AI Powered Multi-Disease Diagnostic System

A PROJECT REPORT

Submitted by

HARSH PATEL [21BECE30191]

VISHVA PATEL [21BECE30242]

VRAJ PATEL [21BECE30244]

CHARMI RUPARELIYA [21BECE30278]

In fulfillment for the award of the degree

of

BACHELOR OF ENGINEERING

in

Computer Engineering

LDRP Institute of Technology and Research, Gandhinagar

Kadi Sarva Vishwavidyalaya

April, 2025

AI Powered Multi-Disease Diagnostic System

A PROJECT REPORT

Submitted by

HARSH PATEL [21BECE30191]

VISHVA PATEL [21BECE30242]

VRAJ PATEL [21BECE30244]

CHARMI RUPARELIYA [21BECE30278]

In fulfillment for the award of the degree

of

BACHELOR OF ENGINEERING

in

Computer Engineering

LDRP Institute of Technology and Research, Gandhinagar

Kadi Sarva Vishwavidyalaya

April, 2025

LDRP INSTITUTE OF TECHNOLOGY AND RESEARCH GANDHINAGAR

CE-IT Department

CERTIFICATE

This is to certify that the Project Work entitled **"AI Powered Multi-Disease Diagnostic System"** has been carried out by **Harsh Patel (21BECE30191)** under my guidance in fulfilment of the degree of Bachelor of Engineering in Computer Engineering Semester-8 of Kadi Sarva Vishwavidyalaya University during the academic year 2024-2025.

Prof. Tejasvee Gupta	Dr. Ashishkumar Patel
Internal Guide	**Head of the Department**
LDRP ITR	**LDRP ITR**

LDRP INSTITUTE OF TECHNOLOGY AND RESEARCH

GANDHINAGAR

CE-IT Department

CERTIFICATE

This is to certify that the Project Work entitled **"AI Powered Multi-Disease Diagnostic System"** has been carried out by **Vishva Patel (21BECE30242)** under my guidance in fulfilment of the degree of Bachelor of Engineering in Computer Engineering Semester-8 of Kadi Sarva Vishwavidyalaya University during the academic year 2024-2025.

Prof. Tejasvee Gupta	Dr. Ashishkumar Patel
Internal Guide	**Head of the Department**
LDRP ITR	**LDRP ITR**

LDRP INSTITUTE OF TECHNOLOGY AND RESEARCH

GANDHINAGAR

CE-IT Department

CERTIFICATE

This is to certify that the Project Work entitled **"AI Powered Multi-Disease Diagnostic System"** has been carried out by **Vraj Patel (21BECE30244)** under my guidance in fulfilment of the degree of Bachelor of Engineering in Computer Engineering Semester-8 of Kadi Sarva Vishwavidyalaya University during the academic year 2024-2025.

Prof. Tejasvee Gupta	Dr. Ashishkumar Patel
Internal Guide	**Head of the Department**
LDRP ITR	**LDRP ITR**

LDRP INSTITUTE OF TECHNOLOGY AND RESEARCH

GANDHINAGAR

CE-IT Department

CERTIFICATE

This is to certify that the Project Work entitled **"AI Powered Multi-Disease Diagnostic System"** has been carried out by **Charmi Rupareliya (21BECE30278)** under my guidance in fulfilment of the degree of Bachelor of Engineering in Computer Engineering Semester-8 of Kadi Sarva Vishwavidyalaya University during the academic year 2024-2025.

Prof. Tejasvee Gupta	Dr. Ashishkumar Patel
Internal Guide	**Head of the Department**
LDRP ITR	**LDRP ITR**

ACKNOWLEDGEMENT

We would like to express our heartfelt gratitude to all those who have contributed to the successful completion of our **AI Powered Multi-Disease Diagnostic System** project. Without their support, guidance, and encouragement, this project would not have been possible.

First of all we are thankful to the **LDRP Institute of Technology and Research** for undertaking this project. We would like to extend our deepest appreciation to **Prof. Tejasvee Gupta**, our project supervisor, for providing us with invaluable guidance throughout the project. Your expertise, patience, and continuous support were instrumental in shaping our project and enhancing our understanding of AI Powered Multi Disease Diagnostic System.

We would like to express our gratitude to our classmates and friends who provided us with moral support and valuable suggestions during the course of this project.

With Regards,

Harsh Patel (21BECE30191)
Vishva Patel (21BECE30242)
Vraj Patel (21BECE30244)
Charmi Rupareliya (21BECE30278)

ABSTRACT

Improving patient outcomes and reducing mortality rates in the current medical environment depend heavily on timely and precise disease detection. However, traditional diagnostic techniques can take a lot of time and specialised expertise, which delays treatment. By creating a multi-disease detection system that makes use of machine learning and deep learning approaches, this project offers a solution to this problem. To identify a number of diseases, including diabetes, heart attacks, Parkinson's disease, breast cancer, brain tumor and tuberculosis, the system examines a variety of patient data sources. The technology offers very accurate real-time diagnostic predictions. This tool will empower healthcare professionals to make faster, more informed decisions and enhance accessibility to diagnostic tools in remote areas.

