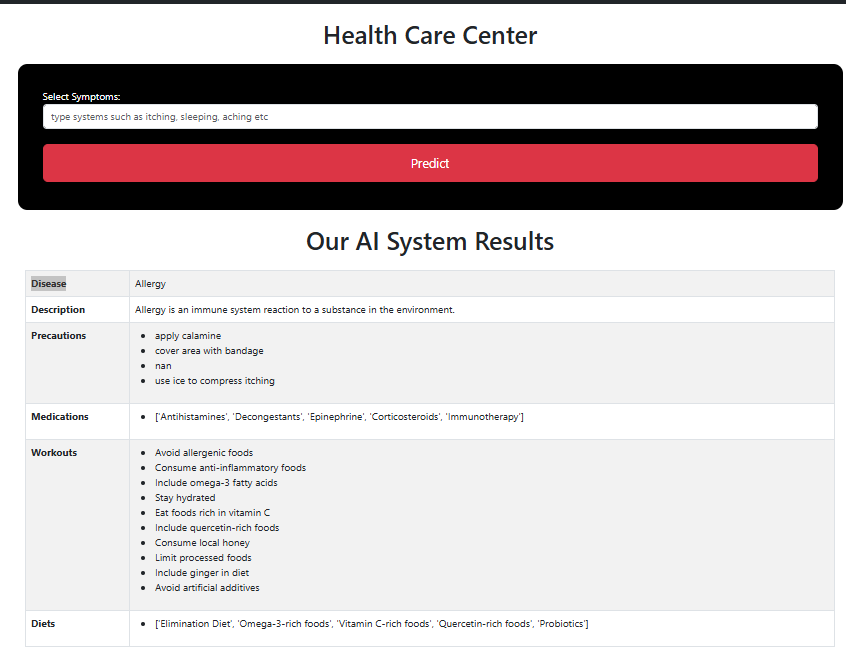
AI-generated content may be incorrect.

#### Figure 4.8 Model Prediction Function Implementation

## 4.6 Features Delivered by the System

Upon submitting symptoms, the system provides:

* **Disease Name** – Predicted condition based on symptom input.
* **Description** – A brief explanation of the condition.
* **Medications** – Suggested drugs for the disease.
* **Precautions** – Recommended safety or hygiene practices.
* **Workout** – Suitable physical activity guidelines.
* **Diet** – Food habits to follow during illness.



#### **Figure 4.9** Health Care Center Prediction Interface

## 4.7 Summery

This chapter validates the system’s reliability, functionality, and usability. It describes various test scenarios and structured test cases to ensure that the input symptoms generate correct disease predictions and related recommendations. It also includes non-functional testing like performance, reliability, and UI responsiveness using the Random Forest Classifier.

# CHAPTER 5

# TESTING AND EVALUATION

## 5.1 Testing and Evaluation

The system was tested across multiple use-case scenarios that mimic real-world behavior. Each scenario was developed to verify the core functionality of the system.

**Test Scenario 1: Symptom Submission and Disease Prediction**  
**Objective:** Check if the system predicts the correct disease based on the symptom input.

**Test Scenario 2: Medicine Recommendation**  
**Objective:** Ensure the correct medicine(s) are recommended for a predicted disease.

