TABLE OF CONTENTS

List of Figures

Chapter-1
Introduction

Chapter-1
Introduction

1.1 Overview and issues involved

Disease Prediction using Machine Learning is a system which predicts the disease based on the information provided by the user. It also predicts the disease of the patient or the user based on the information enter into the system and provides the accurate results based on that information. If the patient is not much serious and the user just wants to know the type of disease, he/she has been through. This Disease Prediction Using Machine Learning is completely done with the help of Machine Learning and Python Programming language and also using the dataset that available on kaggle using that we will predict the disease. Now a day's doctors are adopting many scientific technologies and methodology for both identification and diagnosing not only common disease, but also many fatal diseases. The successful treatment is always attributed by right and accurate diagnosis, pathologists may sometimes fail to take accurate decisions while diagnosing the disease of a patient, therefore disease prediction systems help to get the result with high accuracy.

Medicine and healthcare are some of the most crucial parts of the economy and human life. There is a tremendous amount of change in the world we are living in now and the world that existed a few weeks back. Everything has turned gruesome and divergent. In this situation, where everything has turned virtual, the doctors and nurses are putting up maximum efforts to save people’s lives even if they have to danger their own. There are also some remote villages that lack medical facilities. Machines are always considered better than humans as, without any human error, they can perform tasks more efficiently and with a consistent level of accuracy. A disease predictor can be called a virtual pathologist, which can predict the disease of any patient without any human error.

Many of the existing models are concentrating on one disease per analysis. Like one analysis for diabetes analysis, one for heart disease analysis, and one for breast disease like that. There

is no common system present that can analyze more than one disease at a time. Thus, we are concentrating on providing immediate and accurate disease predictions to the users about the symptoms they enter along with the disease predicted.

Breast cancer, diabetes, and heart disease are for the most part driving reasons for death in the present society. Heart disease is a term that refers to a group of illnesses that affect your heart. 2.4 lakh breast cancer cases are to be expected to be the most common site in India by 2025. Diabetes is a type of disease that occurs when your glucose level in the blood, also called blood sugar is very high. Accurate and on-time analysis of any health-related problem is important for the prevention and treatment of the illness. In this digital world, data is an asset, and enormous data was generated in all the fields. Data in the healthcare industry consists of all the information related to patients. Here a general architecture has been proposed for predicting the disease in the healthcare industry.

1.2 Problem definition

Clinical decisions are often made based on doctors' intuition and experience rather than on the knowledge-rich data hidden in the database. This practice leads to unwanted biases, errors, and excessive medical costs which affects the quality of service provided to patients. We have proposed that the integration of clinical decision support with computer-based patient records could reduce medical errors, enhance patient safety, decrease unwanted practice variation, and improve patient outcomes. This suggestion is promising as data analysis have the potential to generate a knowledge-rich environment that can help to significantly improve the quality of clinical decisions.

Many of the existing machine learning models for health care analysis are concentrating on one disease per analysis. For example first is for liver analysis, one for cancer analysis, and one for lung diseases like that. If a user wants to predict more than one disease, he/she has to go through different sites. There is no common system where one analysis can perform more than one disease prediction. Some of the models have lower accuracy which can seriously affect patients' health. When an organization wants to analyze their patient's health report they

have to deploy many models which in turn increases the cost as well as the time Some of the existing systems consider very few parameters which can yield false results.

Now a day's in Health Industry there are various problems related to machines or device which will give wrong or unaccepted results, so to avoid those results and get the correct and desired results we are building a program or project which will give accurate predictions based on information provided by the user. So, with the help of algorithms, techniques, and methodologies we have done this project.

1.3 Proposed solution

In multiple disease prediction, it is possible to predict more than one disease at a time. So the user doesn't need to traverse different sites in order to predict the diseases. We are taking three diseases that are Breast Cancer, Diabetes, and Heart. we are concentrating on providing immediate and accurate disease predictions to the users about the symptoms they enter along with the disease predicted. So, we are proposing a system which used to predict multiple diseases by using Flask. In this system, we are going to analyze Diabetes, Heart, Breast, and Kidney disease analysis. Later many more diseases can be included. To implement multiple disease prediction system we are going to use machine learning algorithms and Flask.

If this Prediction is done at the early stages of the disease with the help of this project and all other necessary measures the disease can be cured and in general this prediction system can also be very useful in the health industry. The general purpose of this Disease prediction is to provide prediction for the various and generally occurring diseases that when unchecked and sometimes ignored can turns into fatal disease and cause lot of problem to the patient and as well as their family members. This system will predict the most possible disease based on the information provided by user. Here we can easily prepare the data and transform that data into algorithm, which will reduce the overall work of the project.

This is done based on the previous datasets of the hospitals so after comparing it can provide up to 97% of accurate results for breast cancer, 87% of accurate results for heart disease, and

77% accuracy for diabetes, and the project is still developing further get 100% accurate results.

A brief summary of the major product functions and what the end user may perform on the application include:

- Users can go to the prediction page of the disease they wish to get the result
- Take out their report, and fill the values from the report into the form present on their screen
- Click on the prediction button, and the result will be on the screen. If the disease is the result is positive then precautions will also be shown.

Flow Diagram –

Figure 1.1 (Flow Diagram -1)

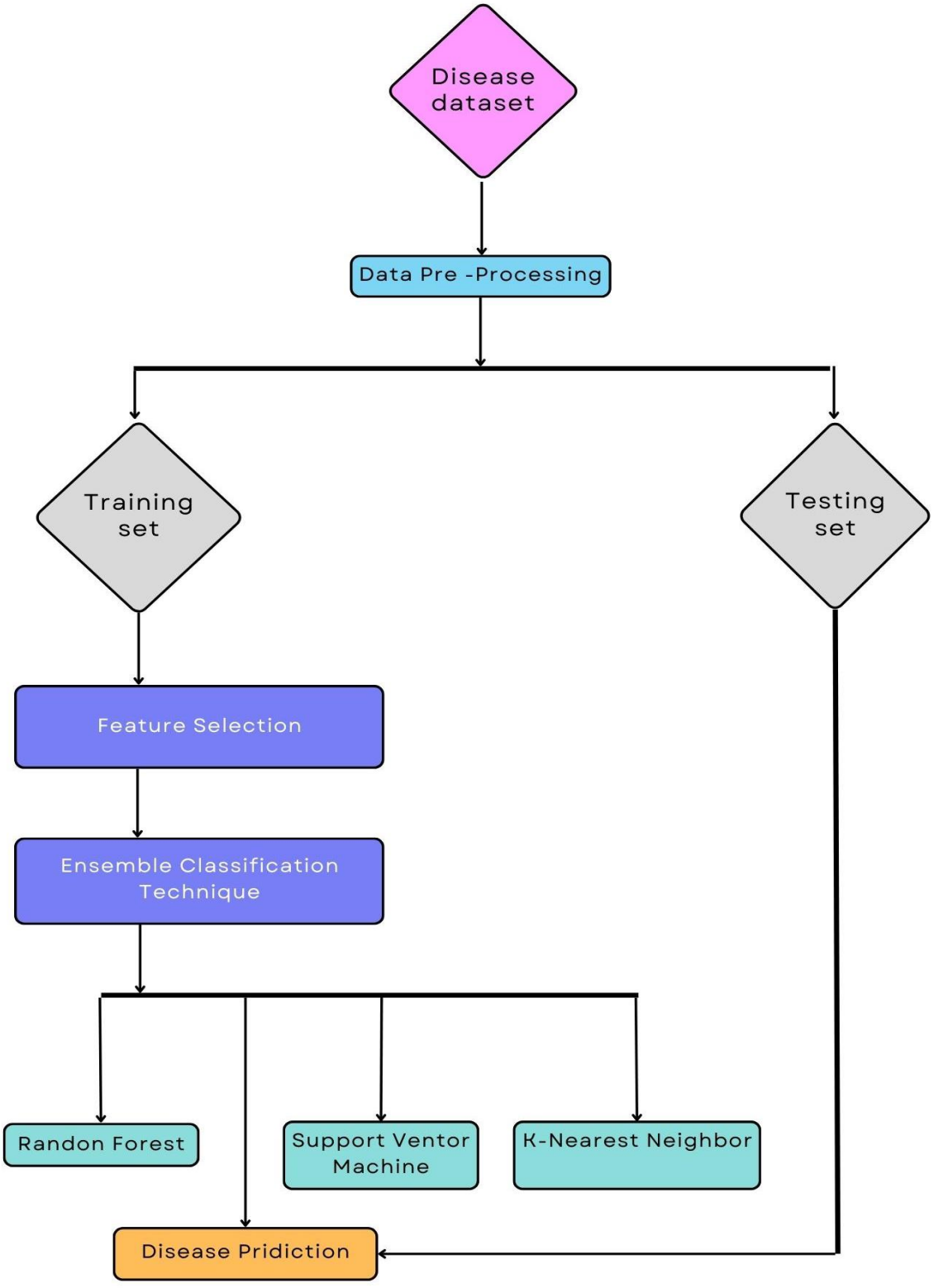

Figure 1.2(Flow Diagram - 2)

Chapter – 2
Literature Survey

Chapter-2
Literature Survey

2.1 Methodology

The proposed system for disease prediction. The symptoms of an individual along with the age and gender can be given to the ML model to further process. After preliminary processing of the data, the ML model uses the current input, trains, and tests the algorithm resulting in the predicted disease.