LIST OF FIGURES

LIST OF TABLES

TABLE OF CONTENTS

1 INTRODUCTION

- **INTRODUCTION**
- **AIMS AND OBJECTIVE OF WORK**
- **BRIEF LITERATURE REVIEW**
- **PROBLEM DEFINITION**

1.1 Introduction

In order to improve outcomes for patients and reduce the burden on healthcare systems, the healthcare industry is dealing with an increasing demand for precise and timely disease detection. Current methods for diagnosis can take a lot of time and specialised medical knowledge, which can cause treatment delays and worse patient results. In addition, a more effective, scalable, and automated method of illness detection is required due to the global increase in diseases including Parkinson's disease, heart attacks, diabetes, breast cancer, brain tumor and TB. By creating a multi-disease identification system that makes use of machine learning and deep learning techniques, this study aims to overcome these issues. In order to detect multiple diseases simultaneously, the system is designed to analyse a variety of data sources, including lab results and patient histories. The technology provides real-time, precise diagnostic predictions through the integration of advanced algorithms for prediction, empowering healthcare professionals to make quicker and more accurate decisions. The goal of the project is to enhance the treatment of patients and facilitate more effective healthcare delivery.

1.2 Aims and objective of the work

Develop a Multi-Disease Detection System:

- To build a system that can predict multiple types of diseases, such as diabetes, heart attacks, Parkinson's disease, breast cancer, brain tumor and TB, by evaluating different patient data, such as test results and medical records.

Implement Machine Learning and Deep Learning Models:

- Process and analyse medical data using advanced ML and DL algorithms to ensure accuracy in disease prediction.

Real-time Disease Prediction:

- Facilitate quick decision-making in medical facilities by offering predictions in real-time with minimal delay.

Improve Healthcare Efficiency:

- Reduce the time spent on diagnosis and boost the overall effectiveness of healthcare delivery by enhancing healthcare professionals' ability to diagnose various diseases more quickly.

Encourage Data-Driven Healthcare:

- Improve healthcare systems by offering an accurate and effective data-driven method of disease diagnosis.

1.3 Brief Literature Review

The healthcare sector has undergone a transformation due to machine learning and deep learning, which make it possible to diagnose diseases accurately and efficiently. Several studies demonstrate its capacity to evaluate intricate medical data, such as imaging data, lab test findings, and patient histories. These developments make diagnostic tools more widely available and effective by addressing important issues including time restrictions and the need for specialised expertise.

Current studies focus on multi-disease diagnostic systems that diagnose diseases including Parkinson's disease, heart attacks, diabetes, breast cancer, brain tumor and TB using a variety of data sources. In order to improve model performance, data preprocessing methods are essential. According to studies, well selected datasets are essential for reliable and efficient model training.

Real-time applications in distant and resource-constrained environments are still limited, despite progress. According to research, scalable systems that offer prompt predictions and aid medical professionals in making decisions have to be developed. AI Powered Multi-Disease Diagnostic Systems can greatly enhance healthcare outcomes and lower mortality rates by addressing these gaps.

1.4 Problem Definition

The increasing prevalence of diseases such as Parkinson's Disease, heart attacks, diabetes, brain tumors, breast cancer, and tuberculosis necessitates a timely and accurate diagnostic approach to reduce mortality rates and improve patient outcomes. However, traditional diagnostic methods are often time-consuming, require specialized expertise, and may not be accessible in remote or resource-constrained areas. There is a critical need for a scalable, efficient, and automated diagnostic system that can simultaneously detect multiple diseases from diverse data sources such as medical images, lab reports, and patient histories. This project aims to develop an AI Powered Multi-Disease Diagnostic System using machine learning (ML) and deep learning (DL) techniques to provide real-time, high-accuracy predictions, enabling healthcare professionals to make faster, more informed decisions and enhance accessibility to diagnostic tools in underserved regions.

1.5 Plan of their work

1. **Establishing the Project and Initial Planning:**

   Project Initialization:

   - Configure the environment, tools, and project repository.

   Requirement analysis:

   - Precisely define the purposes and requirements of the system, including the types of diseases it must be able detect and the sources of its data such as lab reports, patient records.

   Data Collection:

   - To detect the diseases, collect relevant test results, patient records, and medical history data.

2. **Collecting and Processing Data:**

   Data Acquisition:

   - Acquire the data and remove or correct inaccurate, incomplete, or irrelevant records.Handle missing data through imputation or removal strategies.