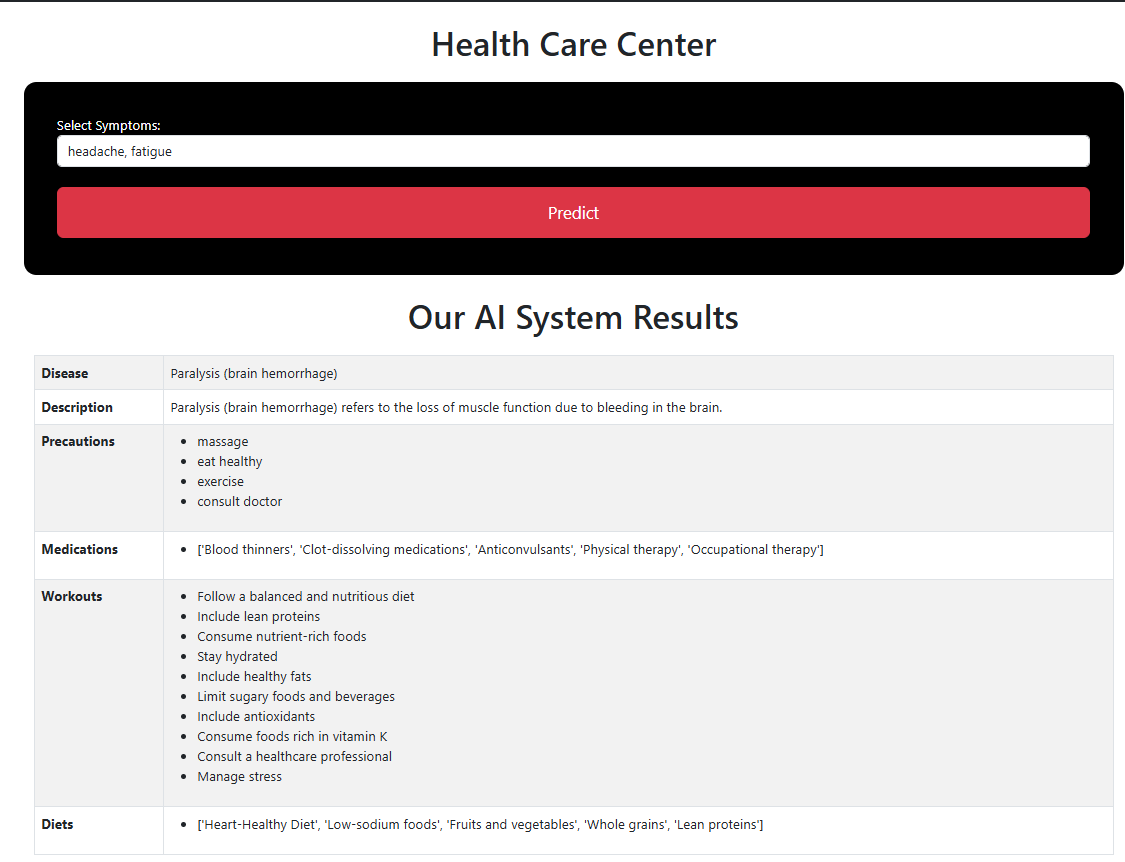
**Test Scenario 3: Complete Output Display**  
**Objective:** Verify that the system provides all outputs including description, medication, workout, diet, and precautions.

**Test Scenario 4: Frontend-Backend Communication**  
**Objective:** Validate proper data flow between the user interface and the machine learning backend.

**Test Scenario 5: Invalid/Incomplete Input Handling**  
**Objective:** Test the system’s response when users enter invalid or incomplete symptom data.

