Project Report On

PREDICTIX - A MULTI DISEASE PREDICTOR

A dissertation submitted in partial fulfilment of the requirements of Bachelor of Technology Degree in Computer Science and Engineering (Data Science) of the Maulana Abul Kalam Azad University of Technology for the year 2021-2025.

Submitted by

Rhitam Chaudhury (14830521033)

Arunetri Dhar (14830521034)

Utsha Majumder (14830521017)

Soumik Sen (14830521027)

Under the guidance of

Prof. Madhurima Das

Assistant Professor

Dept. of Computer Science and Engineering (Data Science)

Future Institute of Engineering and Management

Department of Computer Science and Engineering (Data Science)

Future Institute of Engineering & Management

(Affiliated to Maulana Abul Kalam Azad University of Technology, West Bengal)

Kolkata – 700150, WB

CERTIFICATE OF APPROVAL

This is to certify that this report of B. Tech. 8th semester project, entitled **PredictiX - A Multi Disease Predictor** is a record of bona-fide work, carried out by Rhitam Chaudhury, Arunetri Dhar, Utsha Majumder, Soumik Sen under my supervision and guidance.

In my opinion, the report in its present form is in partial fulfilment of all the requirements, as specified by the **Future Institute of Engineering & Management** and as per regulations of the **Maulana Abul Kalam Azad University of Technology**. In fact, it has attained the standard necessary for submission. To the best of my knowledge, the results embodied in this report are original in nature and worthy of incorporation in the present version of the report for the B.Tech. program in Computer Science and Engineering (Data Science) in the year 2021-2025.

It is understood that by this approval the undersigned does not necessarily endorse or approve any statement made, opinion expressed, or conclusion drawn therein, but approve this thesis for the purpose for which it is submitted.

Guide / Supervisor

_____

Prof. Madhurima Das
Department of Computer Science and Engineering (Data Science)
Future Institute of Engineering & Management

_____

Examiner(s)

_____

Head of the Department
Computer Science and Engineering(Data Science)
Future Institute of Engineering and Management

ACKNOWLEDGEMENT

We sincerely extend our heartfelt gratitude to Prof. Madhurima Das, our mentor, for her invaluable guidance, insightful suggestions, and unwavering support throughout this project. Her expertise and encouragement have been instrumental in shaping our work and helping us overcome challenges. We are also deeply grateful to our institution and the faculty of the Computer Science and Engineering (Data Science) department for providing us with the resources, knowledge, and opportunities that enabled us to pursue this project successfully. Their mentorship and support have been truly inspiring.

Additionally, we would like to express our appreciation to our peers and friends for their thoughtful feedback and collaborative spirit, which played a significant role in refining our ideas. We are equally thankful to our families for their constant encouragement, patience, and motivation, which fueled our determination to give our best. This project is a culmination of the collective support and inspiration we received from all those around us, and we are sincerely grateful to each one of them.

Mr. Rhitam Chaudhury: _____
University Roll No – 14800121033
Registration No - 211480130510020(2021-22)

Ms. Arunetri Dhar: _____
University Roll No – 14830521034
Registration No - 211480130510018(2021-22)

Ms. Utsha Majumder: _____
University Roll No – 14830521017
Registration No - 211480130510058(2021-22)

Mr. Soumik Sen: _____
University Roll No – 14830521027
Registration No - 211480130510050(2021-22)

PROJECT ABSTRACT

Predictix - A Multi Disease Predictor is an online health prediction system designed to help users assess the early detection of four serious medical conditions: **Breast Cancer, Lung Cancer, Diabetes**, and **Heart Disease**. The system allows individuals to either enter their health information manually or upload a structured medical report. Based on the input, it provides an immediate risk of suffering from the mentioned disease.

The purpose of Predictix is to support early awareness and encourage timely medical consultation. It simplifies complex medical evaluations into easy-to-understand results, making it accessible for both healthcare professionals and the general public. The system is designed to be user-friendly, fast, and informative, helping bridge the gap between initial health concerns and professional diagnosis.

By focusing on accuracy, ease of use, and accessibility, Predictix aims to promote preventive healthcare and empower users to take control of their health in a convenient and reliable way.

CONTENTS

1. INTRODUCTION 1
    1.1 Objective 2
    1.2 Scope 3-5
    1.3 Feasibility Study 6
        1.3.1 Technical Feasibility 6
        1.3.2 Operational Feasibility 6-7
        1.3.3 Economic Feasibility 7

2. SOFTWARE REQUIREMENT SPECIFICATION (SRS) 8-11

3. SOFTWARE DEVELOPMENT PROCESS MODEL ADOPTED 12-13

4. OVERVIEW 14-17
    4.1 System Overview 14
        4.1.1 Limitation of Existing System 14
    4.2 Proposed System 15
        4.2.1 Objectives of the Proposed System 15
        4.2.2 Users of the Proposed System 15
    4.3 System Design and Implementation 16
        4.3.1 Breast Cancer Prediction 16
        4.3.2 Lung Cancer Prediction 16
        4.3.3 Diabetes Prediction 17
        4.3.4 Heart Disease Prediction 17

5. ASSUMPTION AND DEPENDENCIES 18-19

6. TECHNOLOGIES 20-22
    6.1 Tools used in Development 20
    6.2 Development Environment 20
    6.3 Software Interface 21
    6.4 Hardware Used 22

7. DESIGN 23-24
    7.1 Workflow Diagram 23
    7.2 Data Flow Diagram 24

8. DATA DICTIONARY 25-28

9. SNAPSHOTS 29-41

10. CONCLUSION 42

11. FUTURE SCOPE 43

12. REFERENCES 44

1. INTRODUCTION

With the rising number of people affected by chronic and life-threatening diseases, early diagnosis has become an important step in improving treatment outcomes and reducing health risks. Many health conditions, such as breast cancer, lung cancer, diabetes, and heart disease, can be better managed or even prevented when they are identified at an early stage. However, people often miss early signs due to lack of awareness or limited access to regular medical checkups. To support early detection in a simple and accessible way, **PredictiX** was developed as a health prediction system.

Predictix focuses on four major health conditions that are common and have a significant impact on public health. The system helps users check for possible signs of breast cancer, lung cancer, diabetes, or heart disease based on the information they provide. It is designed to give quick results and serve as an initial step that encourages users to follow up with medical professionals if any risk is found.

The system provides two input options for users. One option is to manually fill in their health-related details, such as basic information and symptoms. The other option is to upload a structured medical report that contains relevant data. Once the information is submitted, the system gives a simple output indicating whether there is a possible risk of the disease. This helps users get a basic understanding of their health status without needing advanced medical knowledge.

Predictix is built to be user-friendly and easy to navigate. It does not give a final diagnosis but is meant to assist with early detection and raise awareness. It encourages people to take the next step by consulting a doctor or going for further medical tests if needed. The goal is to help users become more informed about their health and take timely action when necessary.

In summary, Predictix is a supportive health prediction tool that focuses on four key diseases. It allows users to input their information, get a quick result, and use that result as a starting point for further medical consultation. The system is meant to complement medical checkups and support early awareness, making health monitoring more accessible.

1.1 Objectives:

The primary objective of Predictix is to develop a comprehensive and intelligent healthcare prediction system capable of early detection and diagnosis of critical medical conditions, including breast cancer, lung cancer, diabetes, and heart disease. By leveraging machine learning and deep learning algorithms, the project aims to provide accurate, fast, and accessible disease prediction to support both patients and healthcare professionals in making informed decisions.

Predictix is designed to:

- Help identify signs of serious health conditions at an early stage.
- Make health checks more accessible through a simple and easy-to-use system.
- Allow users to enter their health information or upload reports for quick analysis.
- Provide fast and clear results without needing heavy resources.
- Support early health awareness in areas with limited medical facilities.

By achieving these objectives, Predictix strives to bridge the gap between cutting-edge AI technologies and practical, impactful healthcare applications, ultimately promoting preventive healthcare and early diagnosis.

1.2 Scope

The Predictix project aims to provide a powerful, intelligent, and user-friendly platform that leverages machine learning and deep learning to predict four major health conditions: breast cancer, lung cancer, diabetes, and heart disease. Its scope includes data preprocessing, model development, web application integration, and user accessibility, while also considering current limitations and future enhancements.

1. Disease Coverage

Predictix supports prediction of:

- **Breast Cancer** using CNNs trained on histopathological image data.
- **Lung Cancer** using ResNet trained on CT scan images.
- **Diabetes** using SVM applied to structured patient data.
- **Heart Disease** using a tuned Decision Tree model with clinical parameters.

2. Input Mechanisms

- **Manual Entry**: Users can input numerical values for features such as glucose level, age, cholesterol, etc.
- **Report Upload**: Users can upload structured health reports (CSV/txt) following a specific format.
- **Planned Feature**: OCR integration to automatically read values from scanned or printed medical reports in PDF/image format.

3. Target Users

Predictix is designed with a wide range of potential users in mind:

- **General Public / Patients**:
    - Individuals seeking early awareness or risk assessment for chronic diseases based on their health metrics.
    - Especially helpful in areas with limited access to advanced diagnostic facilities.

- **Healthcare Professionals**:
    - Doctors and medical practitioners can use it as a **pre-screening tool** to support decision-making and prioritize diagnostic tests.
- **Medical Students and Researchers**:
    - As a learning and experimentation platform to understand how AI/ML can be applied in healthcare.
- **Hospitals and Clinics**:
    - Integration into existing health management systems for streamlined prediction and record keeping.

The interface and predictive reports are designed to be easy to understand, enabling non-technical users to make sense of their results while still offering valuable insights to professionals.

4. System Architecture and Technologies

- **Frontend**: React.js
- **Backend**: Node.js with Express
- **Database**: MongoDB for user and system data
- **ML Models**: Developed using Python, saved as .pkl files, and integrated using Node.js threads
- **Image-based models (CNN, ResNet)** require GPU but are currently not deployed due to resource constraints

5. Performance & Accuracy

- Breast Cancer: ~89%
- Lung Cancer: ~83%
- Diabetes: ~85%
- Heart Disease: ~87%

All models are trained from scratch and optimized through feature selection and hyperparameter tuning.

6. Deployment & Accessibility

- Web-based system for easy accessibility via browser.
- No installation required.
- User-friendly forms and interactive interfaces.
- Currently hosted locally with potential for cloud deployment in the future.

7. Limitations

- Deployment of image-based models is not feasible in real-time due to GPU limitations.
- Fixed report format needed for file uploads.
- No multilingual support currently.
- Not a replacement for actual medical diagnosis—meant to be a decision-support tool.

8. Future Scope

- Integration of **OCR** for unstructured reports.
- Adding more disease predictors.
- Adding user accounts and prediction history tracking.

1.3 Feasibility Study

A feasibility study is an essential part of any software development project to determine whether the proposed system is practical, viable, and beneficial. For the Predictix system, a detailed analysis has been conducted across three key dimensions: technical, operational, and economic feasibility.

1.3.1 Technical Feasibility

Technical feasibility evaluates whether the current technology, tools, and infrastructure are sufficient to build and run the system effectively.

Assessment for Predictix:

- **Technology Stack Availability:** Predictix uses the MERN stack (MongoDB, Express.js, React.js, Node.js), which is widely supported, open-source, and suitable for scalable web applications.
- **Machine Learning Frameworks:** Python-based libraries like scikit-learn, TensorFlow, and Keras are used for model development and are well-documented and compatible with the chosen tech stack.
- **Model Integration:** Machine learning models are saved as .pkl and .h5 files and seamlessly integrated with the Node.js backend using worker threads, demonstrating technical viability.
- **Deployment Limitations:** The deep learning models (CNN and ResNet) for image-based predictions are resource-intensive and currently not deployed due to lack of GPU-enabled servers. However, this is a scalable limitation—cloud services can be used in the future to resolve this.