- Used Datasets:

  Tuberculosis Dataset:
  https://www.kaggle.com/datasets/tawsifurrahman/tuberculosis-tb-chest-xray-datase

  Brain Tumor Dataset :-
  https://www.kaggle.com/datasets/masoudnickparvar/brain-tumor-mri-dataset

  Breast Cancer Dataset :-
  https://www.kaggle.com/datasets/aryashah2k/breast-ultrasound-images-dataset

  Diabetes Dataset:
  https://www.kaggle.com/datasets/naveen1729/diabetis

  Heart Dataset:
  https://www.kaggle.com/code/desalegngeb/heart-disease-predictions

  Parkinsons Dataset:
  https://www.kaggle.com/code/chanchal24/parkinson-s-disease-detection

Normalization/Standardization:

- Normalize or standardize numerical data to ensure consistency across input features.

Data Augmentation:

- For medical images, apply techniques like rotation, flipping, zooming, and cropping to increase dataset size and diversity, improving model robustness.

Labeling:

- Annotate or classify the data with disease labels based on diagnostic criteria or expert input . Ensure labels are consistent and accurate to train the model effectively.

3. Development of the Model:

Model Selection:

- Choose appropriate machine learning and deep learning architectures based on categories of diseases and accessible data.

Training:

- Build and train the chosen models using frameworks such as Scikit-learn or TensorFlow/Keras.
- Use strategies like data augmentation and hyperparameter tuning to increase accuracy.

Evaluation:

- Make use of a validation dataset to assess the model.
- Evaluate the model's accuracy, precision, recall, and other relevant metrics.

4. Analyzing and Improving:

Unit and Integration Testing:

- To verify system functionality, thoroughly test each individual component and how they work together.

User Acceptance Testing:

- Test the system with medical experts to evaluate its usability and performance.
- To enhance user experience and system performance, gather feedback.

5. Training and Documentation:

Documentation:

- Provide technical documents, training materials, and user manuals for healthcare professionals.

Training Sessions:

- Organize training sessions for end users and stakeholders to familiarize them with the system's features.

6. Future Improvements (Continued):

Front-end Development:

- For displaying results, create an interactive front-end with tools like Streamlit or React.js. Provide intuitive user interfaces so that medical professionals may view predictions and enter data.

Back-end Development:

- Set up back-end infrastructure using Node.js to handle API requests, manage user interactions, and integrate with the ML models.

Continuous Improvement:

- Use machine learning strategies to keep improving the model based on user feedback and new data.

This approach ensures that the AI Powered Multi Disease Diagnostic System is created, evaluated, and implemented for use in a systematic way, successfully satisfying the demands of medical professionals and improving the availability of diagnostic resources in remote areas.

2 TECHNOLOGY AND LITERATURE REVIEW

- **TOOLS AND TECHNOLOGY**
- **PROJECT SCHEDULING**

2.1 Tools and Technology

Development Frameworks and Libraries:

- **TensorFlow/Keras:** Popular deep learning frameworks for model construction and training. While Keras gives a high-level API to facilitate the design and experimentation of models more easily, TensorFlow offers a rich environment for building machine learning models.
- **PyTorch:** An alternative to TensorFlow that is well-liked for its adaptability and simplicity of usage, particularly in research environments, is PyTorch. Dynamic computation graphs are made possible by PyTorch, which is useful for testing and training models.
- **Scikit-learn:** A library for putting classic machine learning techniques like support vector machines, random forests, and decision trees into practice.

Data Processing and Analysis Tools:

- **Pandas:** A data manipulation package that cleans, analyses, and formats data so that it can be used to train models.
- **NumPy:** A library that supports massive, multi-dimensional arrays and matrices for numerical computation.
- **Matplotlib:** Data visualisation libraries that make it possible to create graphs that show data distributions and model performance.

Version Control and Development Tools:

- **Git/GitHub:** To facilitate collaboration and version control, make sure that model code and associated scripts are properly maintained and distributed among team members.
- **Jupyter Notebook:** Jupyter Notebooks are useful for developing and testing models prior to Stream-lit deployment, albeit they are mostly utilized for exploratory data analysis and model training.
- **Google Colab:** Python code can be written and run in a collaborative setting with Google Colab, a cloud-based platform.

- **VS Code/ Pycharm:** Integrated Development Environments (IDEs) for writing and managing code.

Frontend:

- **React.js:** To create a responsive and user-friendly interface.

Backend:

- **FastAPI:** For building fast and efficient REST APIs with automatic documentation.
- **Node.js:** For scalability and concurrent backend operations.

2.2 Project Scheduling

Project scheduling involves separating the total work in a project into separate activities and judging the time required to complete these activities. Usually, some of these activities are carried out in parallel.

2.2.1 Work Breakdown Structure

- Work Breakdown Structure is used to decompose a given task set recursively into small activity [Fig 2.2.1].