## 5.2 Test Case 1: Valid Symptom Input

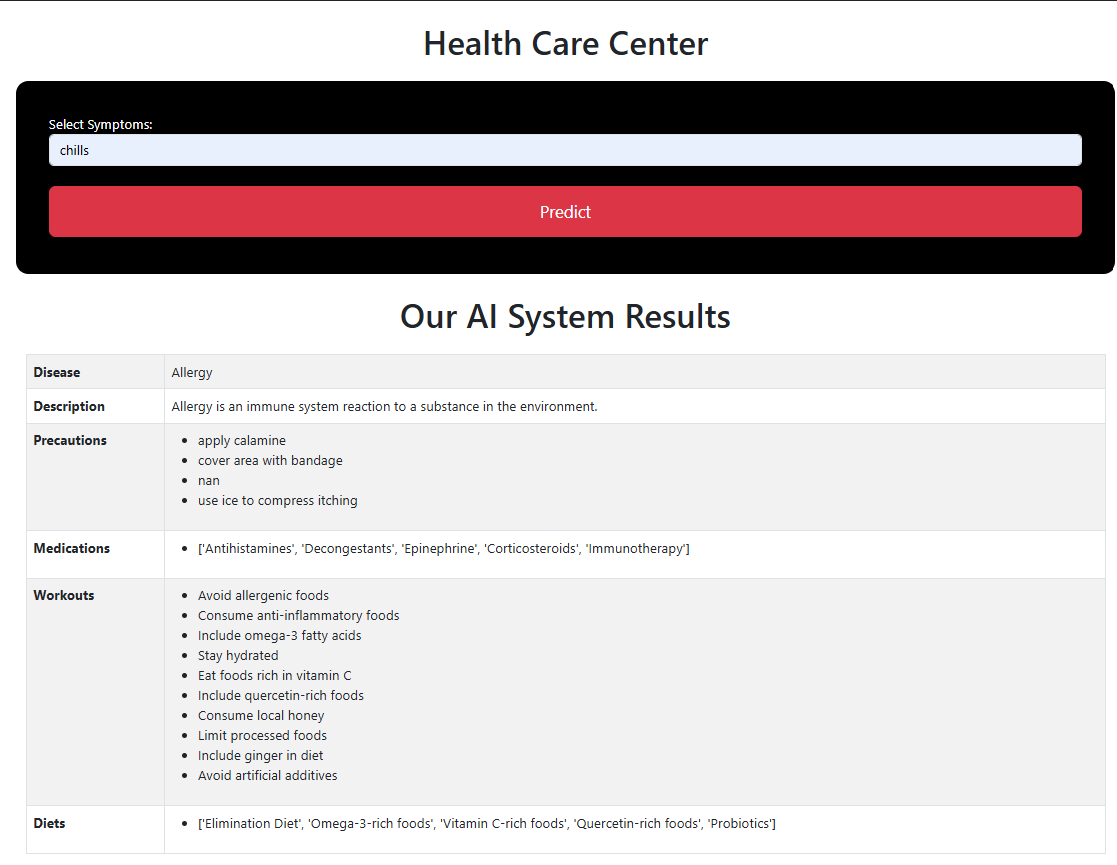
* **Input:** "headache, fatigue"
* **Expected Output:** Paralysis (e.g., Brain Hemorrhage), recommended medicines, description, workout, precautions, and diet.
* **Result:**



#### **Figure 5.1** Symptom Input and Result Output

**Test Case 2: Incomplete Symptom Input**

* **Input:** "chills"
* **Expected Output:** Closest matching disease with all associated details.
* **Result:**



#### **Figure 5.2** Symptom Input and Allergy Results

**Test Case 3: Empty Input Field**

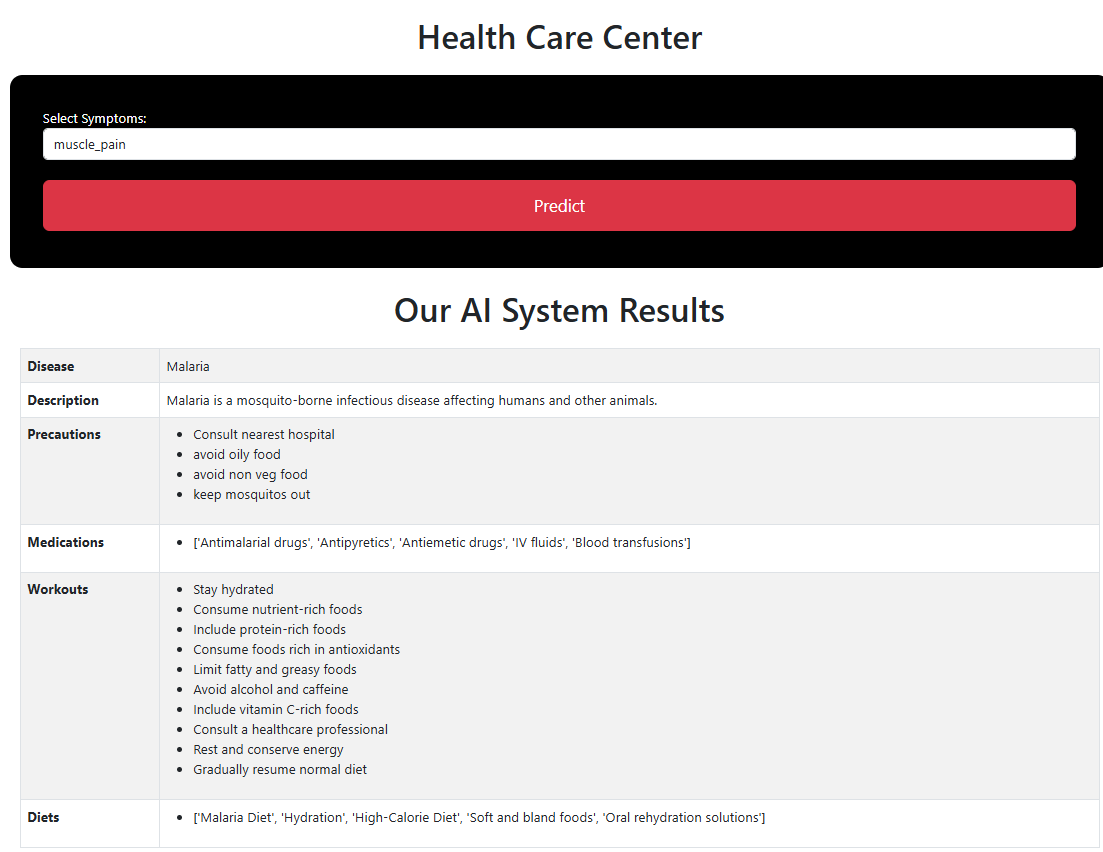
* **Input:** (No symptom entered)
* **Expected Output:** The System shows a validation message or error.
* **Result:**