1.3.2 Operational Feasibility

Operational feasibility assesses how well the system fits into the intended environment and whether users will accept and adopt it.

Assessment for Predictix:

- **User Interface Design:** The frontend is intuitive and user-friendly, designed for both technical and non-technical users, including patients and healthcare providers.

- **Functionality:** Users can either manually enter medical data or upload reports for disease prediction. This flexibility increases user adoption.
- **Accessibility:** As a web-based platform, Predictix requires no installations and can be accessed from any device with a browser, increasing usability across different demographics.
- **Maintainability:** The modular architecture of the system ensures that new diseases, models, or input methods can be added with minimal disruption.

1.3.3 Economical Feasibility

Economic feasibility involves evaluating the cost-benefit analysis of developing and deploying the system.

Assessment for Predictix:

- **Development Costs**: The system was developed using open-source tools and libraries, eliminating licensing costs and reducing development expenses.
- **Infrastructure**: Development and testing were performed on local machines without paid GPU servers. Image-based model deployment was deferred due to high cost but remains a future investment.
- **Maintenance Costs**: The system's architecture supports easy updates and additions, minimizing long-term maintenance costs.
- **Value Addition**: Predictix provides significant value by offering early diagnosis tools that can potentially reduce diagnostic costs in the healthcare sector.

2. SOFTWARE REQUIREMENTS SPECIFICATION (SRS)

Project Title: Predictix – A multi-disease prediction system.

2.1 Introduction

2.1.1 Purpose

The purpose of this document is to define the software requirements for **Predictix**, a web-based health prediction system that uses Machine Learning (ML) and Deep Learning (DL) to detect diseases such as **breast cancer, lung cancer, diabetes, and heart disease**. It serves as a guideline for developers, testers, and stakeholders to understand the system functionalities, design expectations, and performance benchmarks.

2.1.2 Scope

Predictix enables users to input health data manually or upload structured medical reports for disease prediction. It incorporates trained ML/DL models and provides an accessible web interface for predictions. Built using the MERN stack, it emphasizes modularity, accuracy, and user-friendliness. The system is designed for future extensibility, including OCR integration and cloud deployment.

2.1.3 Definitions, Acronyms, and Abbreviations

- **ML**: Machine Learning
- **DL**: Deep Learning
- **CNN**: Convolutional Neural Network
- **OCR**: Optical Character Recognition
- **SVM**: Support Vector Machine
- **UI**: User Interface
- **MERN**: MongoDB, Express.js, React.js, Node.js

2.2 Overall Description

2.2.1 Product Perspective

Predictix is a standalone system but can be integrated into larger health IT infrastructures. It acts as a diagnostic aid rather than a replacement for professional medical judgment.

2.2.2 Product Functions

- Allow users to enter health-related values manually.
- Accept structured medical reports for prediction.
- Load and run pre-trained ML models in the backend.
- Display disease prediction results and risk scores.
- Provide clean, user-friendly web-based interfaces.

2.2.3 User Classes and Characteristics

- **General Users/Patients**: Use the tool for personal risk assessment.
- **Doctors/Medical Professionals**: Use predictions to support early diagnosis.
- **Students/Researchers**: Use the platform for educational and experimentation purposes.

2.2.4 Operating Environment

- **Frontend**: React.js (Browser-based)
- **Backend**: Node.js with Express
- **Database**: MongoDB
- **ML Models**: Trained in Python, integrated into Node.js using worker threads
- **Hardware**: Works on standard desktops/laptops; GPU required for DL model deployment

2.2.5 Design and Implementation Constraints

- Structured report format required for upload.
- Models must be serialized (pickle) for Node.js compatibility.

2.2.6 Assumptions and Dependencies

- Users will enter accurate medical data.
- Uploaded reports will follow the required structure.
- The system assumes a stable internet connection for access.

2.3 Specific Requirements

2.3.1 Functional Requirements

ID	Requirement Description
FR1	The system shall allow users to input individual health values.
FR2	The system shall support file upload for structured reports.
FR3	The backend shall load ML and DL models for predictions.
FR4	The system shall display prediction results to the user with interpretation.
FR5	The system shall validate inputs before processing.
FR6	The system shall store user data and logs in MongoDB.

2.3.2 Non-Functional Requirements

ID	Requirement Description
FR1	The system shall provide responses within 10 seconds for input-based predictions.
FR2	The system shall maintain >90% uptime.
FR3	The user interface shall be intuitive and mobile-responsive.
FR4	The system shall support scalability for future model integrations.
FR5	Security measures shall be taken to prevent unauthorized data access.

2.4. External Interface Requirements

2.4.1 User Interfaces

- Clean UI with separate forms for each disease prediction.
- Upload section with file format guidelines.
- Result display with confidence scores and risk indicators.

2.4.2 Hardware Interfaces

- Basic system requirements for development: 8GB RAM, 2.5 GHz processor.
- GPU needed for future DL model deployment.

2.4.3 Software Interfaces

- **React.js** for frontend
- **Node.js + Express.js** for backend APIs
- **MongoDB** for data storage
- **Python (scikit-learn, TensorFlow, Keras)** for model development

2.4.4 Communication Interfaces

- HTTP/HTTPS for frontend-backend communication via REST APIs.
- JSON for request and response formats.

2.5. Appendices

- **Datasets and Model Accuracies:**
  - **Breast Cancer** (BreakHis dataset, CNN): ~89%
  - **Lung Cancer** (IQ-OTH/NCCD dataset, ResNet): ~83%
  - **Diabetes** (Pima Indian dataset, SVM): ~85%
  - **Heart Disease** (Kaggle dataset by johnsmith88, Decision Tree): ~87%

3. SOFTWARE DEVELOPMENT MODEL PROCESS ADOPTED

Model Used: Iterative Model.

Justification: The Iterative Model was adopted for the development of Predictix to allow progressive development, continuous testing, and incremental improvement. Given the nature of the project—developing machine learning models, integrating them into a web application, and gathering feedback from testing—the Iterative model proved to be a suitable approach.

Phases Followed in Predictix Using the Iterative Model:

i) Requirement Analysis

- Identified key diseases (breast cancer, lung cancer, diabetes, heart disease) to be predicted.
- Defined input types: manual data entry and report upload.
- Finalized performance goals for each model and tech stack (MERN, Python for ML).

ii) Design

- Created a modular system design separating frontend, backend, and ML logic.
- Designed intuitive UI wireframes for user data input and result visualization.
- Planned dataset preprocessing and model training pipelines.

iii) Implementation (First Iteration)

- Implemented the frontend in React.js with forms for manual input.
- Developed and saved initial ML models in Python as .pkl and .h5 files.
- Integrated basic backend endpoints using Node.js.

iv) Testing and Feedback

- Unit-tested ML model predictions for each disease.
- Conducted integration testing between frontend and backend.
- Received peer/faculty feedback and identified areas for improvement:
    - UI enhancements.
    - Better result interpretation.

v) Refinement (Second Iteration)

- Added report upload feature with file parsing logic.
- Enhanced UI and form validations.
- Optimized model performance and handling of edge cases.

vi) Future Iteration Plan

- Integration of OCR for flexible report upload.
- Deployment of deep learning models using cloud GPU resources.

Advantages of Using the Iterative Model for Predictix

- Early working prototypes helped get feedback quickly.
- Risk reduction by isolating ML integration and web components.
- Flexibility to improve and extend the system based on testing outcomes.
- Encouraged continuous learning, especially during model selection and optimization.

4. OVERVIEW

4.1 System Overview

Predictix is a web-based health prediction system that uses Machine Learning (ML) and Deep Learning (DL) models to predict the risk of four major health conditions: breast cancer, lung cancer, diabetes, and heart disease. The system accepts user input either manually or through structured health report uploads, processes the data using pre-trained models, and displays the predicted risk outcome. Built using the MERN stack, the application is designed to be modular, scalable, and accessible to both medical professionals and laypersons.

4.1.1 Limitations of the Existing System

Existing health diagnostic systems face several limitations:

- **Time-Consuming Diagnostics**: Manual analysis of medical images and reports often delays early detection and timely treatment.
- **Lack of Accessibility**: Diagnostic tools are usually expensive and limited to hospital infrastructure, making them inaccessible to people in rural or underdeveloped areas.
- **No Unified Platform**: Most health prediction tools focus on only one disease; there's a lack of a single integrated system for multiple health predictions.
- **Non-Intelligent Systems**: Many existing tools do not utilize AI/ML and thus lack predictive intelligence and adaptability.
- **Dependency on Expert Opinion**: In the absence of automated support tools, decisions depend entirely on specialist availability and interpretation.

4.2 Proposed System

The proposed system, Predictix, addresses the above limitations by providing a unified, intelligent platform for disease prediction that is fast, lightweight, and accessible from any device with an internet connection.

4.2.1 Objectives of the Proposed System

- To build an intelligent web-based system for early prediction of breast cancer, lung cancer, diabetes, and heart disease.
- To allow manual entry and report upload for user flexibility.
- To integrate Machine Learning and Deep Learning models that are trained for high accuracy and reliability.
- To design a system that is user-friendly, and scalable.
- To assist non-specialist users in understanding their risk factors and support healthcare professionals in screening and diagnosis.

4.2.2 Users of the Proposed System

The system is designed for a variety of users:

- **General Public / Patients**
  To self-assess health risks using their personal medical data.
- **Doctors and Healthcare Professionals**
  To use as a decision-support system during patient diagnosis or screening.
- **Medical Students and Researchers**
  For educational purposes, experimentation with ML models, and understanding medical data prediction techniques.
- **Hospitals/Clinics** (for future integration)
  As a part of digital health record systems or diagnostic aids in outpatient departments.

4.3 System Design and Implementation

4.3.1 Breast Cancer Prediction

- **Model Used:** Convolutional Neural Network (CNN)
- **Dataset:** 547 benign and 1146 malignant histology images
- **Why CNN?**
  CNNs are ideal for image classification problems as they automatically learn spatial hierarchies and features. Unlike traditional ML models, CNNs reduce the need for manual feature extraction.

  Spanhol et al. (2016) [1] proposed a benchmark dataset of breast cancer histopathological images and evaluated deep convolutional neural networks for automated breast cancer classification.

- **Architecture Summary:**
  - Input layer → Conv2D → MaxPooling → Flatten → Dense → Output (Sigmoid)
  - Achieved ~89% accuracy
- **Dataset:** BreakHis (Breast Cancer Histopathological Image Classification) [2]

4.3.2 Lung Cancer Prediction

- **Model Used:** ResNet
- **Dataset:** 720 cancerous and 240 non-cancerous CT scan images
- **Why ResNet?**
  ResNet (Residual Network) overcomes the vanishing gradient problem and allows very deep networks to be trained effectively. It was chosen after comparing CNN, MobileNet, and XceptionNet.

  Vinayak et al. [3] highlighted the effectiveness of ResNet-50 in early-stage lung cancer detection through efficient feature learning and image representation.

- **Performance:** ~83% accuracy with improved stability and generalization.
- **Dataset:** The IQ-OTH/NCCD lung cancer dataset [4]

4.3.3 Diabetes Prediction

- **Model Used:** Support Vector Machine (SVM)
- **Dataset:** 769×9 clinical data
- **Why SVM?**
  SVM creates optimal decision boundaries and is effective even when the data is not linearly separable. It performed better than Logistic Regression and kNN in handling overlapping classes.

  Aviral Srivastava et al. [5] demonstrated the potential of SVM in achieving high accuracy for early diabetes detection through a web-based ML system.
- **Accuracy:** ~85%
- **Dataset:** Pima Indian Diabetes Dataset [6]

4.3.4 Heart Disease Prediction

- **Model Used:** Tuned Decision Tree
- **Dataset:** 1026×14
- **Why Decision Tree?**
  Decision Trees are interpretable and can capture non-linear relationships. After testing Logistic Regression and Random Forest, the tuned tree gave the best performance (~87% accuracy) with low complexity.