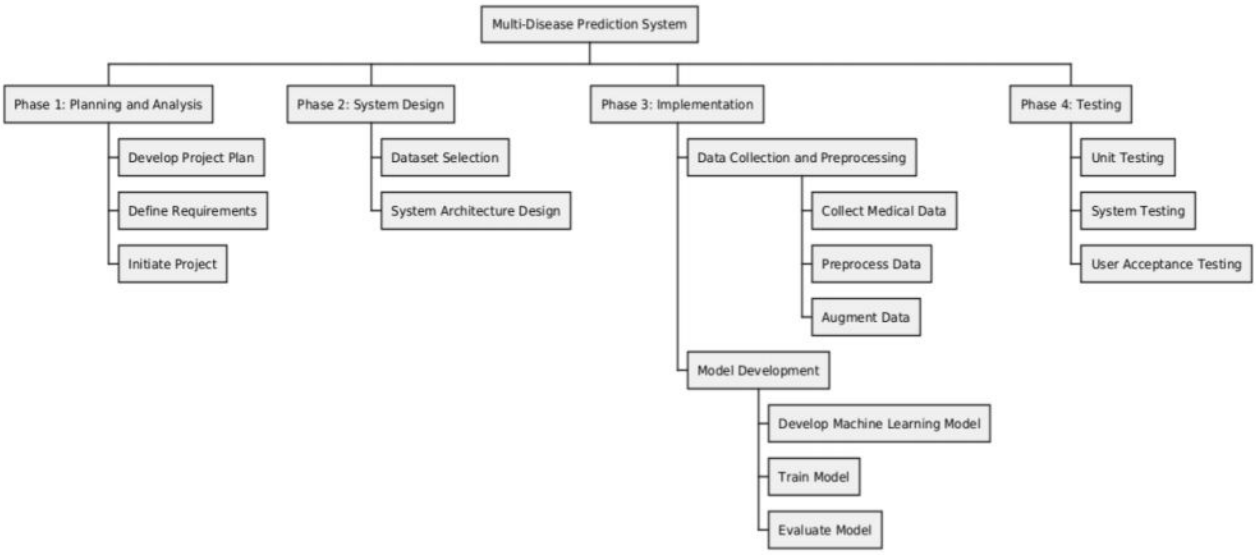

Fig 2.2.1 - Work Breakdown Structure

3 SYSTEM REQUIREMENTS STUDY

- **USER CHARACTERISTICS**
- **HARDWARE AND SOFTWARE REQUIREMENTS**
- **ASSUMPTIONS AND DEPENDENCIES**

3.1 User Characteristics

Primary Users:

- **Patients:** Patients come from a wide range of demographic backgrounds, including different ages, genders, and places of residence. Since many of them might not be medically literate, the system interface needs to be clear, easy to use, and educational. These people may have a medical history of diabetes, heart disease, or other problems, or they may have early-stage, undetected symptoms.
- **Healthcare Professionals:** To handle their workload, medical professionals including physicians and specialists need a system that offers prompt and precise diagnostic assistance. For improved treatment planning, they need tools to upload and analyse patient data, medical picture visualisations, and comprehensive, comprehensible outcomes.
- **Diagnostic Lab Technicians:** Lab technicians deal with imaging data and medical reports. To assist physicians and patients with precise diagnosis, they want tools for uploading numerous reports, integrating with lab equipment, and guaranteeing data quality.

3.2 Software and Hardware Requirements

Software and Hardware Requirements are used to describe the minimum hardware and software requirements to run the Software. These requirements are described below.

3.2.1 Software Requirements

Development Frameworks and Libraries:

- **TensorFlow/Keras:** For developing and refining deep learning models. Keras offers a high-level API to streamline model creation, whereas TensorFlow is an extremely potent open-source toolkit.
- **PyTorch:** An alternative deep learning framework noted for its dynamic computation graphs and ease of use in research.
- **Scikit-learn:** For additional machine learning algorithms and model evaluation.

Data Processing and Visualization:

- **Pandas:** For data manipulation and preprocessing.
- **NumPy:** For numerical operations and array manipulations.
- **Matplotlib/Seaborn:** For data visualization and plotting

Version Control and Development Tools:

- **Jupyter Notebook:** For interactive development, experimentation, and visualization of model training processes.
- **Git/GitHub:** To facilitate collaboration and version control, make sure that model code and associated scripts are properly maintained and distributed among team members.
- **VS Code/ Pycharm:** Integrated Development Environments (IDEs) for writing and managing code.
- **Google Colab:** Python code can be written and run in a collaborative setting with Google Colab, a cloud-based platform. For deep learning and machine learning model training .Google Drive connection with Colab makes it easy for team members to share and manage data.

Frontend:

- **React.js:** To create a responsive and user-friendly interface.

Backend:

- **FastAPI:** For building fast and efficient REST APIs with automatic documentation.
- **Node.js:** For scalability and concurrent backend operations.

3.2.2 Hardware Requirements

Computing Power:

- **High-performance CPU:** A multi-core CPU (such as the AMD Ryzen 7/9 or Intel i7/i9) for effective computing, particularly for data preprocessing and model training.