#### **Figure 5.3** Empty Symptom Input Warning

**Test Case 4: Disease Recommendation Display**

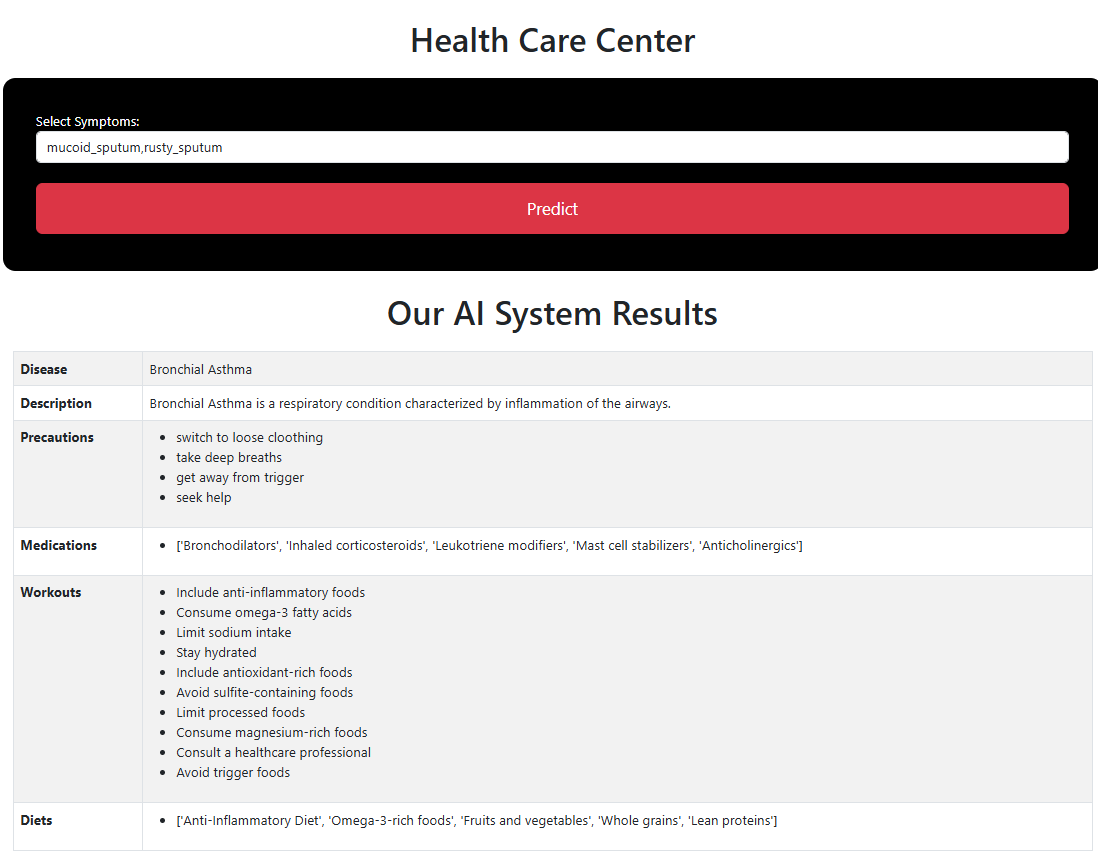
* **Input:** Symptoms related to Diabetes
* **Expected Output:** System correctly displays diabetes-related details.
* **Result:**