  Vijaya Saraswathi et al. [7] showed that decision trees were effective in classifying heart disease cases and performed better than several other classification methods in their study.
- **Dataset:** Heart Disease Dataset (johnsmith88, Kaggle) [8]

5. ASSUMPTIONS AND DEPENDENCIES

Assumptions:

- The users will provide **accurate and valid medical data** when entering values manually or uploading reports.
- Uploaded medical reports will follow the **specified structured format** to ensure proper parsing and data extraction.
- Users have **basic computer literacy** to navigate the web interface and upload documents.
- The system assumes availability of a **stable internet connection** for uninterrupted access to the web platform.
- Machine learning models trained on specific datasets are **generalizable** to the user's input data, assuming similarity in data distribution.
- The system assumes **no malicious use or data tampering**, and users will use the platform responsibly.
- User feedback and testing will be used iteratively to improve and fine-tune the models and interface.

Dependencies:

- **Technology Stack Dependencies:**
    - Frontend: React.js
    - Backend: Node.js with Express.js
    - Database: MongoDB
    - Machine Learning: Python libraries such as scikit-learn, TensorFlow, Keras for model building and serialization.
- **Data Dependencies:**
    - Availability of high-quality and labeled datasets for training models (e.g., histopathology images for breast cancer, CT scans for lung cancer, clinical data for diabetes and heart disease).
    - Dependence on external repositories for diabetes and heart disease datasets.

- **Hardware Dependencies:**
    - Standard desktop/laptop specifications suffice for most operations.
    - GPU hardware needed for training and future deployment of deep learning models efficiently.
- **Software Dependencies:**
    - Python environment with necessary ML libraries.
    - Node.js worker threads to integrate Python ML models into backend.
- **User Dependencies:**
    - Users are dependent on understanding the required report format for uploading files until OCR is integrated.
    - Users depend on accurate data input for meaningful prediction results.
- **Infrastructure Dependencies:**
    - Web hosting server with necessary capacity and uptime to ensure accessibility.
    - Security infrastructure for protecting user data and ensuring privacy.

6. TECHNOLOGIES

6.1 Tools Used in Development

- **Programming Languages:**
    - **JavaScript:** For frontend (React.js) and backend (Node.js) development.
    - **Python:** For developing and training Machine Learning and Deep Learning models.
- **Frameworks and Libraries:**
    - **React.js:** Frontend UI development framework.
    - **Node.js and Express.js:** Backend server and API development.
    - **scikit-learn:** Python library for traditional ML models like SVM, Decision Trees.
    - **TensorFlow and Keras:** For developing Deep Learning models such as CNN and ResNet.
    - **MongoDB:** NoSQL database for storing user data and logs.
- **Other Tools:**
    - **Pickle and H5:** For model serialization and deserialization in Python to integrate ML models with Node.js.
    - **Postman:** API testing tool during backend development.
    - **Git/GitHub:** Version control and source code management.

6.2 Development Environment

- **Operating System:** Windows 10 / Ubuntu Linux (Development done on both environments).
- **Code Editor/IDE:** Visual Studio Code with relevant extensions for JavaScript and Python.
- **Python Environment:** Anaconda distribution or Python 3.x with pip-managed packages for ML libraries.

- **Browser:** Google Chrome (latest version) for frontend testing and debugging.
- **Local Server:** Node.js server running locally during development for API testing.
- **MongoDB Atlas or Local MongoDB:** Used for databases during development and testing.

6.3 Software Interface

- **Frontend Interface:**
    - Web-based user interface built with React.js that communicates with backend APIs via HTTP/HTTPS.
    - Provides forms for manual data entry and file upload sections.
    - Displays prediction results and related information.
- **Backend Interface:**
    - RESTful API endpoints developed using Express.js to handle requests from frontend.
    - Integration of Python ML models via worker threads for prediction processing.
- **Database Interface:**
    - MongoDB interface used to store user inputs, prediction logs, and system metadata.
    - Accessed through Mongoose ODM in the Node.js backend.
- **Model Interface:**
    - Python ML models are saved as .pkl and .h5 files and called via Node.js threads using child processes or worker threads.

6.4 Hardware Used

- **Development Machines:**
  Standard laptops/desktops with minimum specifications:
    - CPU: Intel Core i5 or equivalent
    - RAM: 8GB or more
    - Storage: SSD for faster read/write operations
- **GPU Requirements:**
    - Training of deep learning models was done on systems equipped with GPU.
    - Deployment currently limited due to GPU resource constraints; future deployment will require access to GPU-enabled servers.
- **Testing Devices:**
    - Web application tested across multiple devices including desktops, laptops, and mobile devices for responsiveness.

7. DESIGN

7.1 Workflow Diagram

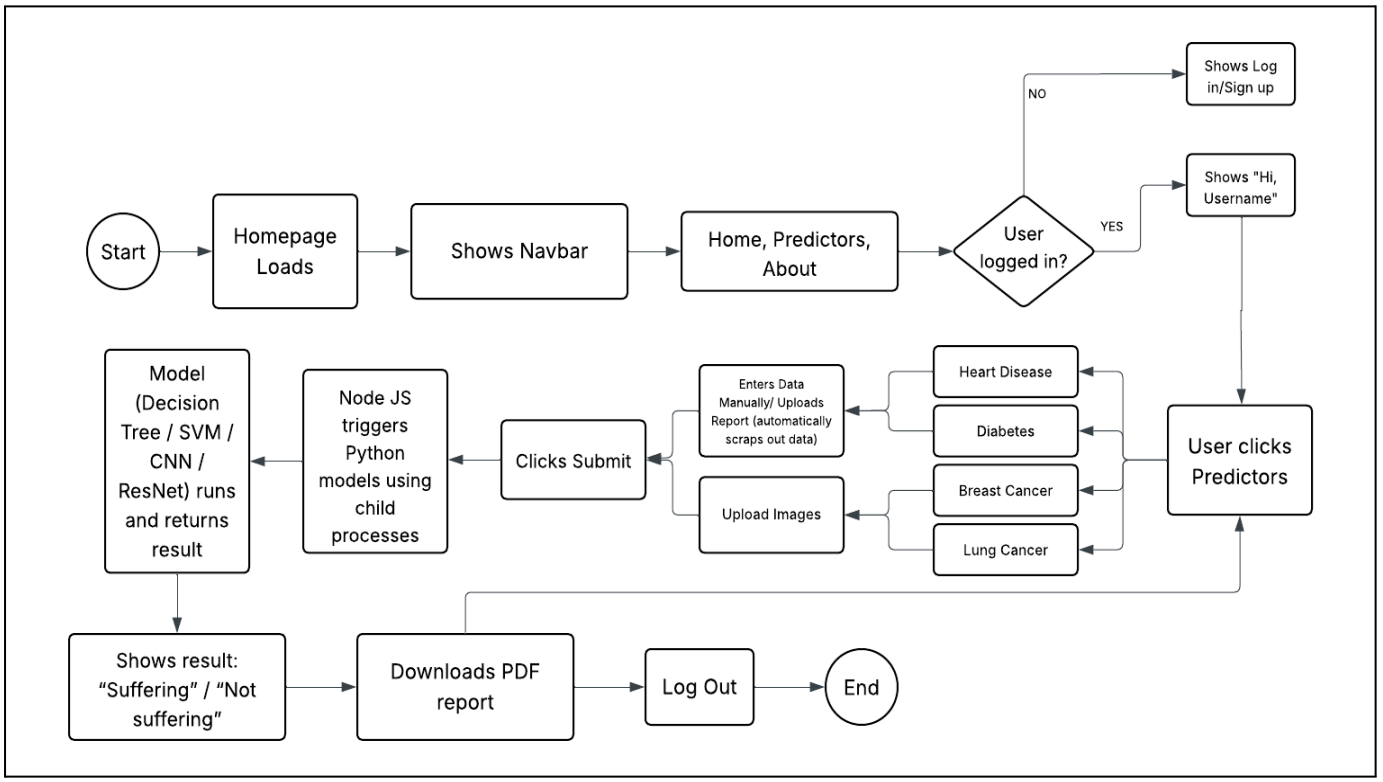

Fig 1: Workflow Diagram

7.2 Data Flow Diagram (DFD)

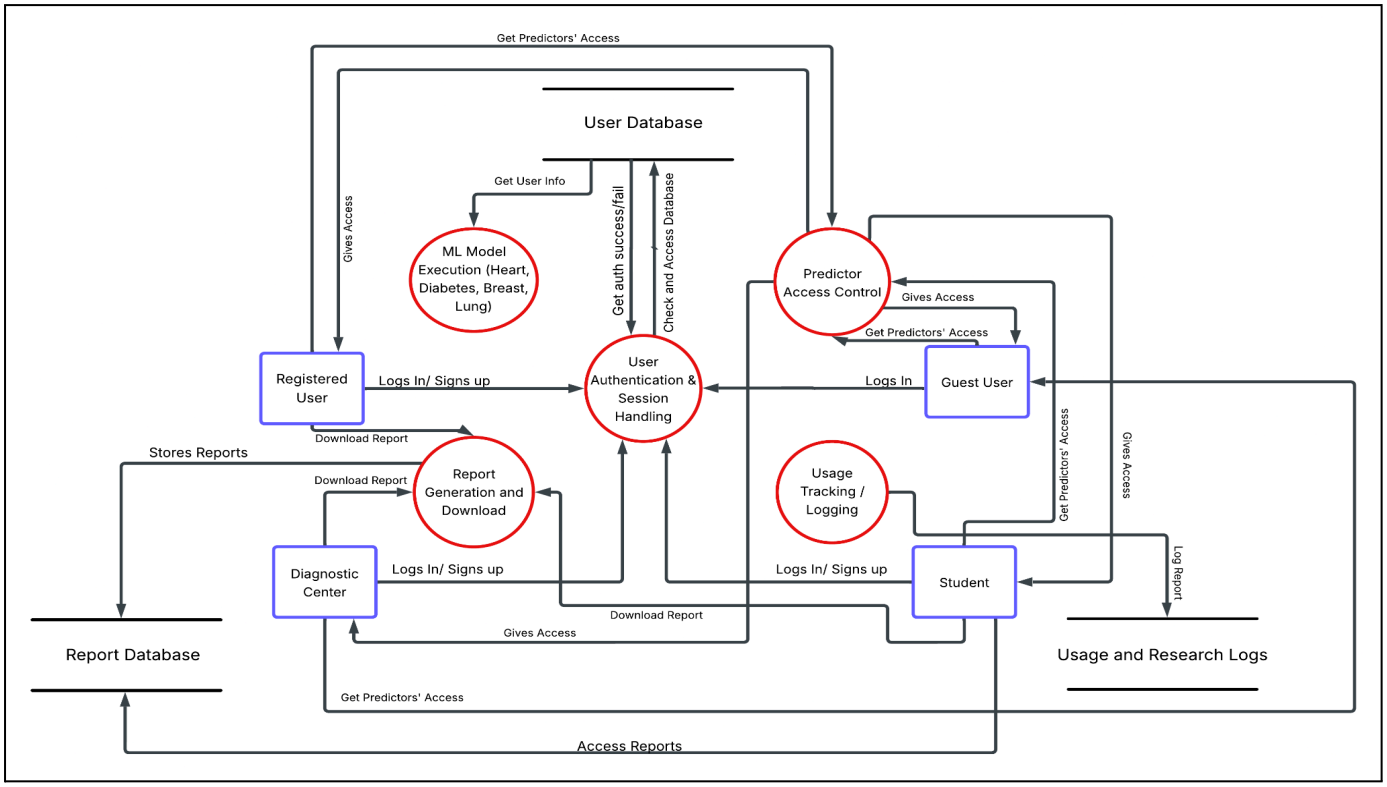

Fig 2: Data Flow Diagram

8. DATA DICTIONARY

This section explains the input features used in Predictix for disease prediction.