- **GPU:** Since we are using Google Colab, the project will utilize its built-in GPU capabilities, such as T4, which are provided for free or with paid plans for more advanced options. These GPUs are sufficient for training deep learning models and processing large datasets.
- **RAM:** 12 GB or more of RAM is advised for managing huge datasets and complicated models, although at least 8 GB is sufficient.

Storage:

- **SSD:** Multi-core processor with sufficient processing power.
- **External Storage:** Additional storage solutions for backup and archiving of datasets and models.

Additional Hardware:

- **External Mouse and Keyboard:** Peripherals with ergonomic design to boost output during extended development sessions.

It is possible to efficiently design, train, and test the model while maintaining project efficiency and performance by making sure that these hardware and software requirements are satisfied.

3.3 Assumptions and Dependencies

3.3.1 Assumptions

- It is assumed that for training and validation, there exist enough and precise datasets for conditions like Parkinson's, heart attacks, diabetes, brain tumours, breast cancer, and tuberculosis.
- It is assumed that The collected datasets are properly labelled and cleaned to ensure that there are minimal errors or inconsistencies.
- Perhaps the selected machine learning and deep learning models developed will achieve satisfactory accuracy based on the available data.
- Health care professionals will accept the system and use it to detect diseases and make decisions.

- The hardware (CPUs, GPUs, RAM) that is available is thought to be adequate for training the model in a fair amount of time. It is expected that the necessary computing resources will be allocated while employing cloud resources.
- The models will generalize well across different populations and medical settings, avoiding overfitting to specific datasets.

3.3.2 Dependencies

- The disease detection models in the system are trained and validated using high-quality, labelled datasets.
- The availability and correct operation of computer vision libraries (OpenCV) and machine learning frameworks (TensorFlow/Keras, PyTorch) are prerequisites for the model's effective training.
- Dependencies include data processing and visualization tools (Pandas, NumPy) and integrated development environments (IDEs) such as VS Code or PyCharm.
- Dependence on sufficient computing resources for effective model training and experimentation, such as powerful GPUs and RAM.
- The system’s operation is dependent on meeting healthcare regulations for managing patient data securely.

4 SYSTEM DIAGRAMS

- **CLASS DIAGRAM**
- **USE CASE DIAGRAM**
- **SEQUENCE DIAGRAM**
- **ACTIVITY DIAGRAM**

4.1 Class Diagram

The AI Powered Multi Disease Diagnostic System's class diagram displays its primary constituents: the System, End-user, Medical Data, Prediction Model, and Prediction Outcome. Attributes and operations are included in each class. Its crucial importance is highlighted by the Composition relationship that connects the Medical Data class to the System.