The dataset consisting of the gender, symptoms, and age of an individual was preprocessed and fed as input to different ML algorithms for the prediction of the disease. The different ML

models used were Decision trees, Random forest classifiers, and logistic regression. The outcome of the models is the disease as per the symptoms, age, and gender given to the processing model.

Functioning of the ML models. The dataset was split into input consisting of age, gender, and symptoms and the output as the diseases based on the input factors. We randomly split the available data into train and test sets. These sets were then encoded and further trained using different algorithms. After which the algorithms test the training set and predict the values, resulting in the accuracy of different ML algorithms. The predicted values were then decoded to give the output as the disease.

2.1.1 Algorithms used-

- **KNN -** K-nearest neighbors (KNN) rule could be a variety of supervised metric capacity unit rules which may be used for each classification yet as regression prophetical issues. However, it's primarily used for the classification of prophetical issues in trade. the subsequent 2 properties would outline KNN well –

• Lazy learning algorithm − KNN could be a lazy learning rule as a result of it doesn't have a specialized coaching section and uses all the information for coaching whereas classification.

• Non-parametric learning algorithm − KNN is additionally a non-parametric learning rule as a result it doesn't assume something regarding the underlying knowledge.

- **Random Forest Algorithm** - Random Forest could be a learning methodology that operates by constructing multiple call trees. the ultimate call is created to support the bulk of the trees and is chosen by the random forest. There square measure heaps of advantages to the victimization Random Forest algorithmic rule, however, one of the most benefits is that it reduces the chance of overfitting and therefore the needed coaching time. in addition, it offers a high level of accuracy. Random Forest the algorithmic rule runs with efficiency in massive information bases and produces extremely correct predictions by estimating missing data.

- **Logistic Regression** - Logistic regression is basically a supervised classification algorithm. In a classification problem, the target variable(or output), y, can take only discrete values for a given set of features(or inputs), X. Contrary to popular belief, logistic regression IS a regression model. The model builds a regression model to predict the probability that a given data entry belongs to the category numbered as "1".

- **Support Vector Machine** - Support Vector Machine or SVM is one of the most popular Supervised Learning algorithms, which is used for Classification as well as Regression problems. However, primarily, it is used for Classification problems in Machine Learning.

  The goal of the SVM algorithm is to create the best line or decision boundary that can segregate n-dimensional space into classes so that we can easily put the new data point in the correct category in the future. This best decision boundary is called a hyperplane.

  SVM chooses the extreme points/vectors that help in creating the hyperplane. These extreme cases are called as support vectors, and hence algorithm is termed as Support Vector Machine.

2.1.2 Data Set –

- **Breast Cancer dataset –** Breast Cancer Wisconsin (Diagnostic) dataset [1]

  - Features are computed from a digitized image of a fine needle aspirate (FNA) of a breast mass. They describe the characteristics of the cell nuclei present in the image in the 3-dimensional space is that described in [2]
  - Also can be found on UCI Machine Learning [3]

  Attribute Information:

  - ID number
  - Diagnosis (M = malignant, B = benign
  - Ten real-valued features are computed for each cell nucleus:

    a) radius (mean of distances from center to points on the perimeter)

    b) texture (standard deviation of gray-scale values)

    c) perimeter

    d) area

    e) smoothness (local variation in radius lengths)

    f) compactness (perimeter^2 / area - 1.0)

    g) concavity (severity of concave portions of the contour)

    h) concave points (number of concave portions of the contour)

    i) symmetry

    j) fractal dimension ("coastline approximation" - 1)

  - All feature values are recoded with four significant digits.
  - Missing attribute values: none
  - Class distribution: 357 benign, 212 malignant
  - **Key Attributes:**

    a) Texture

    b) Perimeter

    c) Smoothness

    d) Compactness

    e) Symmetry

- **Heart Disease dataset** – What causes heart disease? Explaining the model [4]

  - It's a clean, easy-to-understand set of data. However, the meaning of some of the column headers are not obvious. Here's what they mean,

  - **Attribute Information (Key Attributes):**

    a) age: The person's age in years

    b) sex: The person's sex (1 = male, 0 = female)

    c) chest pain : The chest pain experienced (Value 1: typical angina, Value 2: atypical angina, Value 3: non-anginal pain, Value 4: asymptomatic)

    d) Resting BP: The person's resting blood pressure (mm Hg on admission to the hospital)

    e) Serum Cholesterol: The person's cholesterol measurement in mg/dl

    f) Fasting Blood Sugar: The person's fasting blood sugar (> 120 mg/dl, 1 = true; 0 = false)

    g) Electro Cardio Graphy: Resting electrocardiographic measurement (0 = normal, 1 = having ST-T wave abnormality, 2 = showing probable or definite left ventricular hypertrophy by Estes' criteria)

    h) Maximum Heart Rate: The person's maximum heart rate achieved

    i) Exercise induced angina: (1 = yes; 0 = no)

    j) ST depression: ST depression induced by exercise relative to rest ('ST' relates to positions on the ECG plot)

    k) Slope: the slope of the peak exercise ST segment (Value 1: upsloping, Value 2: flat, Value 3: downsloping)

    l) Major vessels Colored: The number of major vessels (0-3)

    m) Defect: A blood disorder called thalassemia (3 = normal; 6 = fixed defect; 7 = reversable defect)

    n) target: Heart disease (0 = no, 1 = yes)

- **Diabetes dataset -**
  - This dataset is originally from the National Institute of Diabetes and Digestive and Kidney Diseases. The objective of the dataset is to diagnostically predict whether a patient has diabetes, based on certain diagnostic measurements included in the dataset. Several constraints were placed on the selection of these instances from a larger database.[5]

  - **Attribute Information (Key Attributes):**

    a) Pregnancies: To express the Number of pregnancies

    b) Glucose: To express the Glucose level in blood

    c) Blood Pressure: To express the Blood pressure measurement

    d) Skin Thickness: To express the thickness of the skin

    e) Insulin: To express the Insulin level in blood

    f) BMI: To express the Body mass index

    g) Diabetes Pedigree Function: To express the Diabetes percentage

    h) Age: To express the age

2.2 Existing Solutions

A structural model and a collection of conditional probabilities are used by Bayesian classifiers. They make the assumption that the contributions of all factors are independent. It first calculates the prior probability for each class, and then applies the occurrence of each variable value to an unknown scenario. A Bayes network classifier is built on a Bayesian network, which reflects a joint probability distribution over a set of category characteristics.

The fuzzy technique with a membership function was used to forecast cardiac disease [6]. Using the Fuzzy KNN Classifier, the authors attempted to eliminate ambiguity and uncertainty from data. The 550-record dataset was separated into 25 classes, with each class having 22 items. The dataset was separated into two equal parts: training and testing. The fuzzy KNN

methodology was implemented after pre-processing techniques were used. This technique was examined using several assessment metrics such as accuracy, precision, and recall, among others. Based on the data, it was discovered that the fuzzy KNN classifier outperformed the KNN classifier in terms of accuracy.

For the prediction of cardiac disease, a novel technique based on the ANN algorithm was devised [7]. The researchers created an interactive prediction method based on categorization using an artificial neural network algorithm and taking into account the thirteen most important clinical parameters. The suggested method proved effective for predicting heart disease with an accuracy of 80% and can be very useful for healthcare practitioners.

Authors in [8] presented an automated approach for answering difficult inquiries for heart disease prediction. The Naive Bayes methodology was used to create this intelligent system in order to provide quick, better, and more accurate outcomes.

It might aid doctors in making clinical judgments about heart attacks. This system may be enhanced by including SMS
functionality, building Android and IOS mobile applications.

Diabetes and breast cancer were diagnosed by incorporating the adaptivity characteristic into support vector machines [9]. The goal was to offer a rapid, automated, and adaptable diagnostic method using adaptive SVM. To achieve better results, the bias value in conventional SVM was changed. The suggested classifier produced output in the form of 'if-then' rules. The proposed method was used to diagnose diabetes and breast cancer, and it provided 100% right classification rates for both conditions. Future research should focus on developing more efficient ways for changing the bias value in conventional SVM.

For the prediction of type 2 diabetes, a hybrid model based on clustering followed by classification was proposed [10]. For prediction, the suggested model uses K-means clustering and the C4.5 classification method with k-fold cross-validation. The model generated encouraging results with a classification accuracy of 88.38 percent using the hybrid technique, which might be highly useful for clinicians in making appropriate clinical choices related to diabetes.

Chapter – 3

Analysis and Design

Chapter-3
Analysis and Design

3.1 Software Requirements

- **Flask -** Flask is a micro web framework written in Python. It is classified as a micro framework because it does not require particular tools or libraries.[2] It has no database abstraction layer, form validation, or any other components where pre-existing third-party libraries provide common functions. However, Flask supports extensions that can add application features as if they were implemented in Flask itself. Extensions exist for object-relational mappers, form validation, upload handling, various open authentication technologies, and several common framework-related tools.