For structured datasets like diabetes and heart disease, a detailed list of attributes, their types, and valid ranges is provided.

For image-based models (breast and lung cancer), traditional data dictionaries don't apply. Instead, we describe the image formats, preprocessing steps, and labeling used for model training.

8.1 Heart Disease Prediction

Field Name	Description	Data Type	Allowed Values / Ranges	Remarks
age	Age of patient	Integer	29 - 77	Older individuals are at higher risk
sex	Biological sex	Integer	0 = Female, 1 = Male	Binary encoding
cp	Chest Pain type	Integer	0 = Typical Angina, 1 = Atypical Angina, 2 = Non-Anginal pain, 3 = Asymptomatic	Indicates type of chest discomfort
trestbps	Resting Blood Pressure (mm Hg)	Integer	94 - 200	Measured on admission
chol	Serum cholesterol (mg/dL)	Integer	126 - 564	High levels indicate risk
fbs	Fasting Blood Sugar > 120 mg/dL	Integer	0 = False, 1 = True	Binary variable
restecg	Resting Electrocardiographic results	Integer	0 = Normal, 1 = ST-T abnormality, 2 = Left ventricular hypertrophy	Categorical ECG interpretation

thalach	Maximum heart rate achieved	Integer	71 - 202	Lower values may indicate issues
exang	Exercise-induced Angina	Integer	0 = No, 1 = Yes	Angina triggered by exercise
oldpeak	ST depression induced by exercise related to rest	Float	0.0 - 6.2	Indicates heart stress under exertion
slope	Slope of the peak exercise ST segment	Integer	0 = Upsloping, 1 = Flat, 2 = Downsloping	ECG parameter related to ischemia
ca	Number of major vessels (0-3) coloured by fluoroscopy	Integer	0 - 4	Greater number indicates worse condition
thal	Thalassemia level	Integer	1 = Fixed Defect, 2 = Normal, 3 = Reversible Defect	Categorical variable describing thalassemia condition
target	Final prediction table	Integer	0 = No heart disease, 1 = Heart disease	Model output — binary classification

8.2 Diabetes Prediction

Field Name	Description	Data Type	Allowed Values / Ranges	Remarks
Pregnancies	Number of times the patient has been pregnant	Integer	0 - 17	Used to assess gestational diabetes risk; applicable to females
Glucose	Plasma glucose concentration after 2 hours in an oral glucose test	Integer	0 - 200+	Key factor for diabetes diagnosis
BloodPressure	Diastolic blood pressure (mm Hg)	Integer	0 - 122	A low value may indicate hypotension
SkinThickness	Triceps skin fold thickness (mm)	Integer	0 - 99	Used to estimate body fat
Insulin	2-Hour serum insulin level (mu U/ml)	Integer	0 - 846	Helps determine insulin resistance
BMI	Body Mass Index (weight in kg / height in m^2)	Float	0.0 - 67.1	BMI above 30 indicates obesity, a diabetes risk factor
DiabetesPedigree Function	Genetic likelihood of diabetes based on family history	Float	0.078 - 2.42	Higher values suggest higher hereditary risk
Age	Age of the patient (in years)	Integer	21 - 81	Risk increases with age
Outcome	Final diagnosis result (prediction label)	Integer	0 = Non-diabetic, 1 = Diabetic	Target variable for supervised learning

8.3 Breast Cancer Prediction

Parameter	Details
Input Type	Colored histopathological images (microscopic tissue scan)
Image Format	JPG/PNG
Image Dimensions	Resized to 224 × 224 pixels
Channels	RGB (3 channels)
Preprocessing	Normalization, resizing, augmentation (flip, rotate)
Model Used	CNN (Convolutional Neural Network)
Labels	0 = Benign, 1 = Malignant
Dataset Reference	Breast Cancer Histopathological Image Classification Dataset

8.4 Lung Cancer Prediction

Parameter	Details
Input Type	Chest CT Scan images
Image Format	JPG/PNG
Image Dimensions	Resized to 224 × 224 pixels
Channels	Grayscale (1 channel)
Preprocessing	Histogram equalization, resizing, normalization
Model Used	ResNet (Residual Neural Network)
Labels	0 = Non-Cancerous, 1 = Cancerous
Dataset Reference	Breast Cancer Histopathological Image Classification Dataset

9. SNAPSHOTS

9.1 Home page

Home Page: Introduces users to Predictix, outlining its purpose of providing intelligent disease prediction through user-friendly tools focused on heart, diabetes, breast cancer, and lung cancer detection.

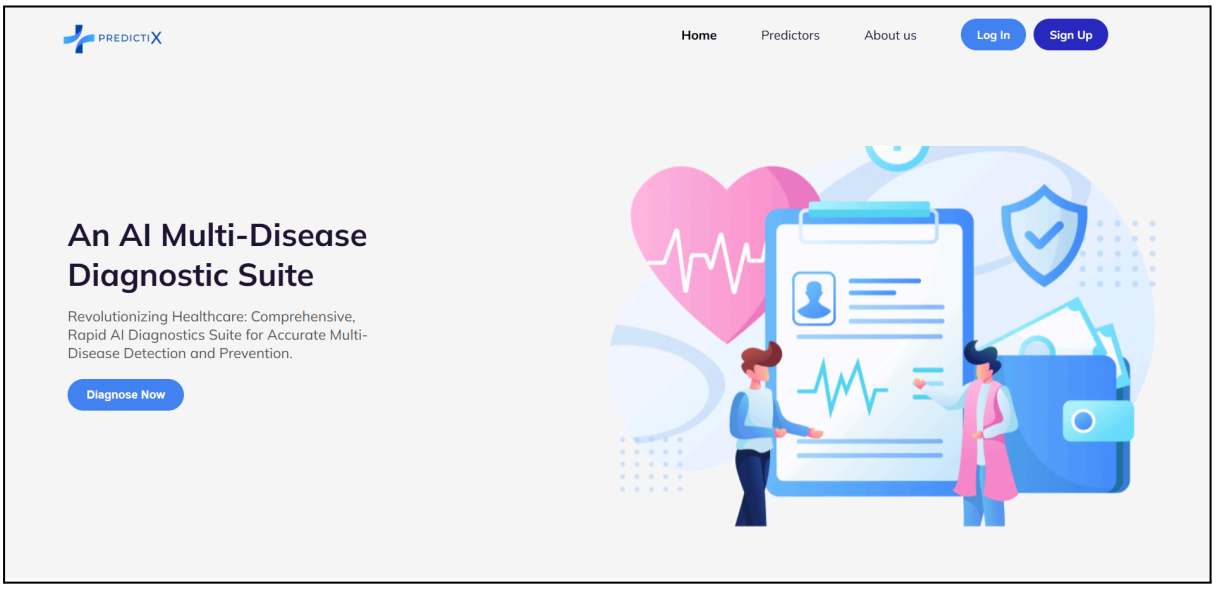

Fig 3: Predictix Home Page

9.2 Sign Up page and Login page

Sign Up Page: Allows new users to register on Predictix and start predicting diseases based on health inputs.

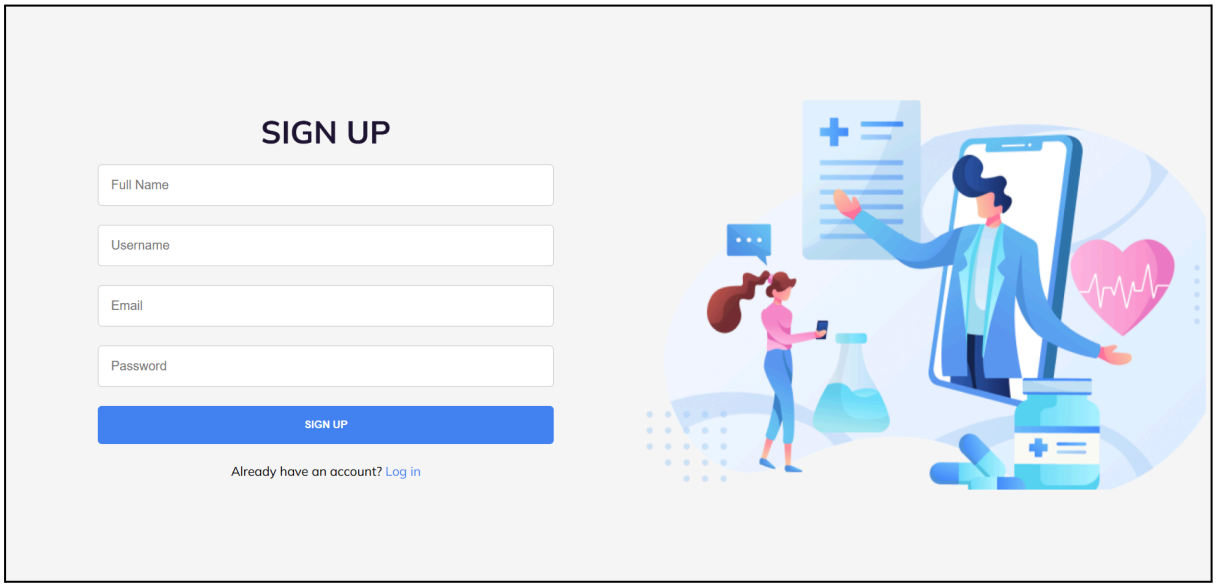

Fig 4: Sign Up Page

Login Page: Lets existing users securely access their personalized Predictix dashboard using their credentials.

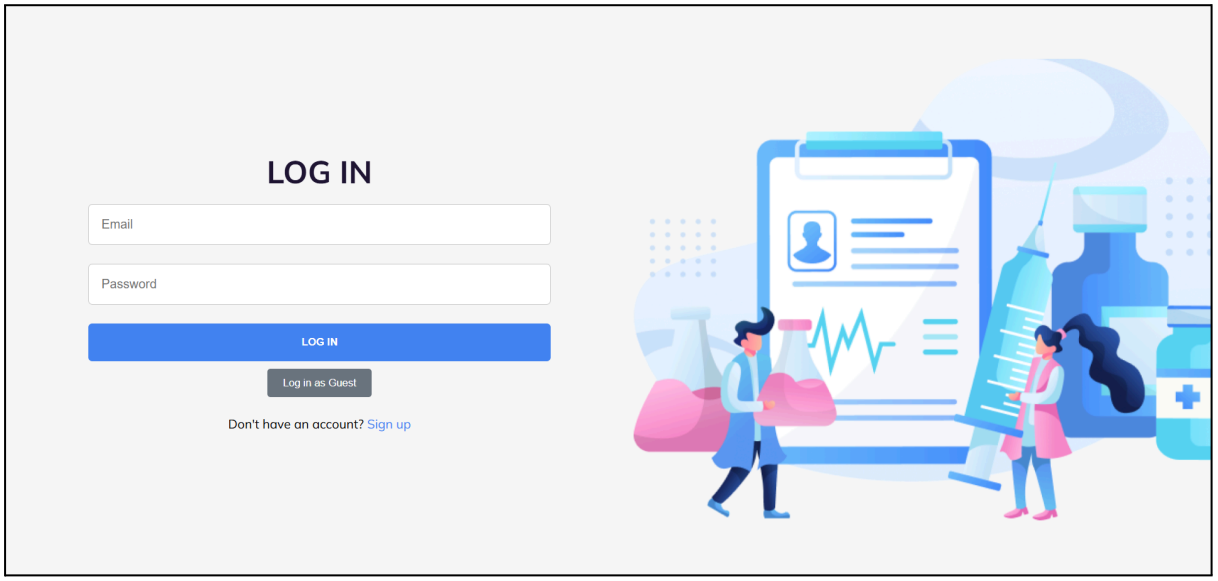

Fig 5: Login Page

9.3 Disease predictors

The disease predictors in Predictix use machine learning algorithms to analyze user-provided information and images, helping identify the chances of heart disease, diabetes, breast cancer, and lung cancer with accuracy.