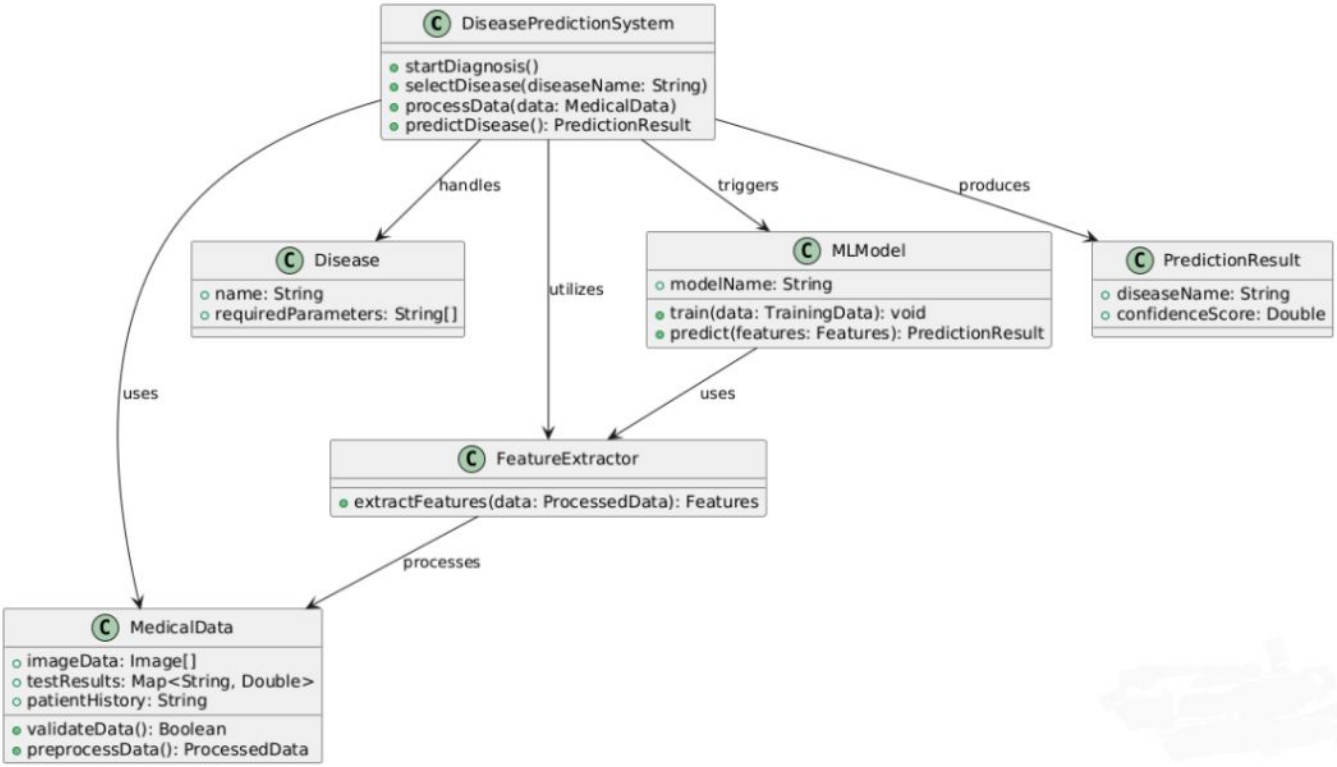

Fig 4.1 - Class diagram

4.2 Use Case Diagram

The AI Powered Multi Disease Diagnostic System's use case diagram illustrates how the patient and system administrator communicate. While the Administrator maintains datasets and trains algorithms, the Patient provides medical data for analysis. Data management, illness prediction, model training, and feedback gathering to increase system accuracy are important use cases. With feedback, the system continuously improves its performance while processing inputs to forecast diseases.