- **Numpy** - NumPy is a library for the Python programming language, adding support for large, multi-dimensional arrays and matrices, along with a large collection of high-level mathematical functions to operate on these arrays. It also has functions for working in the domain of linear algebra, Fourier transform, and matrices.

- **Pandas –** Pandas is a fast, powerful, flexible, and easy-to-use open-source data analysis and manipulation tool, built on top of the python programming language. In particular, it offers data structures and operations for manipulating numerical tables and time series. Pandas allows allows data manipulation operations such as merging, reshaping, data-cleaning, etc. The development of pandas introduced into Python many comparable features of working with Data Frames that were established in the R programming language. The pandas library is built upon another library NumPy, which is oriented to efficiently working with arrays instead of the features of working on Data Frames.

- **Scikit -** Scikit-learn (Sklearn) is the most useful and robust library for machine learning in Python. It provides a selection of efficient tools for machine learning

and statistical modeling including classification, regression, clustering and dimensionality reduction via a consistence interface in Python. This library, which is largely written in Python, is built upon NumPy, SciPy and Matplotlib. It features various algorithms including Support Vector Machine, Random Forest Classifier, and K-Means, etc.

- **HTML** – It is used for describing the structure of web pages. It provides a basic structure of the page upon which CSS (Cascading Style Sheets) are overlaid to change its appearance.
- **CSS** – It is used for describing the presentation of web pages, including colors, layouts, and fonts
- **Bootstrap –** It helps accelerate the design process by providing pre-made design themes that are customizable and easy to use.

3.2 Hardware Requirements

The application can be used on any personal computer, laptop, smartphone, or any similar device. It does not require any specialized hardware for its work.

- Minimum 2 GB RAM required.
- Internet Connection
- Being a responsive website there is no constraints for the screen size of the device.

3.3 Analysis Diagrams

3.3.1 Use Case Model

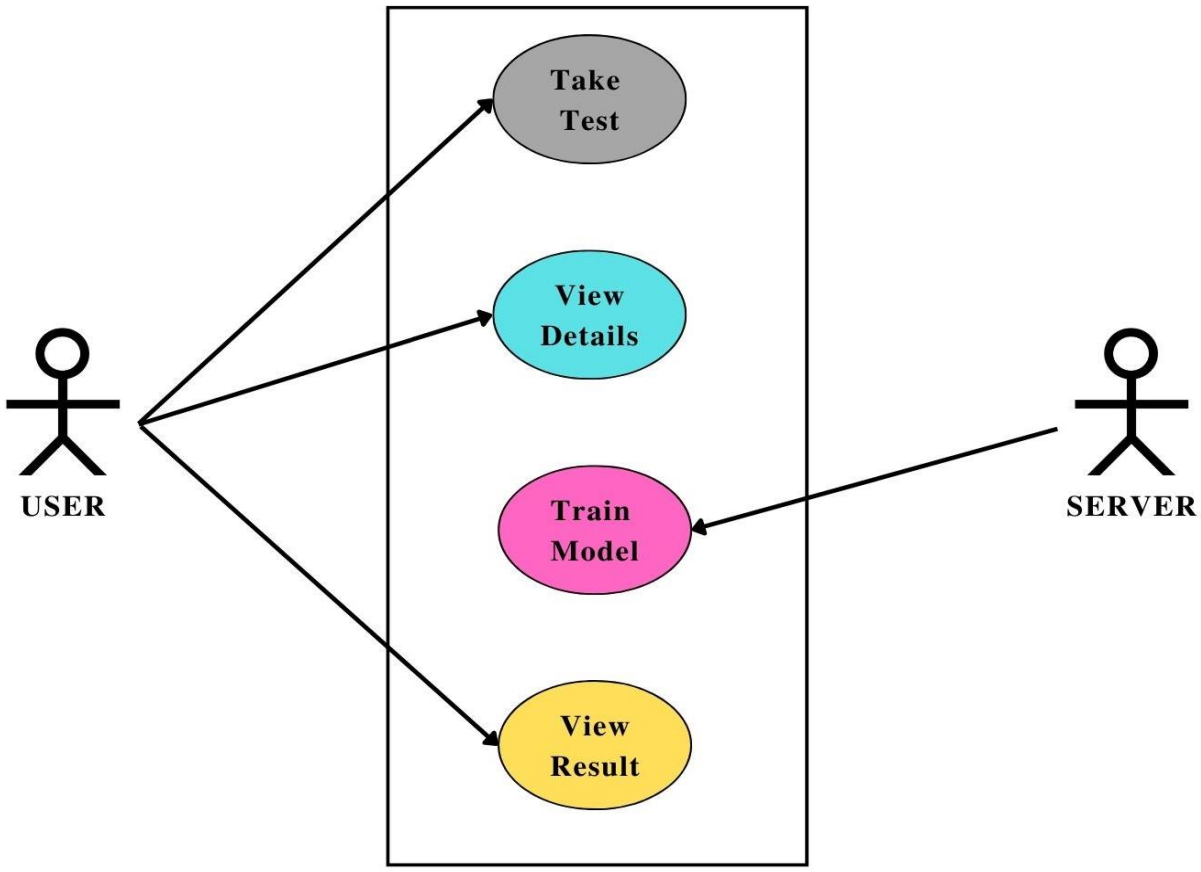

Figure 3.1 (Use Case Diagram)

3.3.2 Use Case Description

1. View Disease Details –

- **Brief Description** – This use case describes how users can view the details of diseases.
- **Flow of Events –**

  * When a user comes to the home page of the web application there are options to select disease.

  * After selecting a particular disease users have to click on the 'Read Details' button.

  * Then user navigates to a page where details of the disease are shown.

- **Pre-Conditions –** None
- **Post-Conditions –** If the use case is successful the user can view the details of the disease.

2. Take Test –

- **Brief Description** – This use case describes how users can take the test for disease prediction.
- **Flow of Events –**

  * When a user comes to the home page of the web application there are options to select disease

  * After selecting a particular disease users have to click on the 'Take Test' button.

  * Then user navigates to a page where he has to fill out a form with appropriate details to take the test.
- **Pre-Conditions –** None
- **Post-Conditions –** If the use case is successful the user can view the details of the disease.

3. View Result –

- **Brief Description** – This use case describes how users can view the results of the test for disease prediction.
- **Flow of Events –**
  * When a user has filled the form for all the required details to take the test then he has to click on the ‘View Result’ button.
  * Then the application will be navigated to a new page where the result of the prediction is shown
- **Pre-Conditions –** None
- **Post-Conditions –** If the use case is successful, the user can view the test result of disease prediction.

3.4 Design Diagrams

3.4.1 Architecture Diagram

The first step is to the dataset for heart disease, diabetes disease, and breast cancer we have imported the UCI dataset, PIMA dataset, and Indian liver dataset respectively. Once we have imported the dataset then visualization of each input takes place. After visualization pre-processing of data takes place where we check for outliers, and missing values and also scale the dataset then on the updated dataset we split the data into training and testing. Next is the training dataset we applied KNN, logistic regressor, and random forest algorithm and applied knowledge on the classified algorithm using the testing dataset. After applying the knowledge

we will choose the algorithm with the best accuracy for each of disease. Then we build a pickle file for the disease and then integrated the pickle file with the Flask for the output of the model on the webpage.