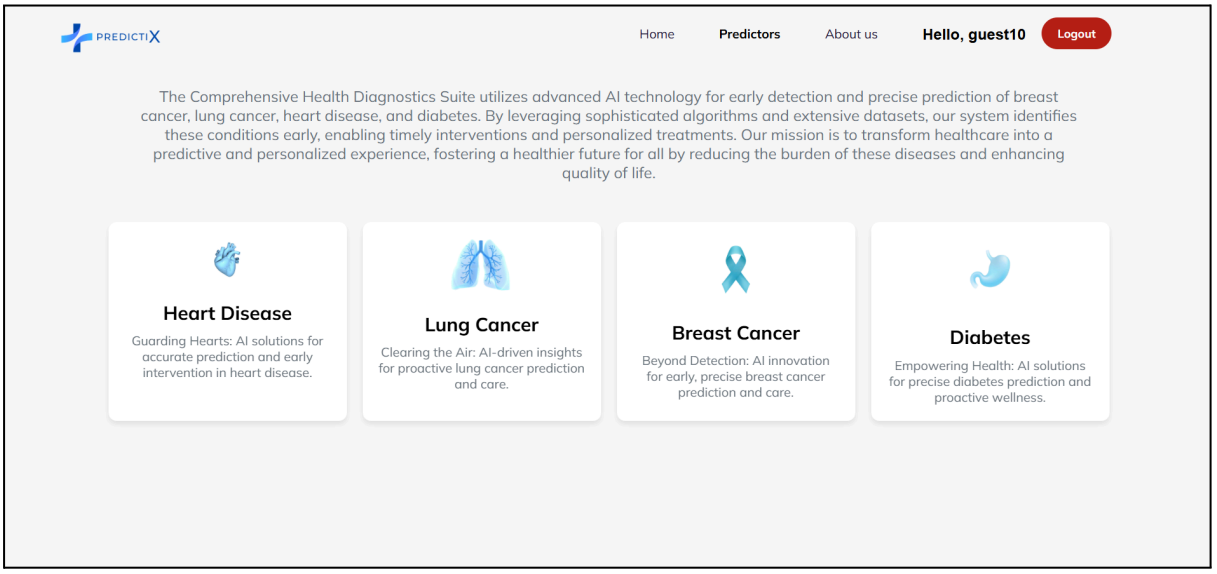

Fig 6: Disease Predictors

9.3.1 Heart disease predictor

Heart Disease Predictor: Analyzes clinical indicators like age, cholesterol, and blood pressure using a decision tree model to estimate the risk of heart disease.

Users can either manually enter the values or they can upload their reports.

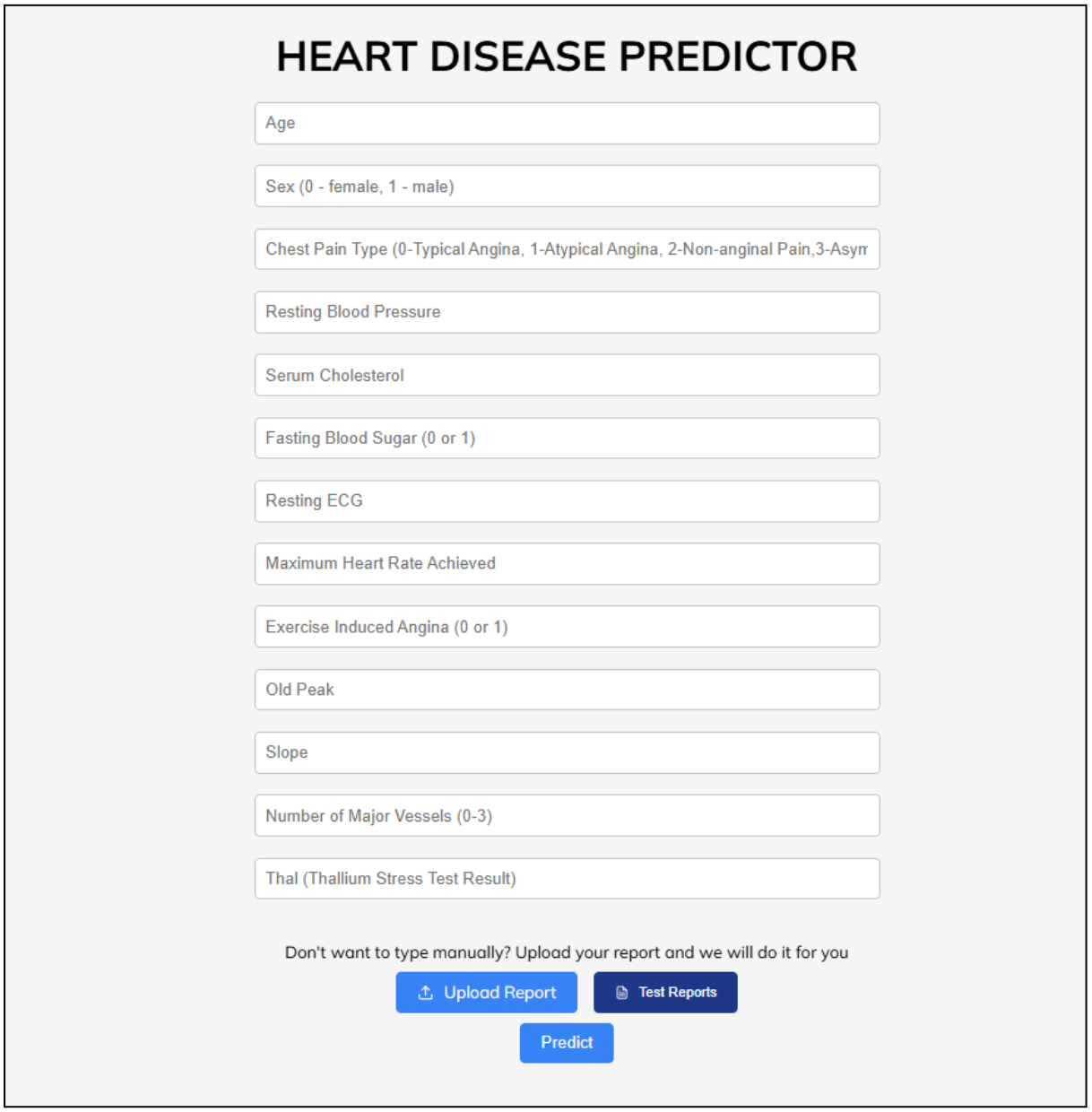

Fig 7: Heart Disease Predictor

The snapshots show Predictix results for two cases: one with a positive heart disease prediction and one with no detected heart disease.

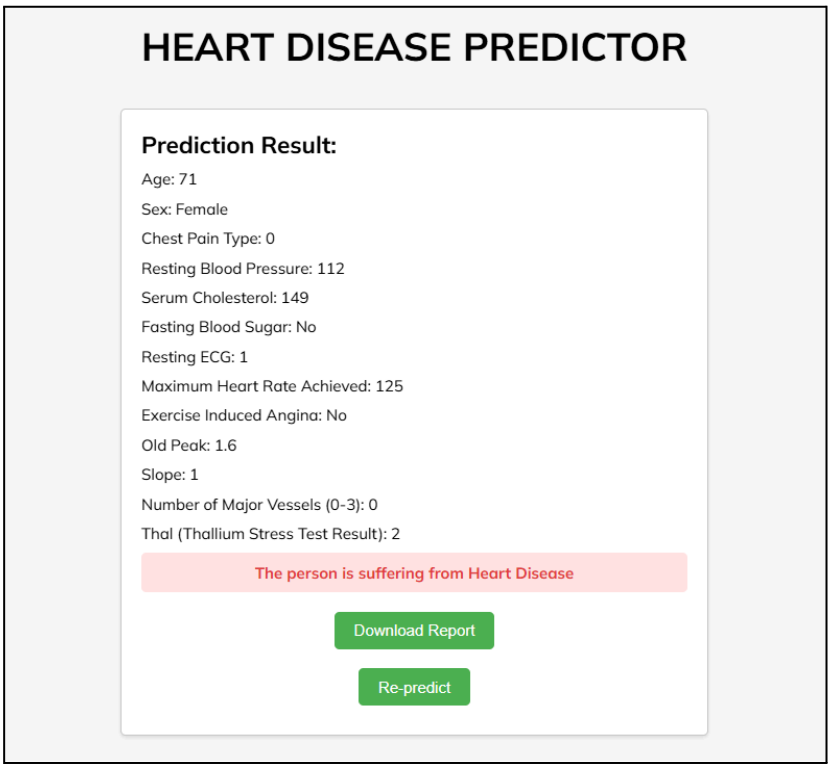

Fig 8: Prediction – Person Diagnosed with Heart Disease

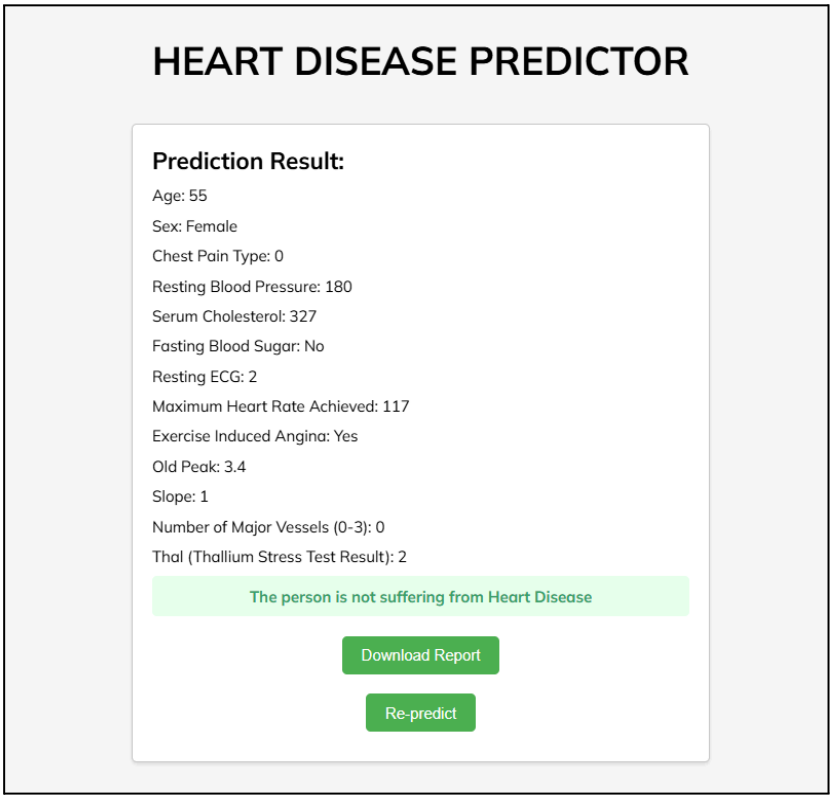

Fig 9: Prediction – Person Not Diagnosed with Heart Disease

9.3.2 Lung cancer predictor

Lung Cancer Predictor: Leverages a CNN-based image classification model to detect early signs of lung cancer from medical scans for timely intervention.