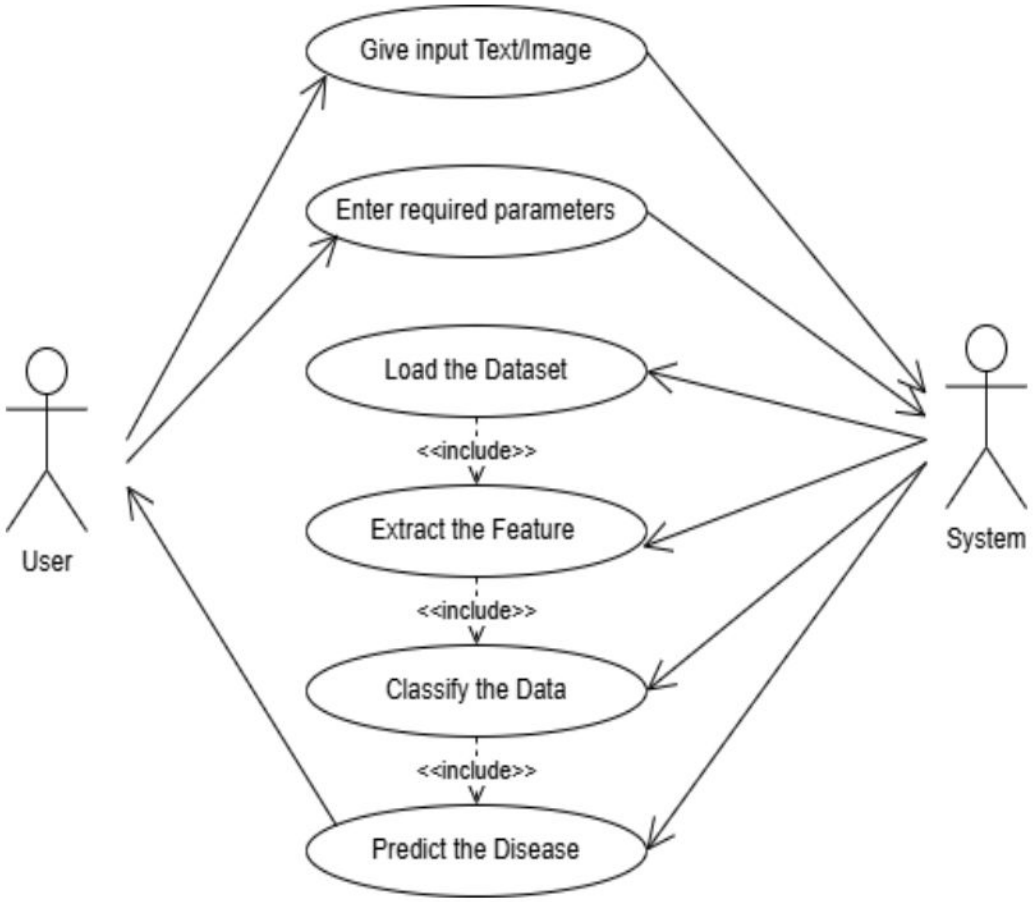

Fig 4.2 - Use case diagram

4.3 Sequence Diagram

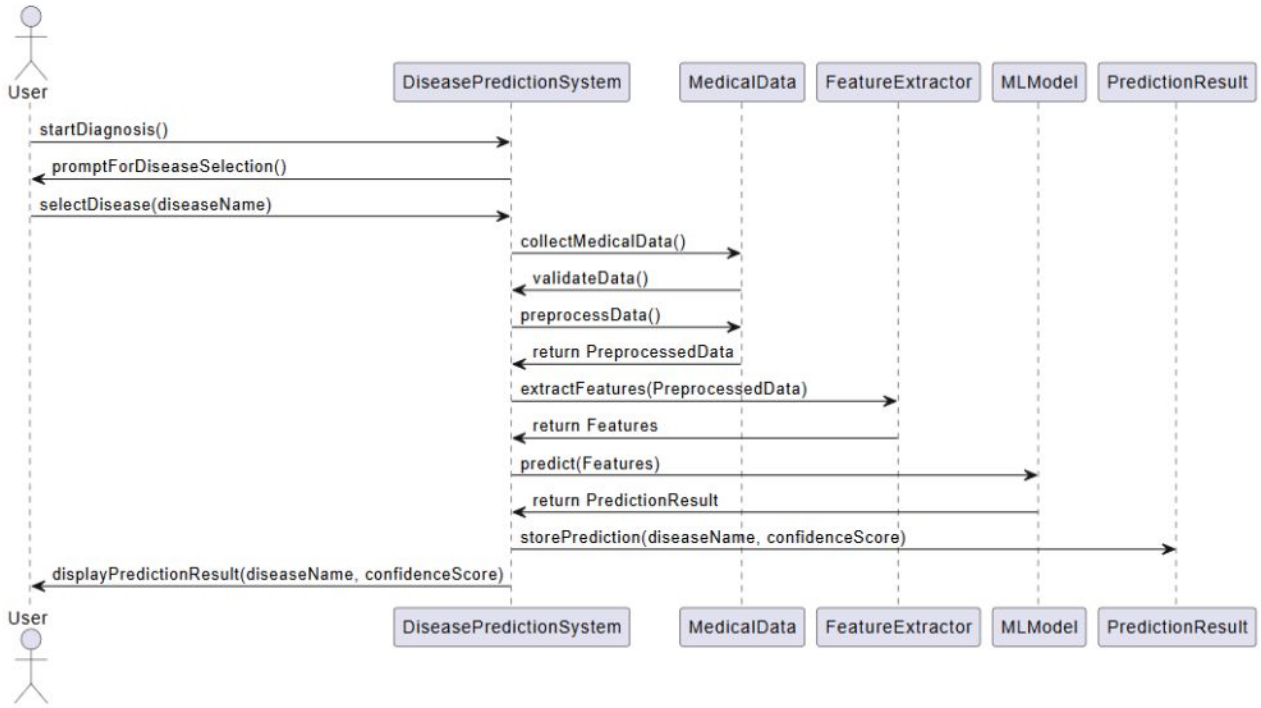

Fig 4.3 - Sequence diagram

4.4 Activity Diagram

An activity diagram is a special case of a state diagram in which all of the states are action states and in which all of the transitions are triggered by completion of the actions in the source states. Below are the activity diagrams for the actions performed by the end-user and the response of the system.

4.4(i) Activity diagram for User Input

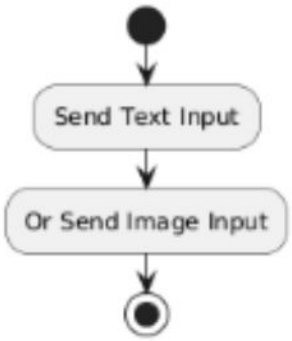

Fig 4.6(i) - Activity diagram for User Input

4.4(ii) Activity diagram for Data Pre-processing

Fig 4.6(ii) - Activity diagram for Data Pre-processing

4.4(iii) Activity diagram for Training Model and Feature Extraction

Fig 4.6(iii) - Activity diagram for Training Model and Feature Extraction

4.4(iv) Activity diagram for Disease Prediction

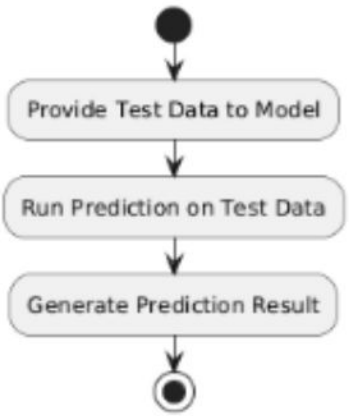

Fig 4.6(iv) - Activity diagram for Disease Prediction

4.4(v) Activity diagram for Display Result

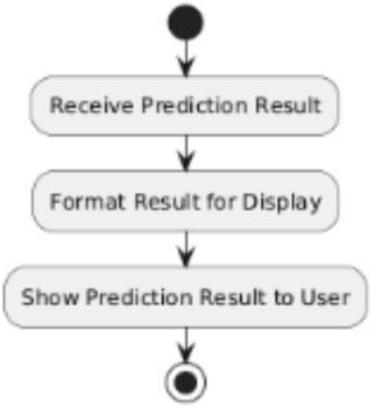

Fig 4.6(v) - Activity diagram for Display Result

5 DATA DICTIONARY

■ DATA DICTIONARY

5.1 Data Dictionary

The different fields or data components that are utilized in AI Powered Multi Disease Diagnostic System, along with their definitions, data types, and any constraints or linkages, are usually listed in a data dictionary. As an illustration, consider this:

Table 5.1 - Patient

Field Name	Data Type	Description
Patient_ID	INT	Unique identifier for the patient (Primary Key).
Name	VARCHAR(100)	Name of the patient.
Age	INT	Age of the patient.
Gender	CHAR(1)	Gender of the patient (e.g., M/F/O).
ContactInfo	VARCHAR(255)	Contact information of the patient

Table 5.2 - Medical_Data

Field Name	Data Type	Description
Data_ID	INT	Unique identifier for the medical data (Primary Key).
Symptoms	TEXT	Description of the patient's symptoms.
TestResults	TEXT	Results of medical tests conducted on the patient.
DiseaseHistory	TEXT	Past medical history or diseases of the patient.
Patient_ID	INT	Foreign key reference Patient.Patient_ID.

Table 5.3 - Prediction_Model

Field Name	Data Type	Description
Model_ID	INT	Unique identifier for the prediction model (Primary Key).
ModelType	VARCHAR(100)	Type of prediction model
Accuracy	DECIMAL	Accuracy percentage of the model.

Table 5.4 - Prediction_Result

Field Name	Data Type	Description
Result_ID	INT	Unique identifier for the prediction result (Primary Key).
PredictionDisease	VARCHAR(100)	Predicted disease based on the medical data.
Patient_id	INT	Foreign key referencing Patient.Patient_ID.
Model_ID	INT	Precision of the model for classification tasks.

This is a simplified example, and in a real-world scenario, the data dictionary would likely include additional details such as constraints, validation rules, and relationships between tables.

It serves as a valuable reference for developers, database administrators, and other stakeholders involved in the project.

6 DISCUSSION AND CONCLUSION

- **RESULT AND DISCUSSION**
- **CONCLUSION**

6.1 Result and Discussion

The machine learning and deep learning methods used in the AI Powered Multi Disease Diagnostic System showed great potential in accurately diagnosing conditions like Parkinson's, heart attacks, diabetes, brain tumours, breast cancer, and tuberculosis. Real-time predictions were made possible by the model's capacity to evaluate input data, including lab results and patient histories, which helped medical practitioners make quicker and better decisions. The technology demonstrated its potential to increase diagnostic efficiency in healthcare by using cutting-edge algorithms like TensorFlow and Keras to produce dependable results across a number of disease categories.

Even with the encouraging outcomes, several things still need improvement. Different datasets may perform differently in the model, and variables like feature selection, class imbalance, and data quality may affect prediction accuracy. The adaptability of the model will be improved by ongoing training with updated data and model refinement. Furthermore, as the system relies on the availability of precise data, ensuring thorough data collection from various healthcare settings will be essential to enhancing its generalisation. With continued development, the system can be a useful instrument for improving accessibility and diagnostic capacities in both urban and rural healthcare environments.

6.2 Conclusion

The AI Powered Multi Disease Diagnostic System that was created with the help of deep learning and machine learning algorithms has the potential to revolutionise medical diagnosis. Health care professionals may make more informed decisions faster due to the system's ability to provide precise, real-time predictions for diseases like Parkinson's, heart attacks, diabetes, brain tumours, breast cancer, and tuberculosis. In the long run, this can help patients and shorten treatment delays, particularly in places where access to specialised medical care is limited.

Constant data gathering, model enhancement, and adaptation to various healthcare settings will be required, however, if the system is to reach its full potential. Although there are issues with data quality, system integration, and user acceptance that need to be addressed, with continued research and improvement, the system can greatly improve the diagnostic process and make healthcare reliable, efficient, and accessible for everyone.

7 REFERENCES

■ REFERENCE

7.1 References

Here are some references and resources that can help in the development of a AI Powered Multi-Disease Diagnostic System.

Websites:

1. https://ieeexplore.ieee.org/document/10060903
2. https://www.researchgate.net/publication/381309960_MULTI_DISEASE_PREDICTION_SYSTEM_USING_MACHINE_LEARNING
3. https://www.irjmets.com/uploadedfiles/paper//issue_1_january_2024/48476/final/fin_irjmets1705419474.pdf
4. https://www.e3s-conferences.org/articles/e3sconf/pdf/2023/67/e3sconf_icmpc2023_01051.pdf

Youtube:

5. https://youtu.be/8Q_QQVQ1HZA?si=R5YSBrtPHZ7CwnfN