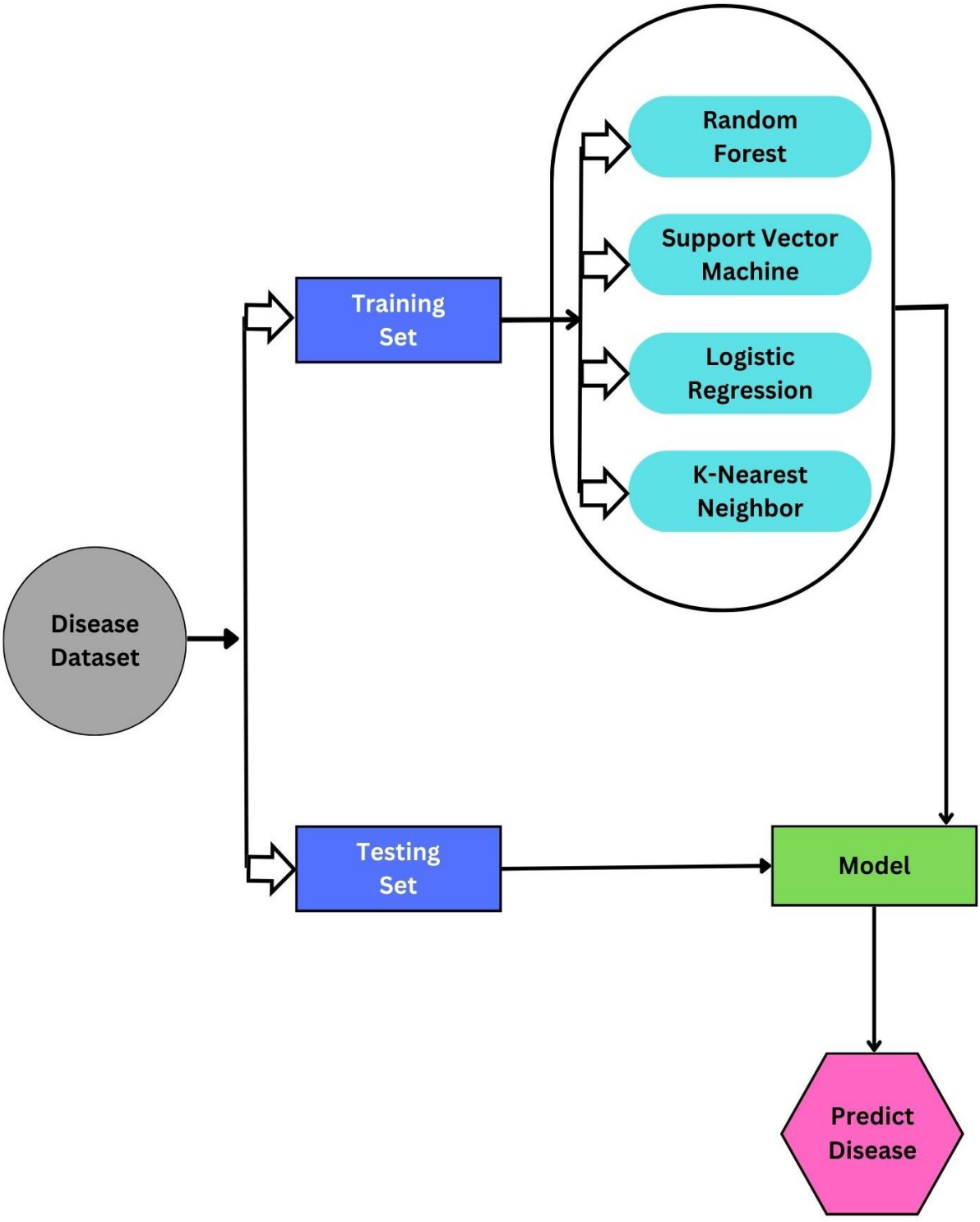

Figure 3.2(Architecture Diagram)

3.4.2 Sequence Diagram

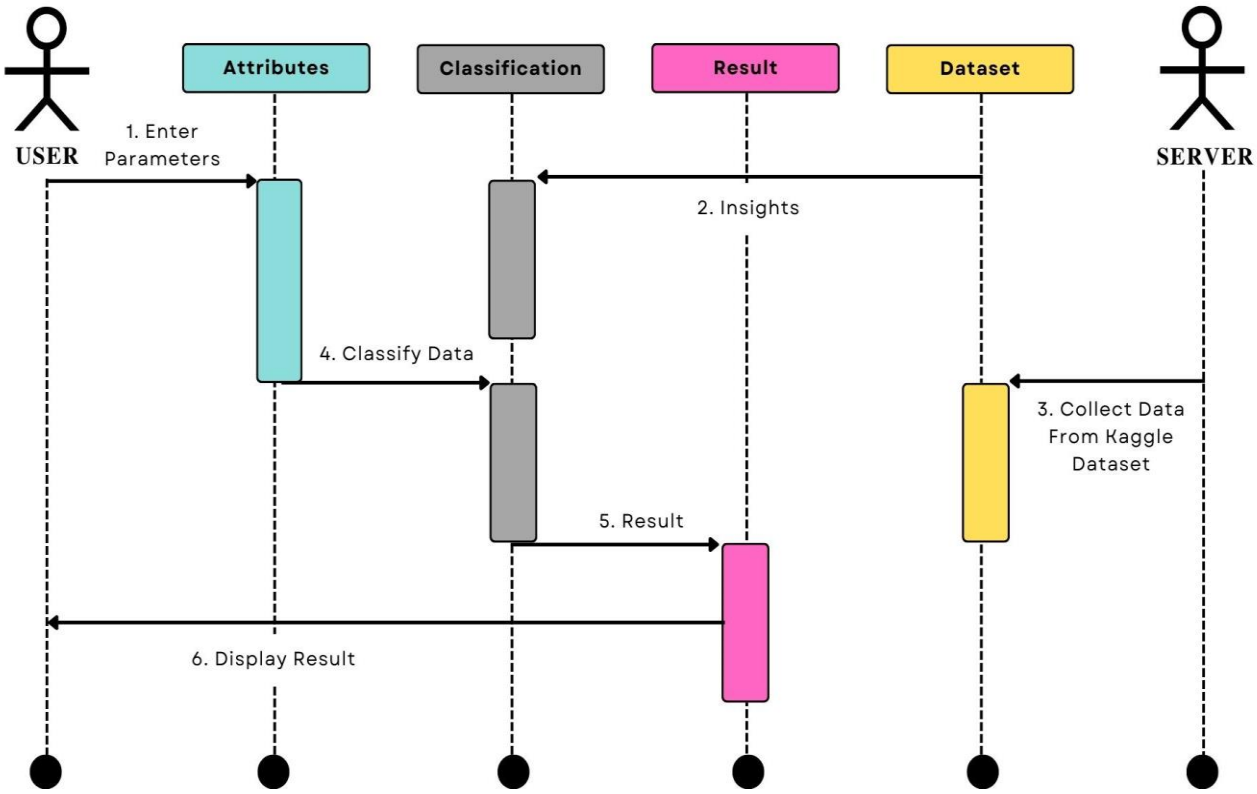

Figure 3.3 (Sequence Diagram)

Chapter – 4

Implementation and Testing

Chapter – 4
Implementation and Testing

4.1 Class diagram

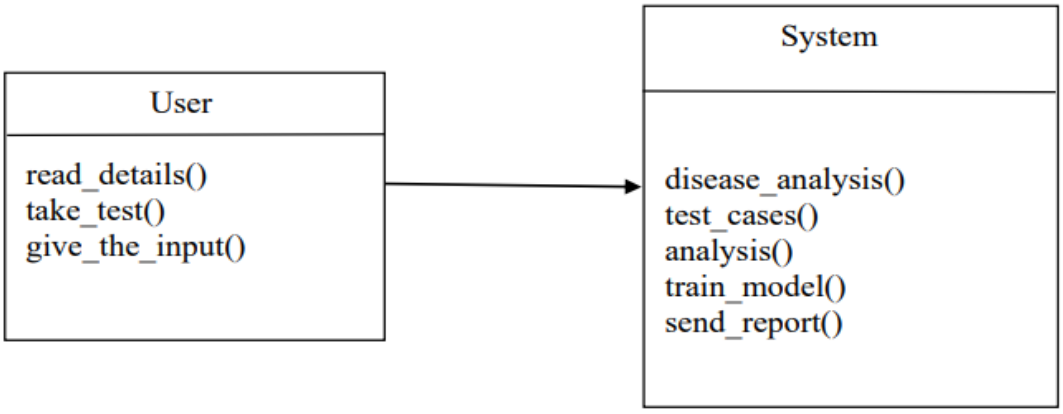

Figure-4.1(Class Diagram)

User- When a user comes to the home page of the web application there are options to select disease. After selecting a particular disease users have to click on the 'Read Details ' button. Then user navigates to a page where details of the disease are shown. When a user comes to the home page of the web application there are options to select disease. After selecting a particular disease users have to click on the 'Take Test' button. Then user navigates to a page where he has to fill out a form with appropriate details to take the test.

System - This system is used by end-users i.e. patients/any user. In this system, the user will enter all the details in the form of the disease he wants to predict. These details then will be given to the machine learning model to predict the disease. Algorithms are then applied to which gives the best accuracy. Then System will predict disease on the basis of details. This system uses Machine Learning Technology. Naïve Bayes algorithm is used for predicting the disease by using details, for classification KNN algorithm is used, Logistic regression is used

for extracting features which are having most impact value, and the Decision tree is used to divide the big User read_details() take_test() give_the_input() System disease_analysis() test_cases() analysis() train_model() send_report() 10 dataset into smaller parts. The final output of this system will be the disease predicted by the model along with the precautions.

4.2 Test

Test Case 1: A test case for empty fields in the form.
Expected Output: Prompt for the request to fill input value.
System Output: Please fill out this field.
Verdict: Passed