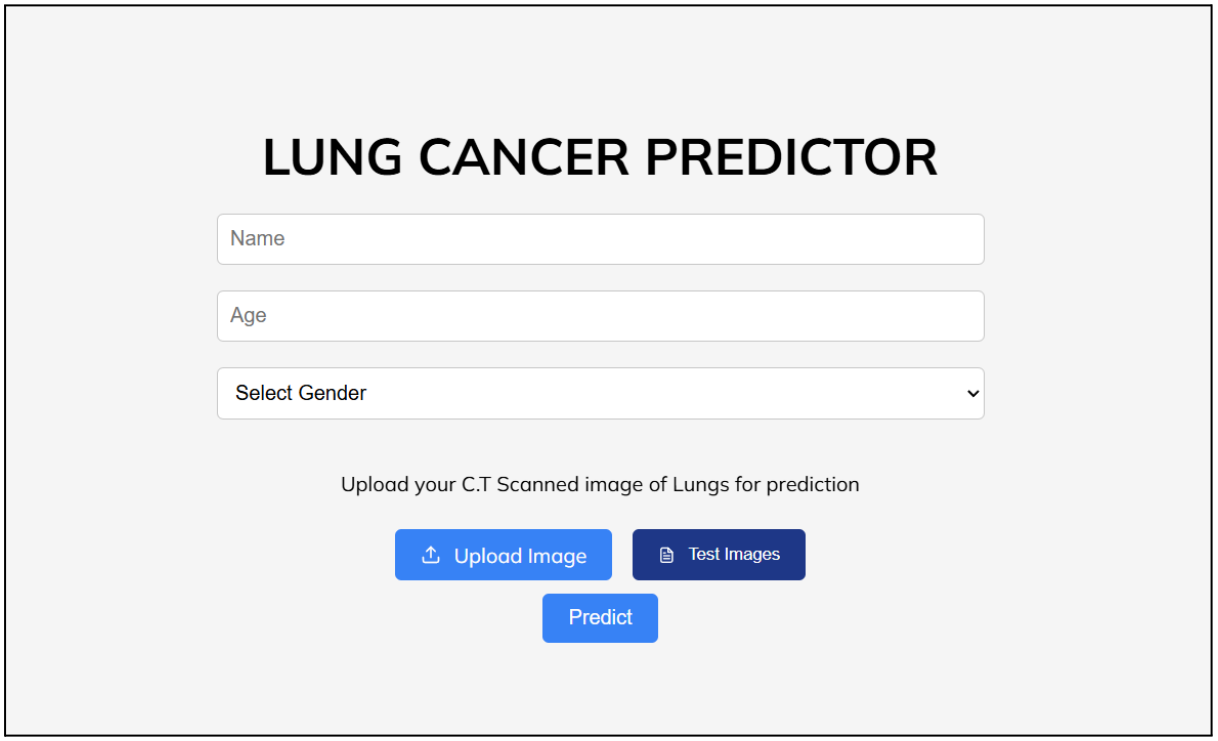

Fig 10: Lung Cancer Predictor

The following snapshots show Predictix's output for two cases: one indicating lung cancer and another indicating no lung cancer.

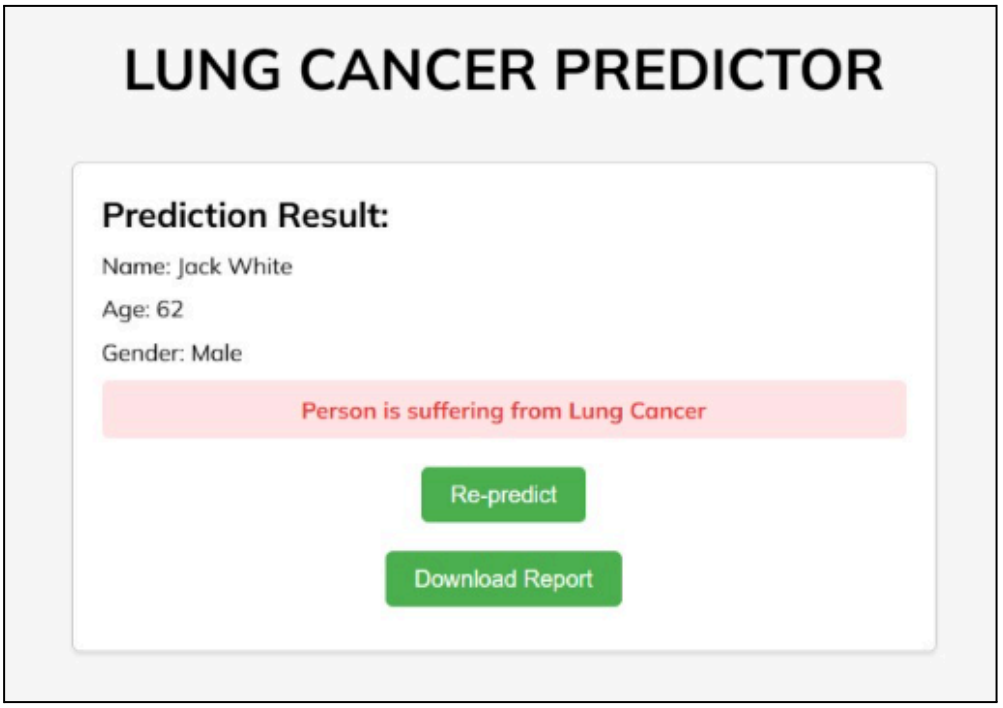

Fig 11: Prediction – Person Suffering from Lung Cancer

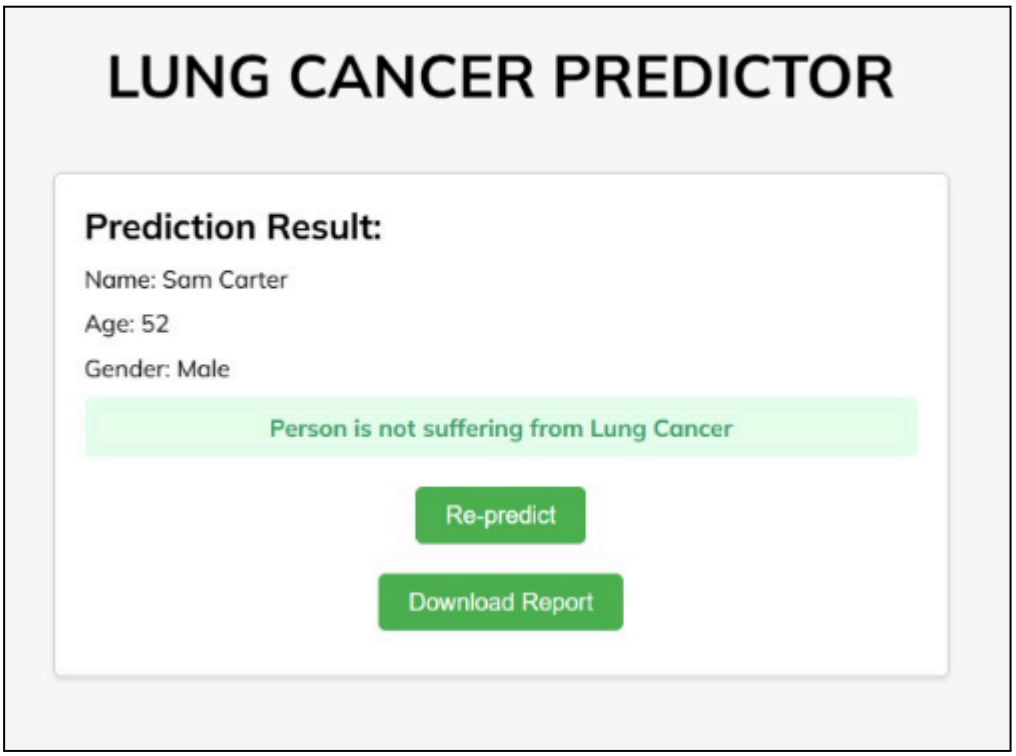

Fig 12: Prediction – Person Not Suffering from Lung Cancer

9.3.3 Breast cancer predictor

Breast Cancer Predictor: Processes diagnostic features through machine learning algorithms to assess the likelihood of benign or malignant breast cancer.

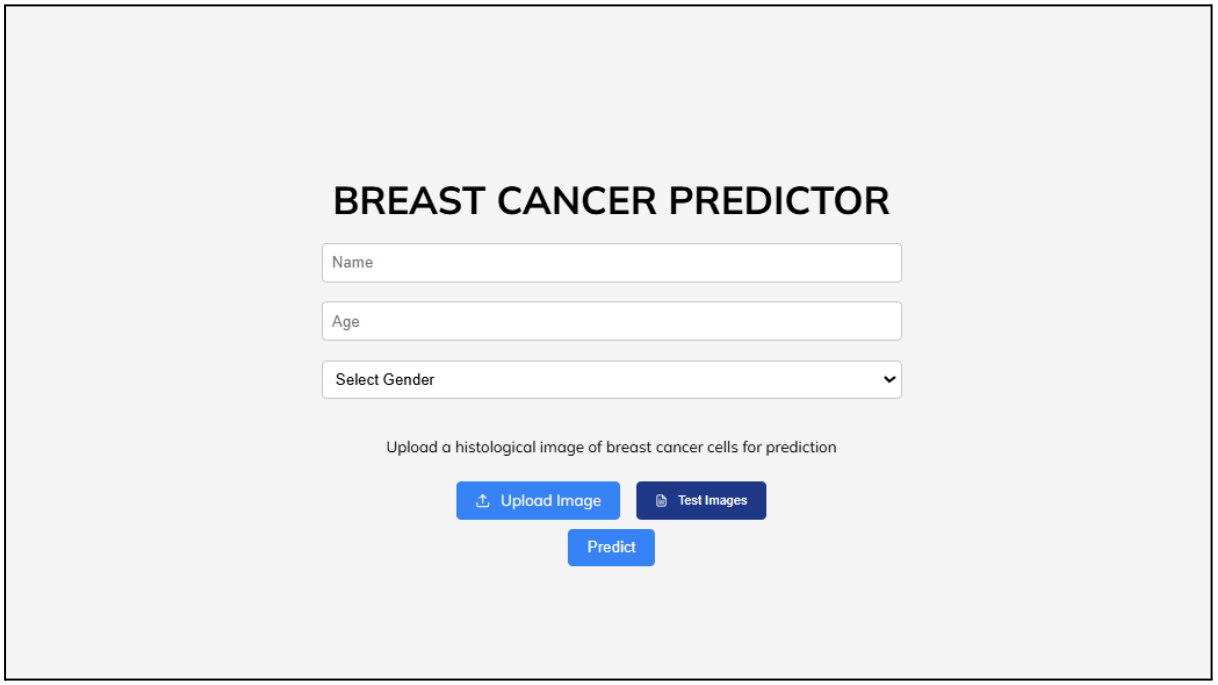

Fig 13: Breast Cancer Predictor

The following snapshots show Predictix's output for two cases: one indicating breast cancer and another indicating no breast cancer.