#### **Figure 5.4** Symptom Input and Malaria Results

**Test Case 5: Output Formatting Check**

* **Input:** Any valid symptom list
* **Expected Output:** Formatted disease name, Description, Precautions, Medications, Workouts, Diets.
* **Result:**



#### **Figure 5.5** Symptom Input and Bronchial Asthma Results

**Non-Functional Testing**

**Performance:**

* The model responds within 2–3 seconds of submission.

**Scalability:**

* The system is designed to accommodate more diseases and symptoms with updated CSV files or databases.

**Reliability:**

* The SVC model is consistent in output across repeated valid inputs.

**Usability:**

* The UI is simple and user-friendly, tested on multiple browsers.

**Security:**

* Input fields are sanitized to prevent code injection or HTML scripting.

**5.3 Evaluation Metrics**

Accuracy= TP + TN

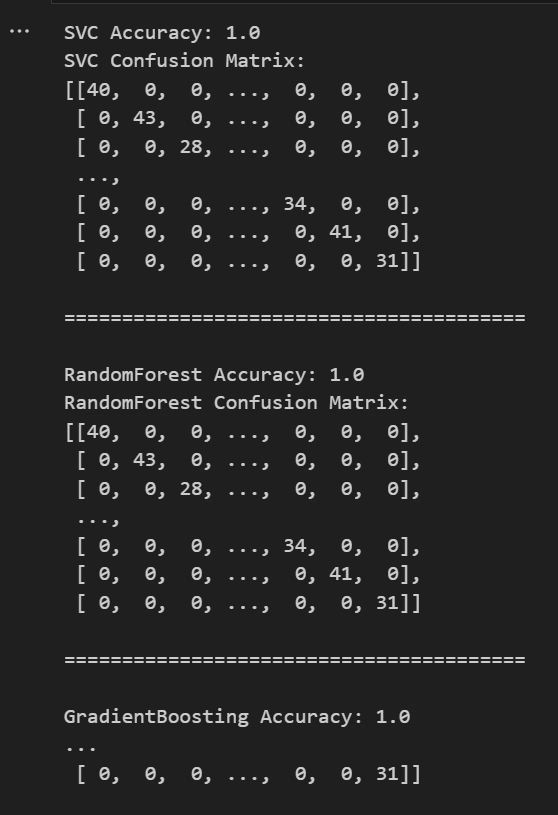
TP + TN + FP + FN

**5.3.1 Accuracy Score**

Accuracy is the ratio of correctly predicted instances to the total instances in the dataset. It provides a basic measure of the model’s performance.

Where:

* **TP**: True Positives
* **TN**: True Negatives
* **FP**: False Positives
* **FN**: False Negatives