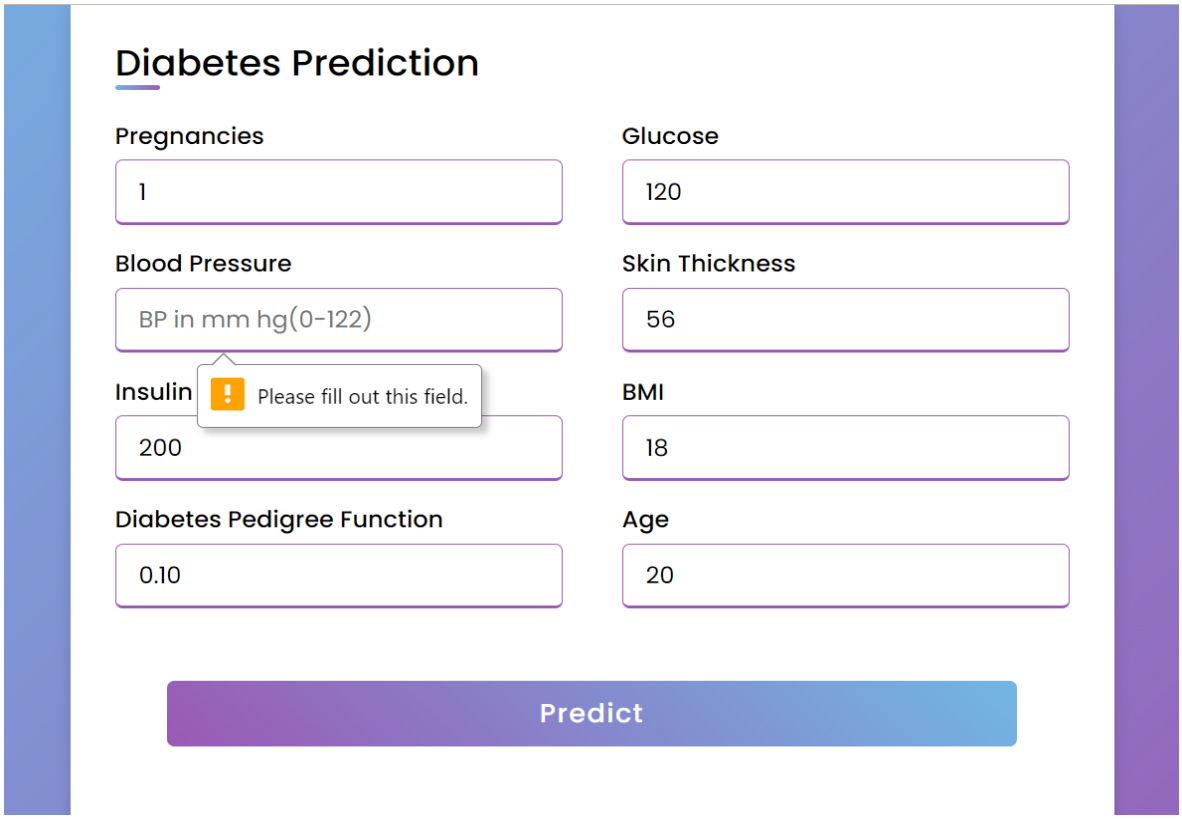

Figure-4.2(Test Case 1)

Test Case 2: A test case for entering an invalid data type value.

Expected Output: Prompt for filling in appropriate values.

System Output: Please fill appropriate value according to the given data type.

Verdict: Passed

INPUT: String value type in Glucose Field instead of integer

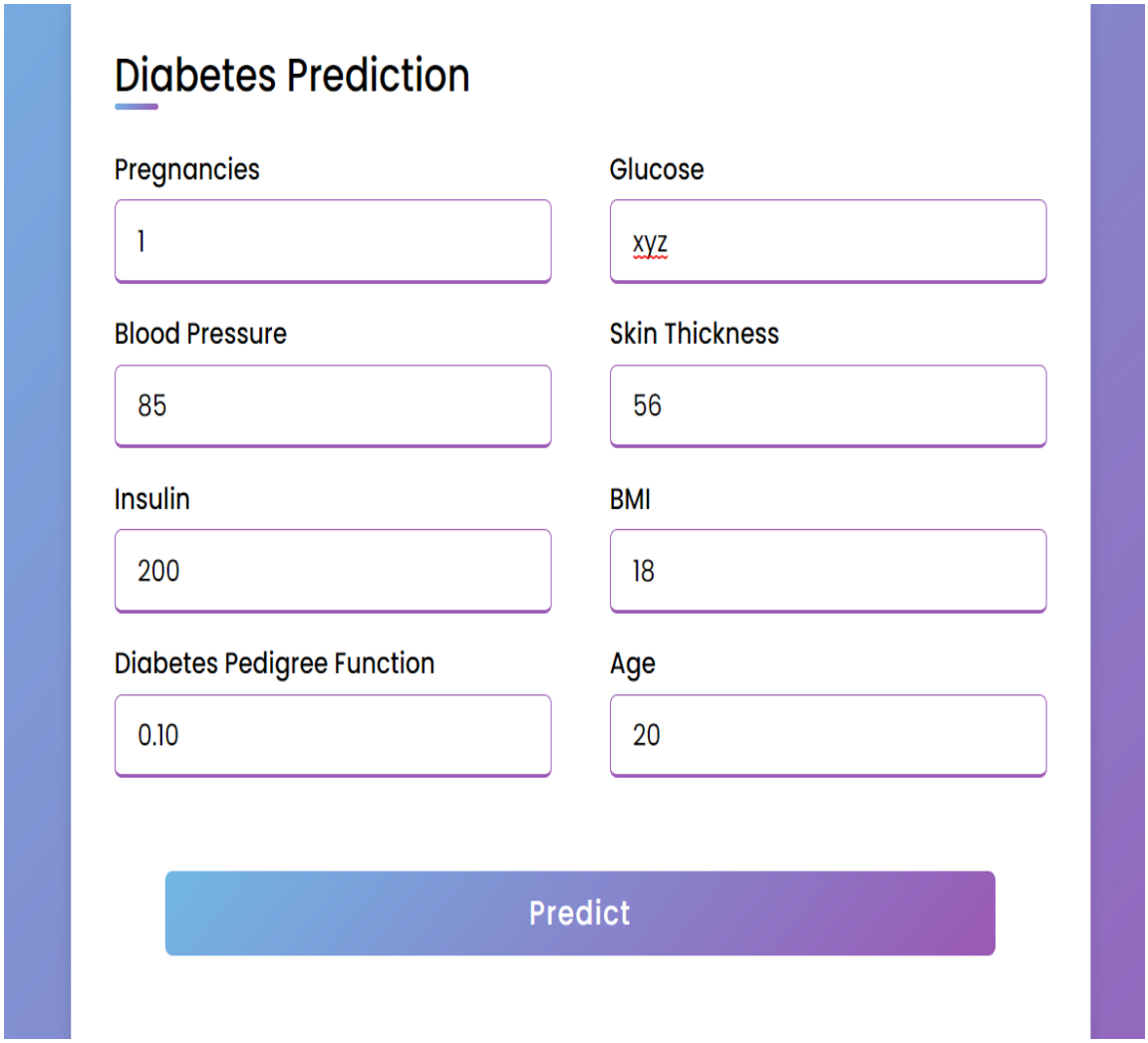

Figure-4.3(Test case 2 - Input)

OUTPUT : Prompt to fill appropriate details.

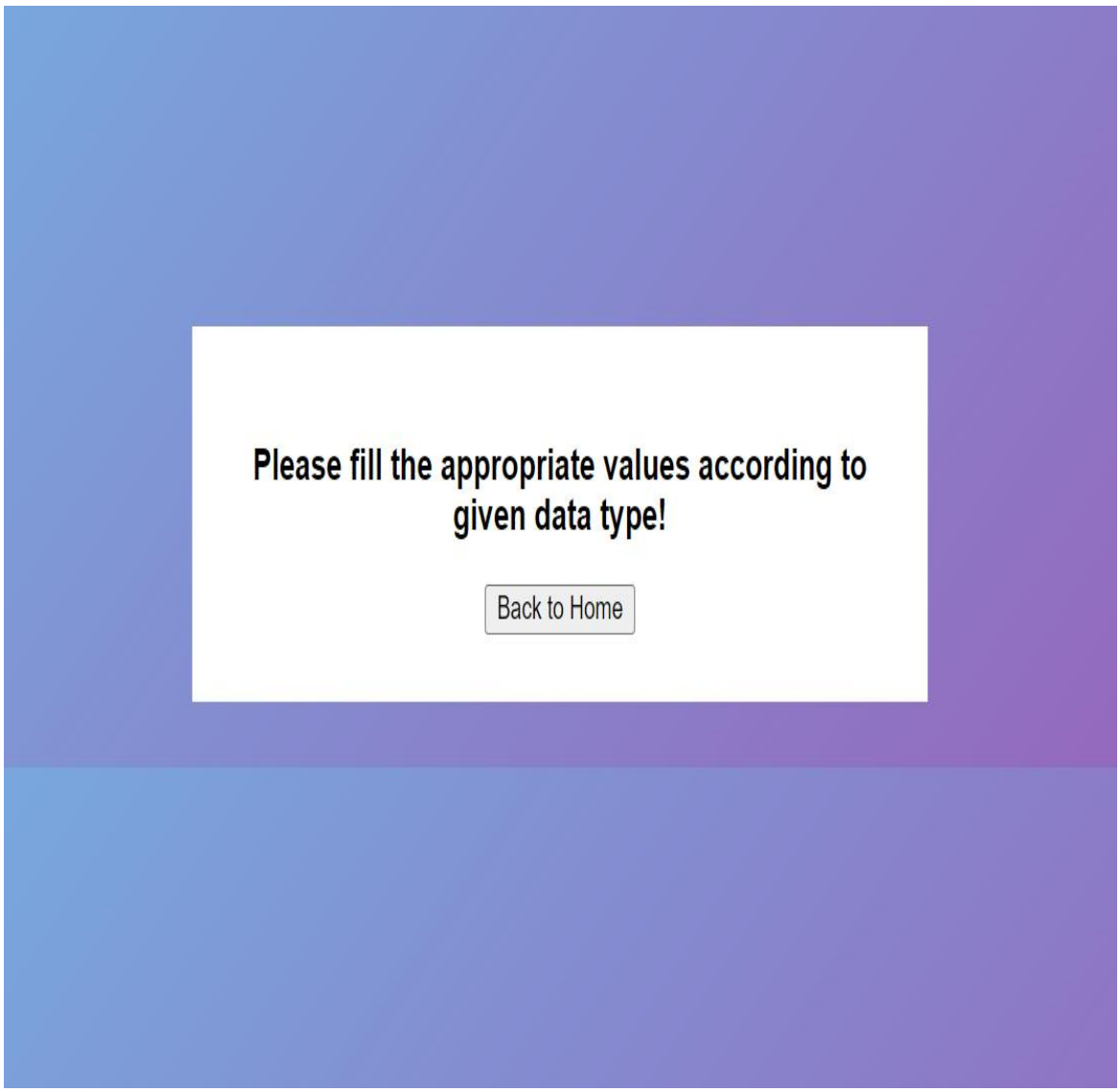

Figure-4.4(Test case 2 - Output)

Test Case 3: A test case for input values outside of mentioned range.