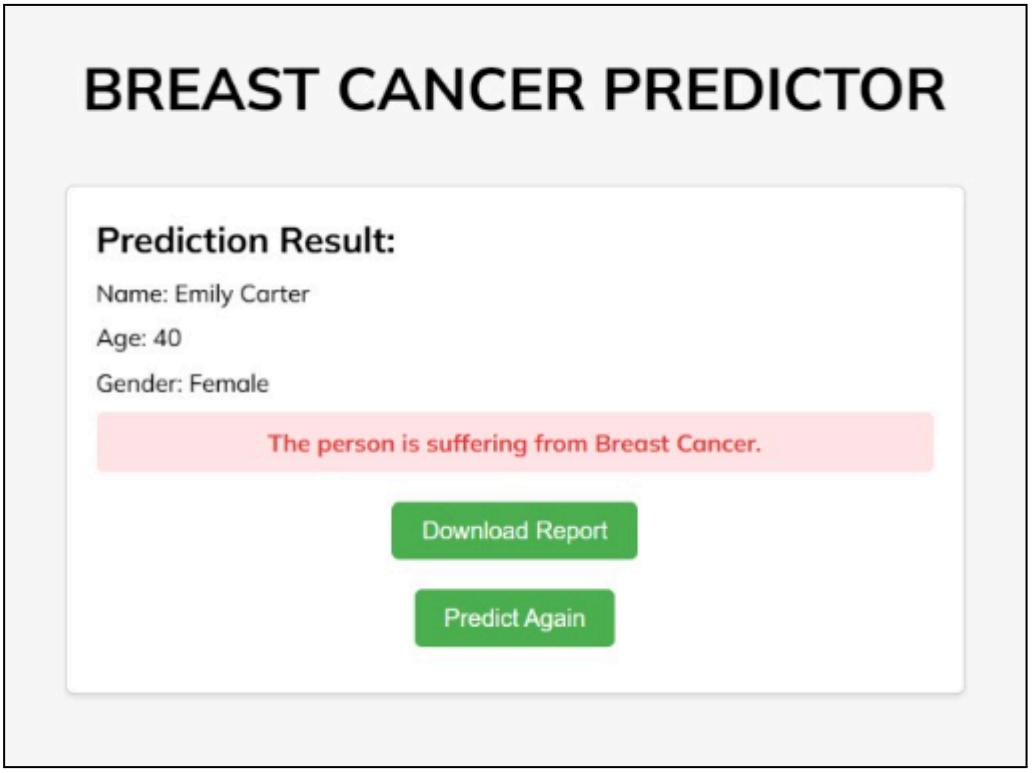

Fig 14: Prediction – Person Suffering from Breast Cancer

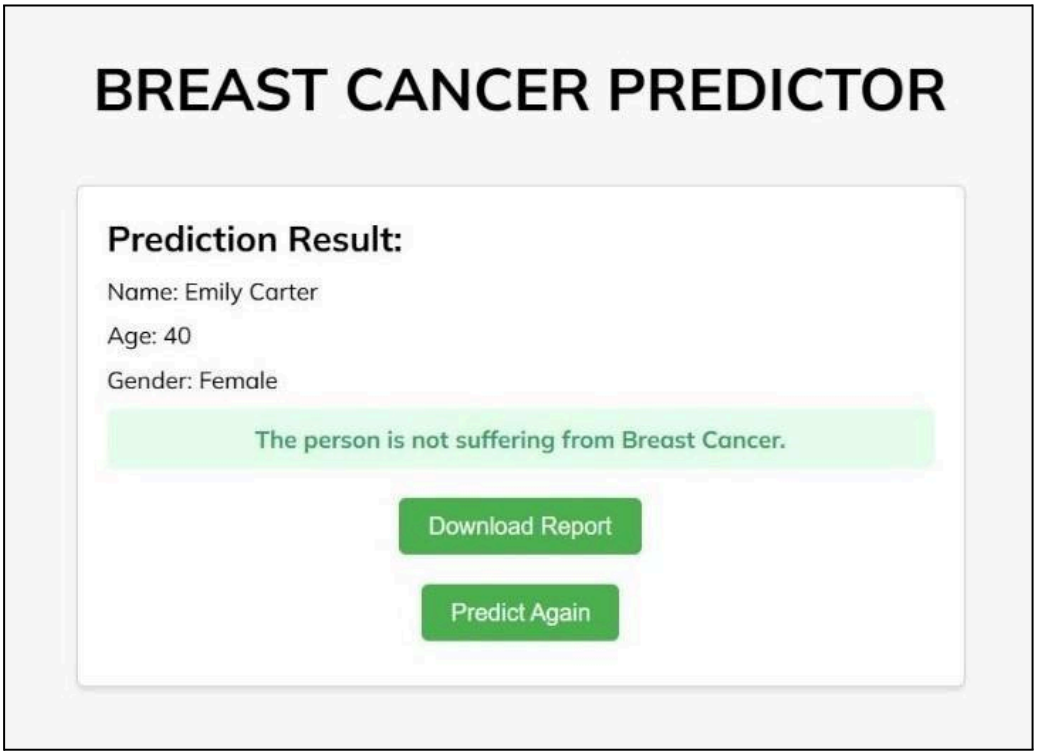

Fig 15: Prediction – Person Not Suffering from Breast Cancer

9.3.4 Diabetes predictor

Diabetes Predictor: Utilizes patient data such as glucose levels, BMI, and insulin history to predict the chances of diabetes with reliable accuracy.

Users can either manually enter the values or they can upload their reports.

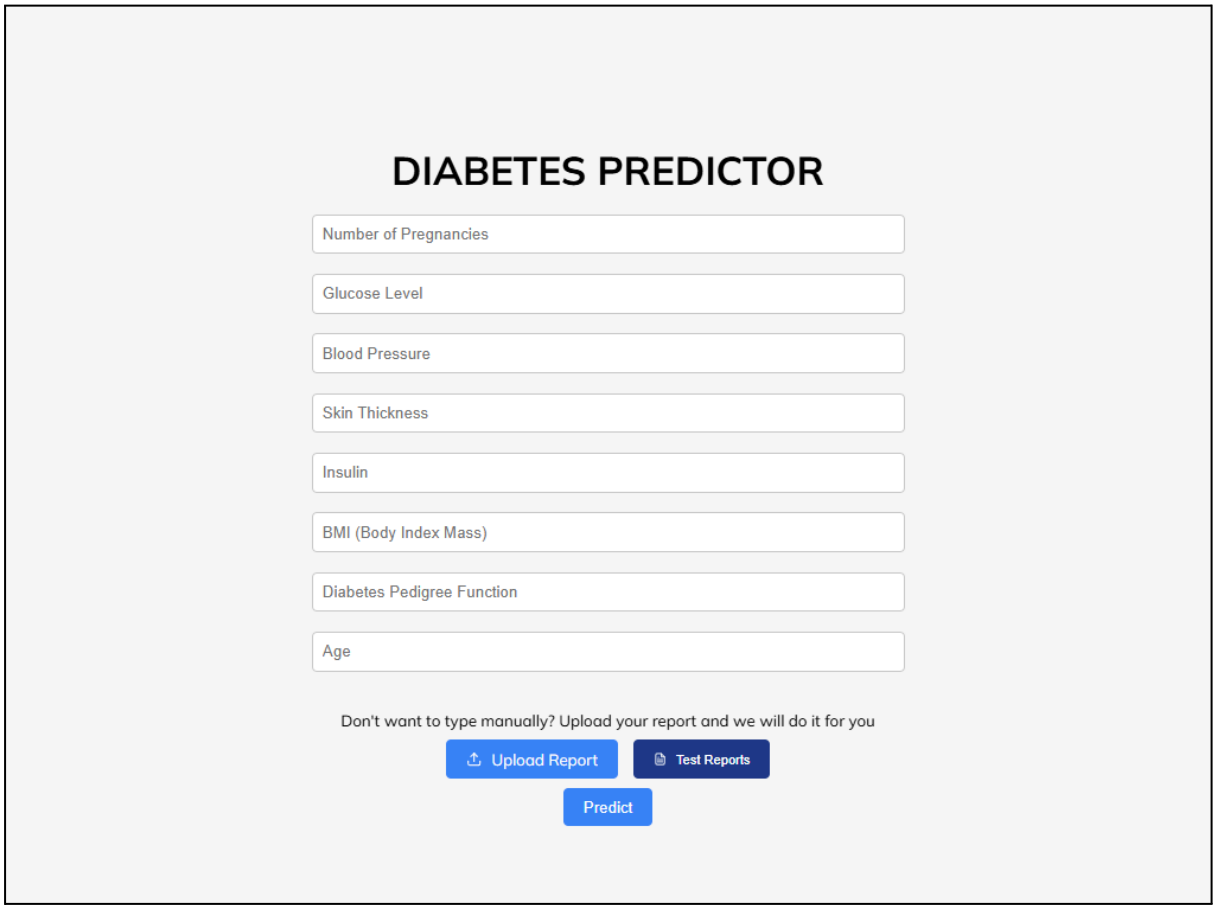

Fig 16: Diabetes Predictor

The following snapshots illustrate the output generated by the Predictix system for two different cases: one where the user is predicted to be suffering from diabetes, and another where the system predicts no presence of diabetes.