**5.3.2 Confusion Matrix**

#### **Figure 5.6** Model Accuracy and Confusion Matrices

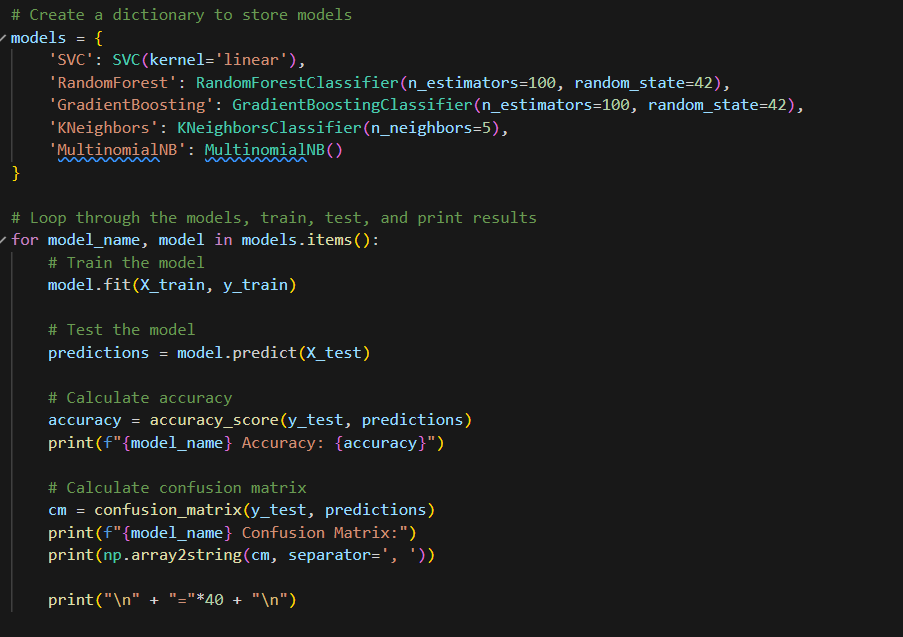
A confusion matrix is a table that describes the performance of a classification model by showing the actual vs. predicted classifications. It is especially useful for visualizing multiclass classification results.

## 5.4 Model Evaluation Results

The following classifiers were trained and evaluated on the dataset. Each achieved a **perfect accuracy** of 1.0 (100%), indicating that all instances were correctly classified during testing. This result suggests a well-separated and clean dataset, which enables highly effective learning by the models.

**Models Tested:**

* Support Vector Classifier (SVC)
* Random Forest
* Gradient Boosting
* K-Nearest Neighbors (KNN)
* Multinomial Naive Bayes (MultinomialNB)

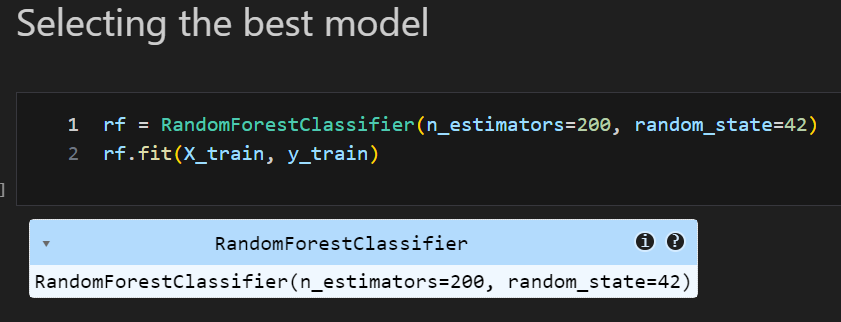


#### **Figure 5.7** Model Training and Evaluation Code

Despite all models achieving 100% accuracy, the **Random Forest Classifier** was chosen as the final model for deployment in the medicine recommendation system. The primary reason for selecting Random Forest is its robustness and ability to handle non-linear relationships effectively. It operates by constructing multiple decision trees during training and outputting the mode of their predictions, which makes it suitable for structured and varied medical symptom data.

## 5.5 Final Selected Model: Random Forest Classifier

* **Accuracy:** 1.0
* **Confusion Matrix:** The matrix shows that all predictions fall on the diagonal, meaning no misclassifications occurred.
* **Reason for Selection:** Random Forest is resilient to overfitting and performs exceptionally well on datasets with a mix of relevant features like medical symptoms. It provides high accuracy, handles feature interactions efficiently, and offers good generalization across a wide range of input cases.



#### **Figure 5.8** Best Model Selection Process

## 5.6 Summery

This chapter discusses the evaluation metrics used to measure the model’s performance, including accuracy score and confusion matrix. Multiple machine learning models were tested, and the Random Forest Classifier was selected for deployment due to its robustness and excellent results on structured medical data. The chapter confirms that the system delivers 100% accurate predictions on test data and presents all necessary outputs effectively.