Expected Output: Prompt for filling in values in the range.

System Output: Please fill out the values in the given range.

Verdict: Passed

Range:

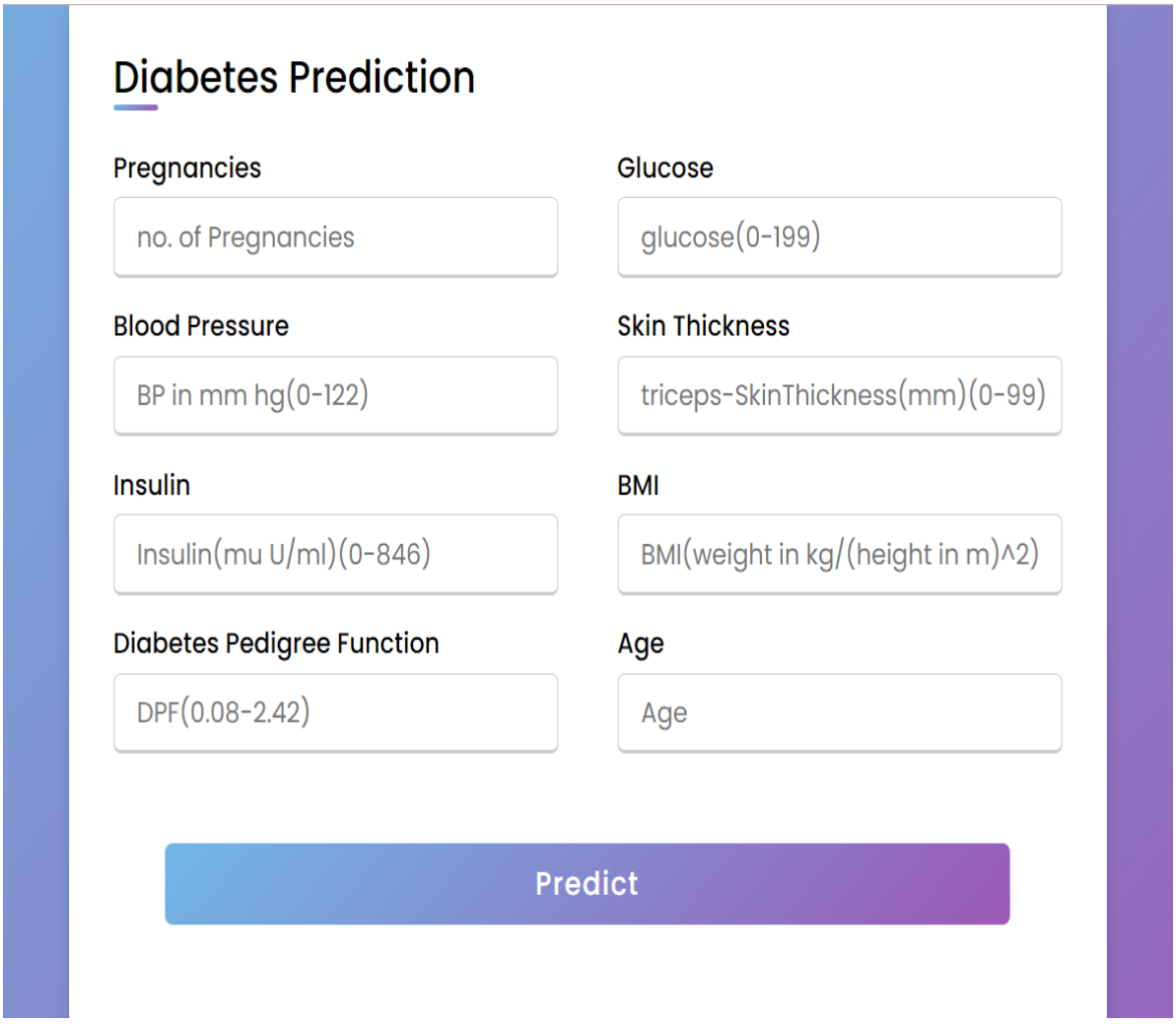

Figure-4.5(Test case 3 - Range)

Input: Blood Pressure out of given range

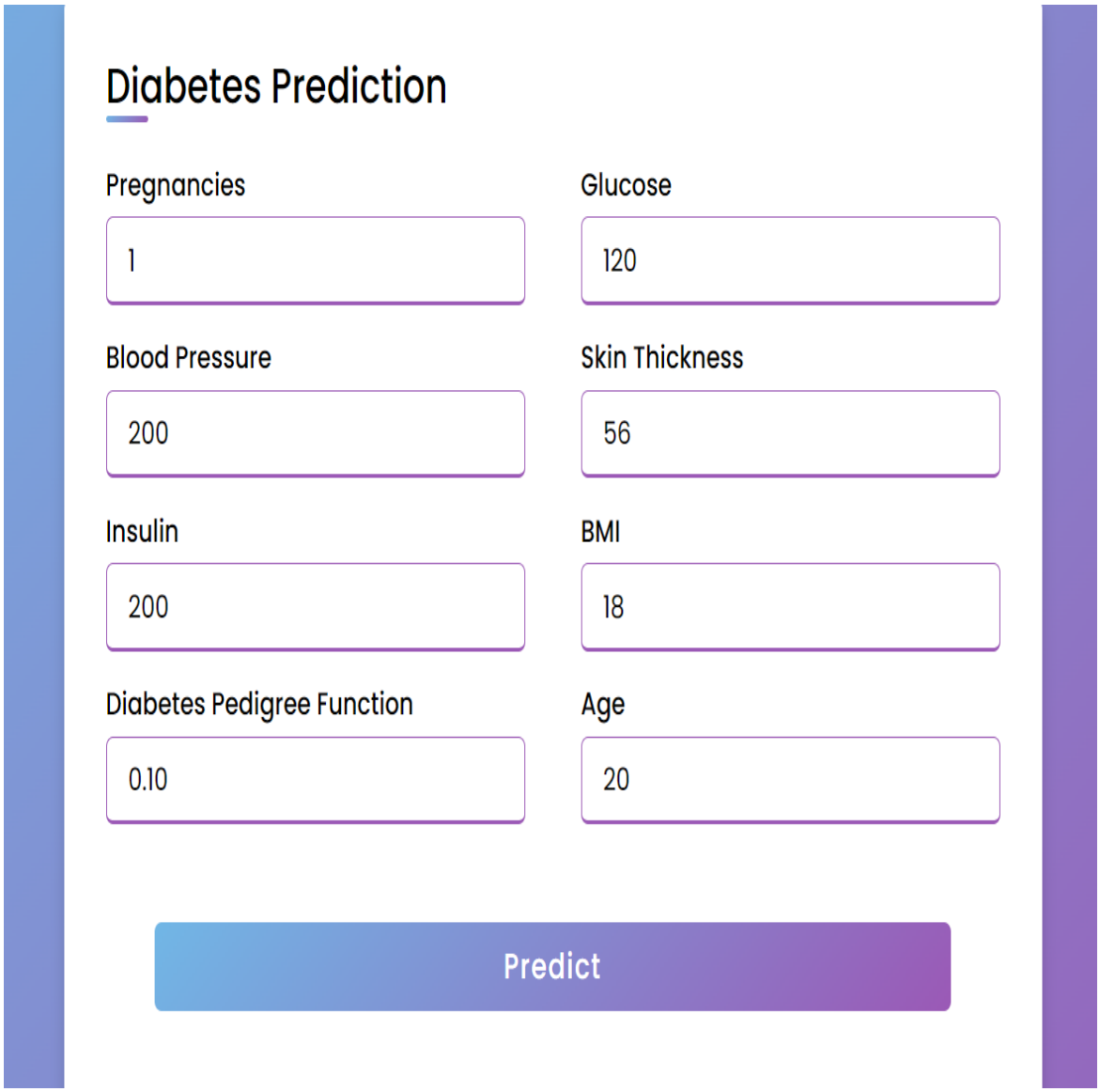

Figure-4.6(Test case 3 - Input)

Output: Prompt to fill out values in the range.