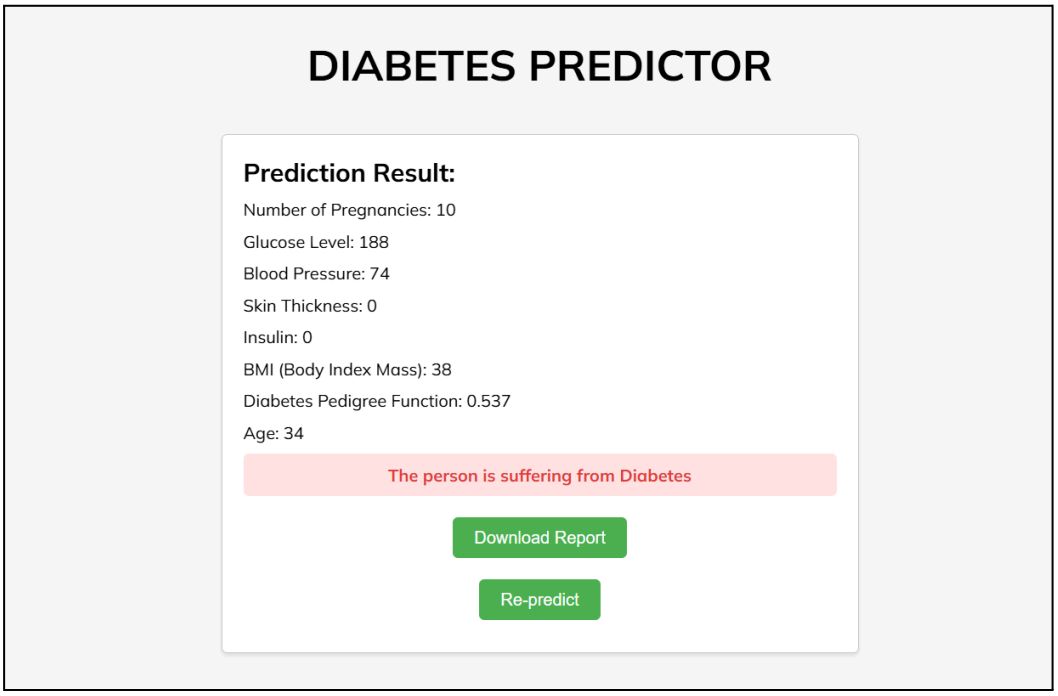

Fig 17: Prediction – Person Diagnosed with Diabetes

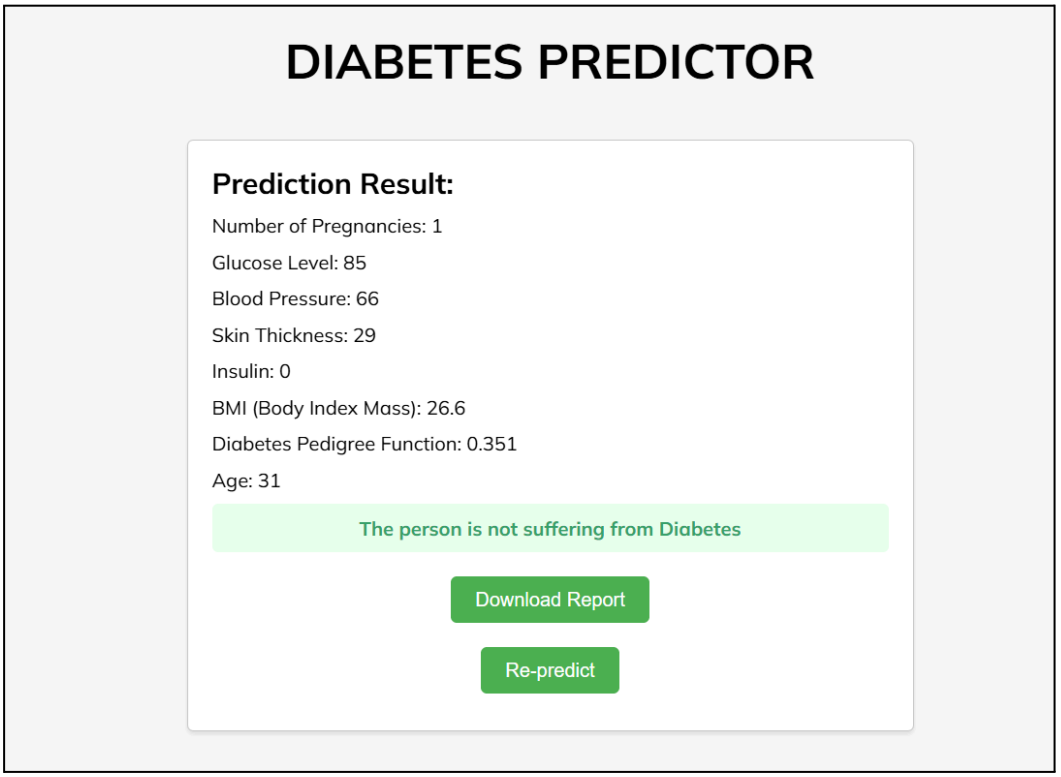

Fig 18: Prediction – Person Not Diagnosed with Diabetes

9.4 Download Report

Report: Provides a clear prediction result indicating whether the person is likely affected or not, and allows users to download the report for future reference or medical consultation.

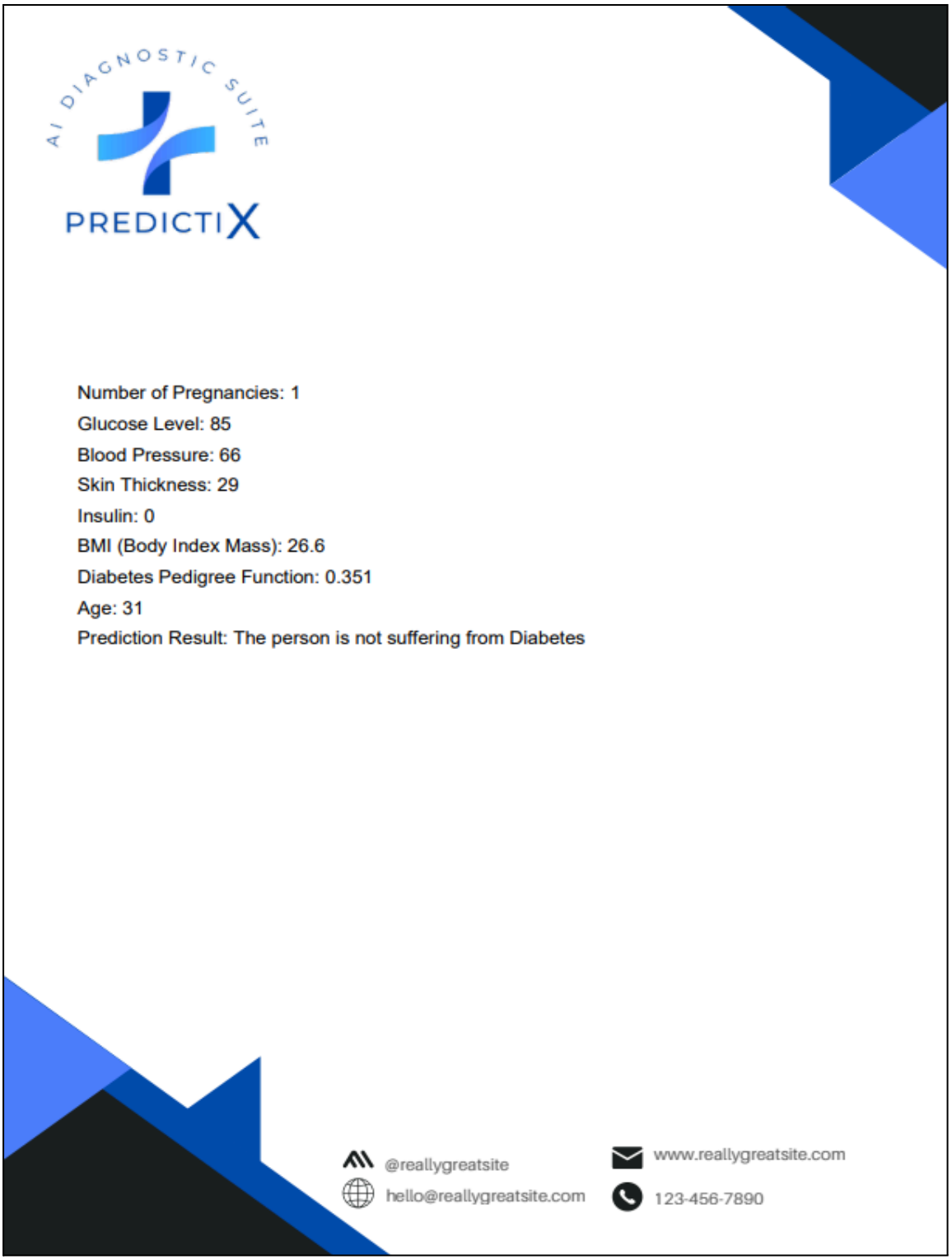

Fig 19: Downloaded Report

9.5 About us page

About Us Page: Highlights the vision behind Predictix and introduces the team dedicated to make the application more accessible and user-friendly.

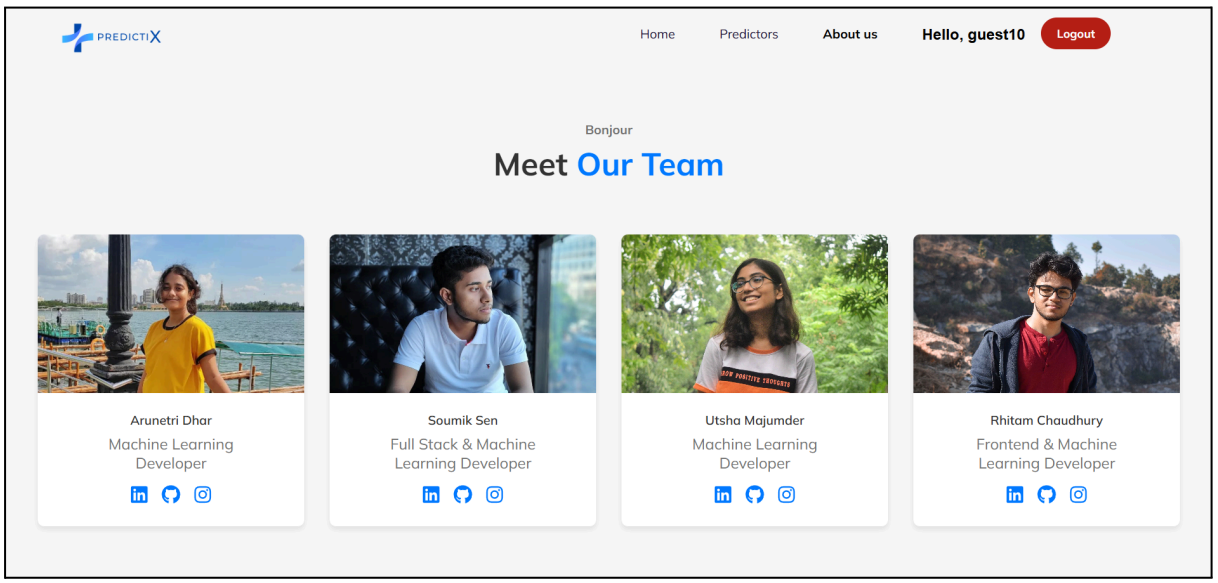

Fig 20: About Us Page

10. CONCLUSION

Predictix is a thoughtfully designed health prediction system created to support early detection and awareness of four serious medical conditions: **breast cancer**, **lung cancer**, **diabetes**, and **heart disease**. The system focuses on simplifying health assessment by allowing users to enter basic medical details or upload structured reports to receive instant, understandable results.

The primary aim of Predictix is to promote preventive healthcare by helping individuals recognize potential risks at an early stage. In doing so, it encourages timely medical consultation and responsible health decisions. Whether used by the general public or healthcare professionals, the system offers a quick and accessible way to gain insight into one's health status.

By transforming raw health information into meaningful outcomes, Predictix serves as a bridge between the user and early medical awareness. It provides an opportunity for users to be more proactive about their well-being, especially in situations where regular check-ups may not be immediately available.

Overall, Predictix is not just a prediction system—it is a step toward empowering individuals with health knowledge, fostering awareness, and making preventive care more approachable and informed.

11. FUTURE SCOPE

While Predictix currently performs as a robust prototype, several enhancements can be made to further improve its functionality, scalability, and clinical relevance:

- **OCR Integration**: Implementing Optical Character Recognition (OCR) to allow users to upload scanned or printed medical reports in diverse formats for automated data extraction.

- **Expanded Disease Coverage**: Incorporating prediction models for additional diseases such as kidney disorders, liver disease, or stroke to broaden the system's medical utility.

- **Multilingual Support**: Making the platform accessible to a wider audience by supporting regional languages and simplified medical terms.

12. REFERENCES

[1] F. A. Spanhol, L. S. Oliveira, C. Petitjean and L. Heutte, "Breast cancer histopathological image classification using Convolutional Neural Networks," **2016 International Joint Conference on Neural Networks (IJCNN), Vancouver, BC, Canada**, 2016, pp. 2560-2567, doi: 10.1109/IJCNN.2016.7727519.

[2] Forderation, **BreakHis – Breast Cancer Histopathological Image Dataset (400x)**, Kaggle, 2023.
Link : https://www.kaggle.com/datasets/forderation/breakhis-400x

[3] Vinayak, A. Rai, A. Khan, A. Srivastava, and R. B. Singh, "Early Stage Lung Cancer Detection Using ResNet," **International Research Journal of Modernization in Engineering, Technology and Science (IRJMETS)**, vol. 6, no. 5, pp. 2600–2603, May 2024.
DOI: https://www.doi.org/10.56726/IRJMETS56243.

[4] M. Nickparvar, **Lung Cancer CT Scan Image Dataset**, Kaggle, 2021.
Link : https://www.kaggle.com/datasets/hamdallak/the-iqothnccd-lung-cancer-dataset

[5] A. Srivastava, A. Tripathi, and A. Singh, "A Review of Diabetes Prediction System Using SVM Machine Learning Algorithm," **International Research Journal of Modernization in Engineering, Technology and Science**, vol. 6, no. 5, pp. 8422–8428, May 2024.
DOI: https://www.doi.org/10.56726/IRJMETS57826

[6] UCI Machine Learning Repository, **Pima Indians Diabetes Database**, Kaggle.
Link : https://www.kaggle.com/datasets/uciml/pima-indians-diabetes-database

[7] R. V. Saraswathi, K. Gajavelly, A. K. Nikath, R. Vasavi, and R. R. Anumasula, "Heart Disease Prediction Using Decision Tree and SVM," in **Proceedings of the Second International Conference on Advances in Computer Engineering and Communication Systems**, Algorithms for Intelligent Systems, Singapore: Springer, 2022, pp. 69–78.
DOI: 10.1007/978-981-16-7389-4_7.

[8] johnsmith88, **Heart Disease Dataset**, Kaggle, 2018.
Link : https://www.kaggle.com/datasets/johnsmith88/heart-disease-dataset