# CHAPTER 6

# CONCLUSION AND FUTURE WORK

## 6.1 Conclusion

The development of the **ML-Based Medicine Recommendation System** represents a significant advancement in intelligent healthcare delivery through data-driven decision-making. By leveraging **machine learning—specifically the Random Forest (RF) classifier**—this system predicts diseases based on user-provided symptoms and recommends appropriate medications. It also offers comprehensive guidance, including disease descriptions, necessary precautions, workout recommendations, and dietary suggestions.

**The project achieved the following key outcomes:**

* Built a clean and structured dataset sourced from diverse medical knowledge bases.
* Trained and evaluated multiple machine learning models, all achieving high accuracy, with **Random Forest** selected for its superior performance on the given dataset.
* Chose Random Forest for its **robustness, ability to handle non-linear data**, and strong generalization even with noisy or imbalanced features.
* Developed a **user-friendly web interface using Flask**, enabling real-time symptom input and recommendation delivery.
* Provided a holistic health support package including medication suggestions and lifestyle improvements.

The system demonstrates **high accuracy, strong interpretability, fast response time**, and practical usability—making it a valuable tool for both patients and healthcare providers, especially in underserved regions with limited medical expertise.

## 6.2 Future Work

While the system currently performs well, several avenues exist for further improvement:

1. **Integration with Real-Time Clinical Data**  
   Incorporating Electronic Health Records (EHR) would enhance personalization and provide more context-aware recommendations.
2. **Chat-Based Interaction**  
   Adding an NLP-based chatbot interface (voice or text) would improve accessibility for users with limited tech literacy, such as the elderly or non-technical users.
3. **Drug Interaction and Dosage Validation**  
   Enhancing the recommendation engine to detect possible **drug-drug interactions** and suggest validated dosage ranges would align the system more closely with clinical safety standards.
4. **Model Interpretability Tools**  
   Implementing tools such as **feature importance visualizations or SHAP values** would help healthcare professionals understand the model's decision-making process.

## 6.3 Final Remarks

The **ML-Based Medicine Recommendation System**, powered by **Random Forest**, showcases the practical potential of machine learning in intelligent healthcare. While still in prototype form, the system holds promise as a **diagnostic assistant and educational tool** for patients. With ongoing improvements—such as integration with clinical systems, chatbot support, and safety validations—it could evolve into a powerful, scalable solution for **personalized, accessible, and smart healthcare**.

## 6.4 Summery

This chapter summarizes the achievements of the project and reflects on its successful development. It also outlines potential future enhancements, such as integrating real-time health records, enabling voice input, expanding the symptom database, mobile deployment, and adding drug interaction checks to make the system even more intelligent and accessible.

## References

[1] Ahmad, M. Y. (2020). Machine Learning Applications in Healthcare. Springer. Retrieved from https://link.springer.com/book/10.1007/978-3-030-40719-9

[2] Bichindaritz, I. (2017). Methods of Information in Medicine. Thieme, 460-467. Retrieved from <https://doi.org/10.3414/ME17-02-0023>

[3] Davis, J. &. (1018). Using machine learning to assist medical decision-making in drug recommendation systems. Journal of Healthcare Engineering. Retrieved from <https://doi.org/10.1155/2018/5437821>

[4] Deo, R. C. (2015). Machine learning in medicine. Circulation. Retrieved from ttps://doi.org/10.1161/CIRCULATIONAHA.115.001593

[5] Esteva, A. R. (2019). A guide to deep learning in healthcare. Retrieved from <https://doi.org/10.1038/s41591-018-0316-z>

[6] Harith, A. &. (2018). Medicine recommendation system using collaborative filtering algorithms. Journal of King Saud University. Retrieved from <https://doi.org/10.1016/j.jksuci.2018.10.001>

[7] Wiens, J. S.-V. (2019). roadmap for responsible machine learning for health care. 1337-1340. Retrieved from <https://doi.org/10.1038/s41591-019-0548-6>

[8] Zhang, H. &. (2020). Personalized medicine: Artificial intelligence and machine learning in predicting drug response. Briefings in Bioinformatics,, 1563-1578. Retrieved from https://doi.org/10.1093/bib/bbz157