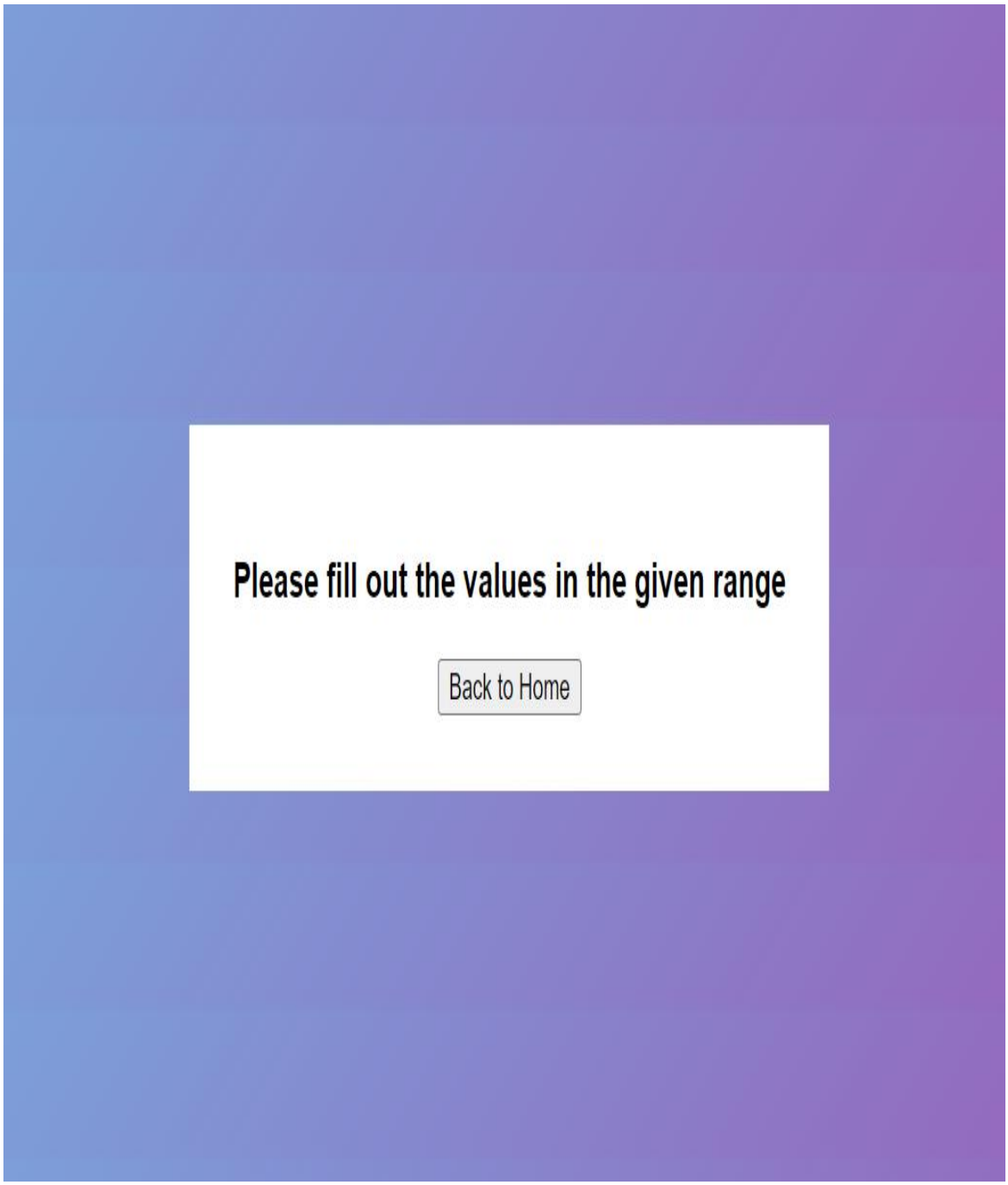

Figure-4.7(Test case 3 - Output)

4.3 Implementation Snapshots

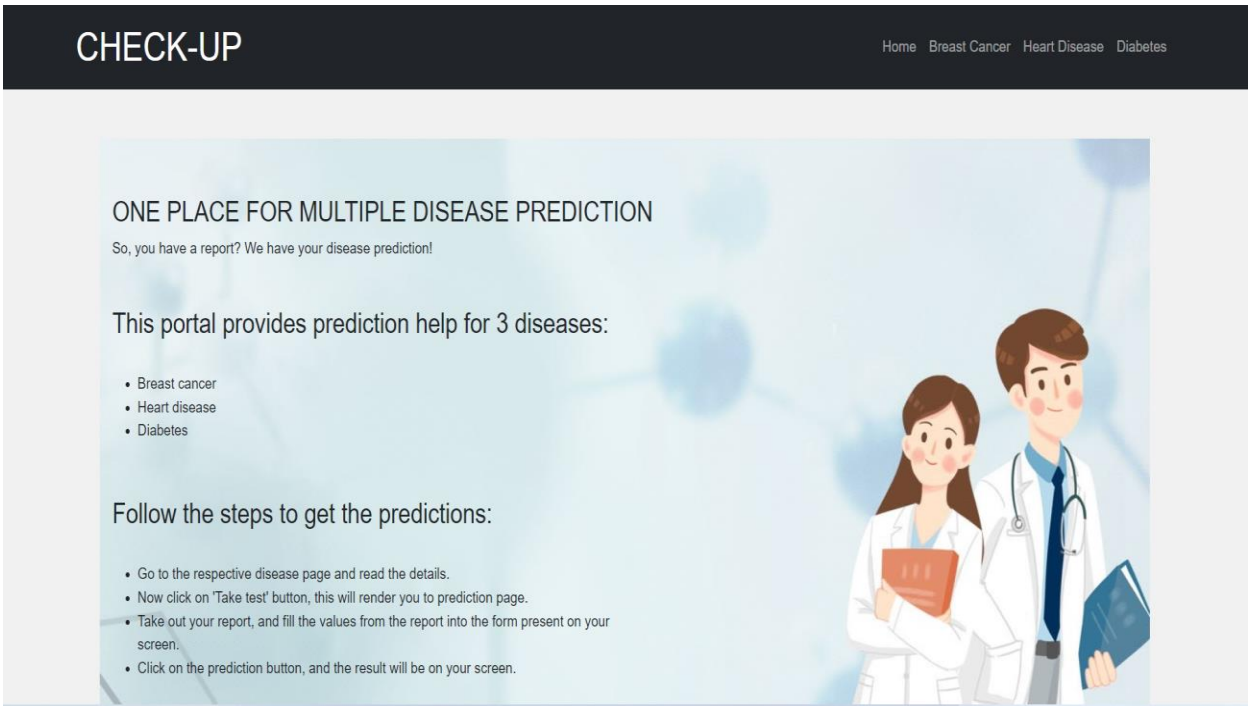

Figure 4.8 (Home Page-I)

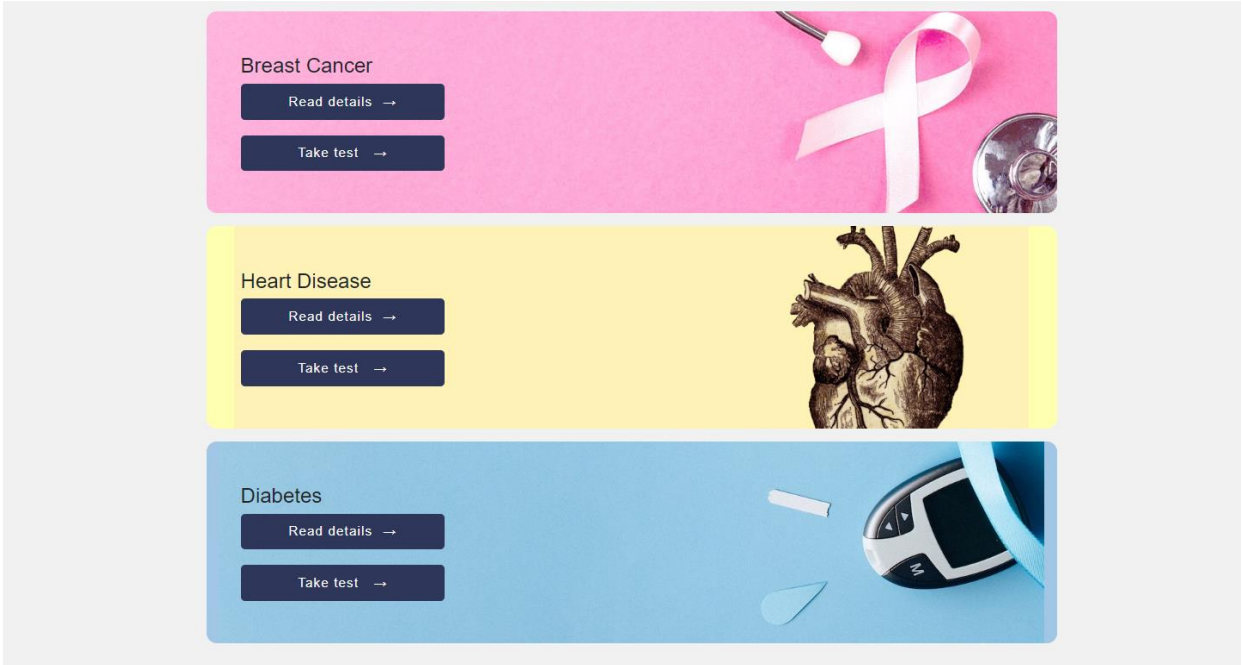

Figure 4.9 (Home Page-II)

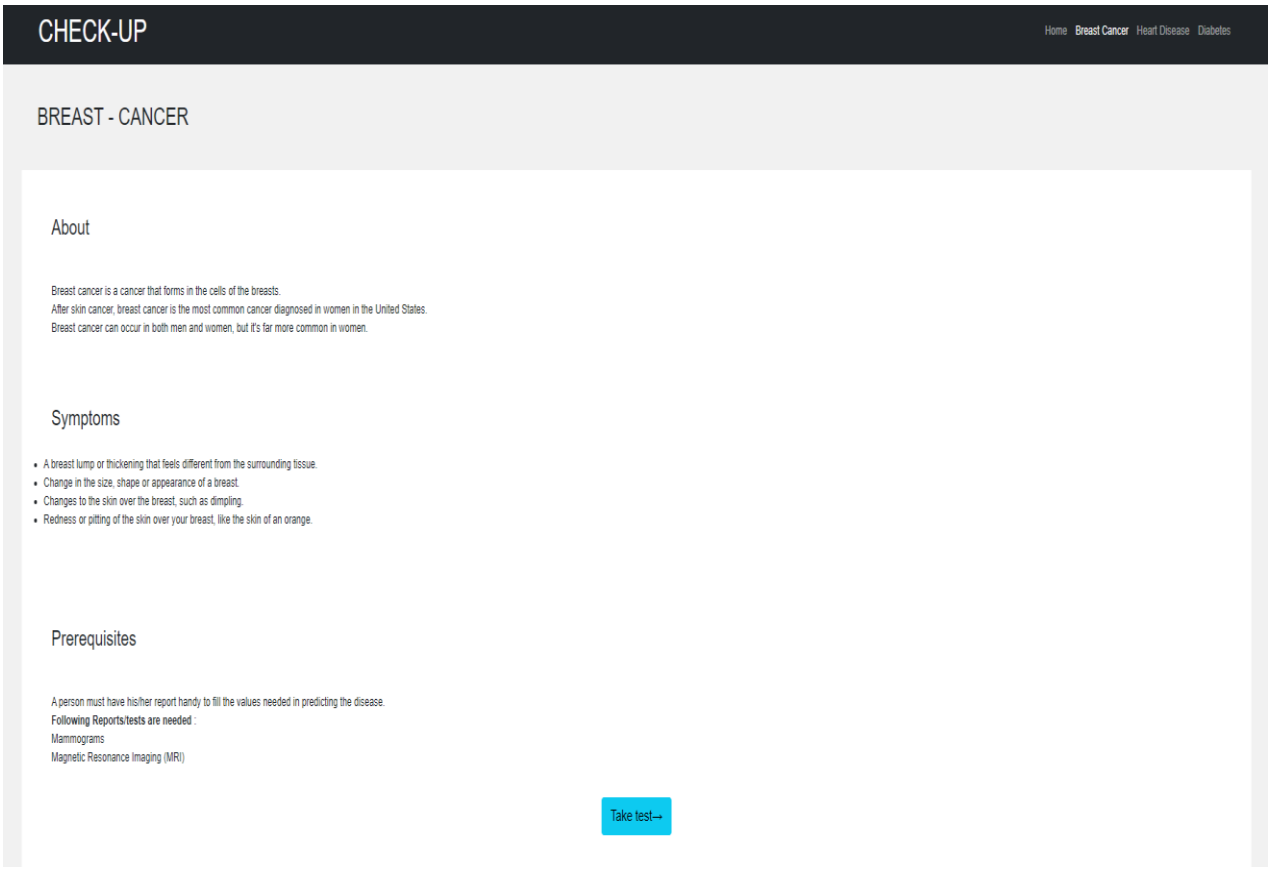

Figure 4.10 (Read Details – Breast Cancer)

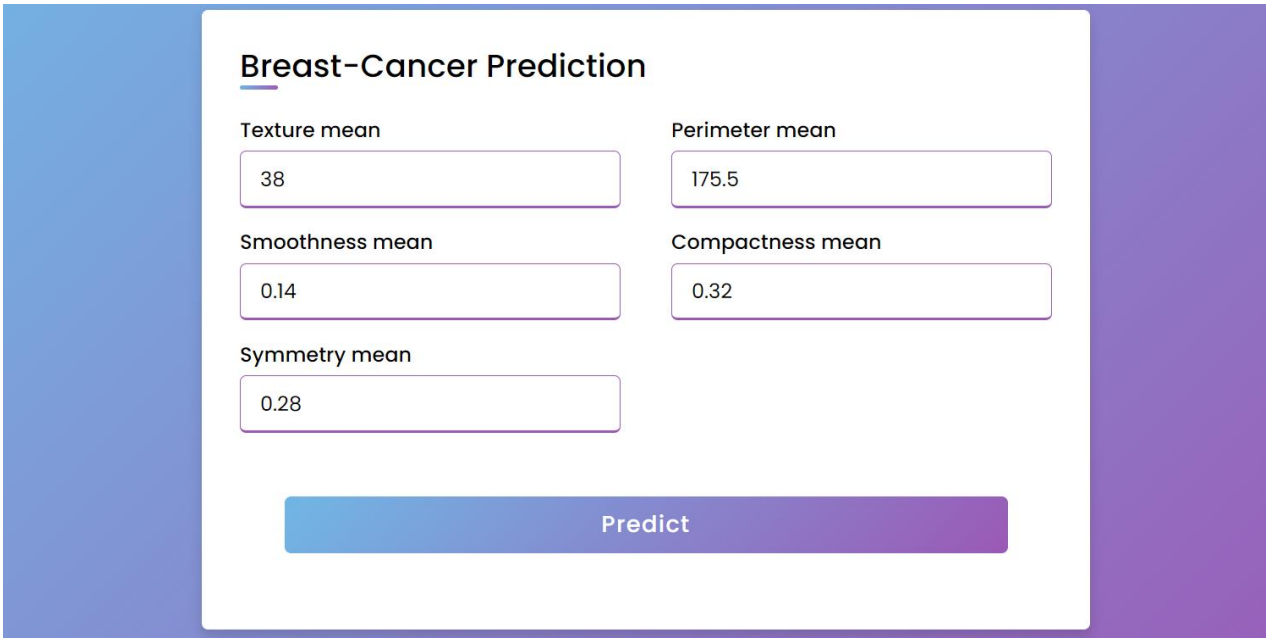

Figure 4.11 (Take test form – Breast Cancer)

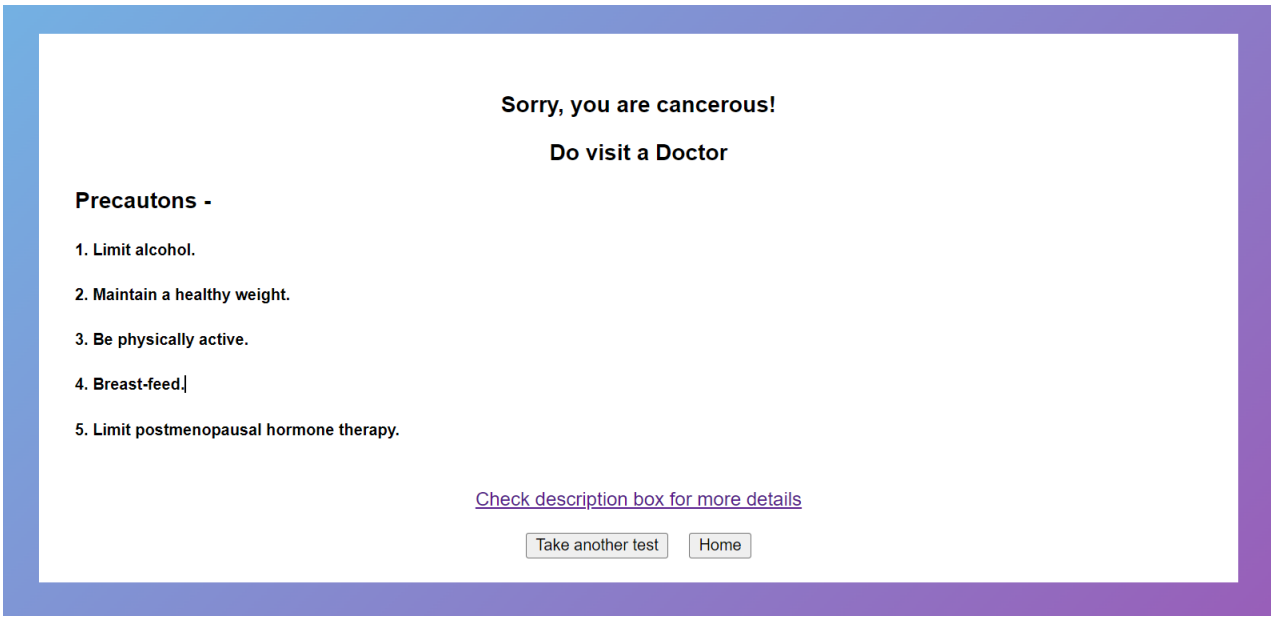

Figure 4.12 (Output – Breast Cancer)

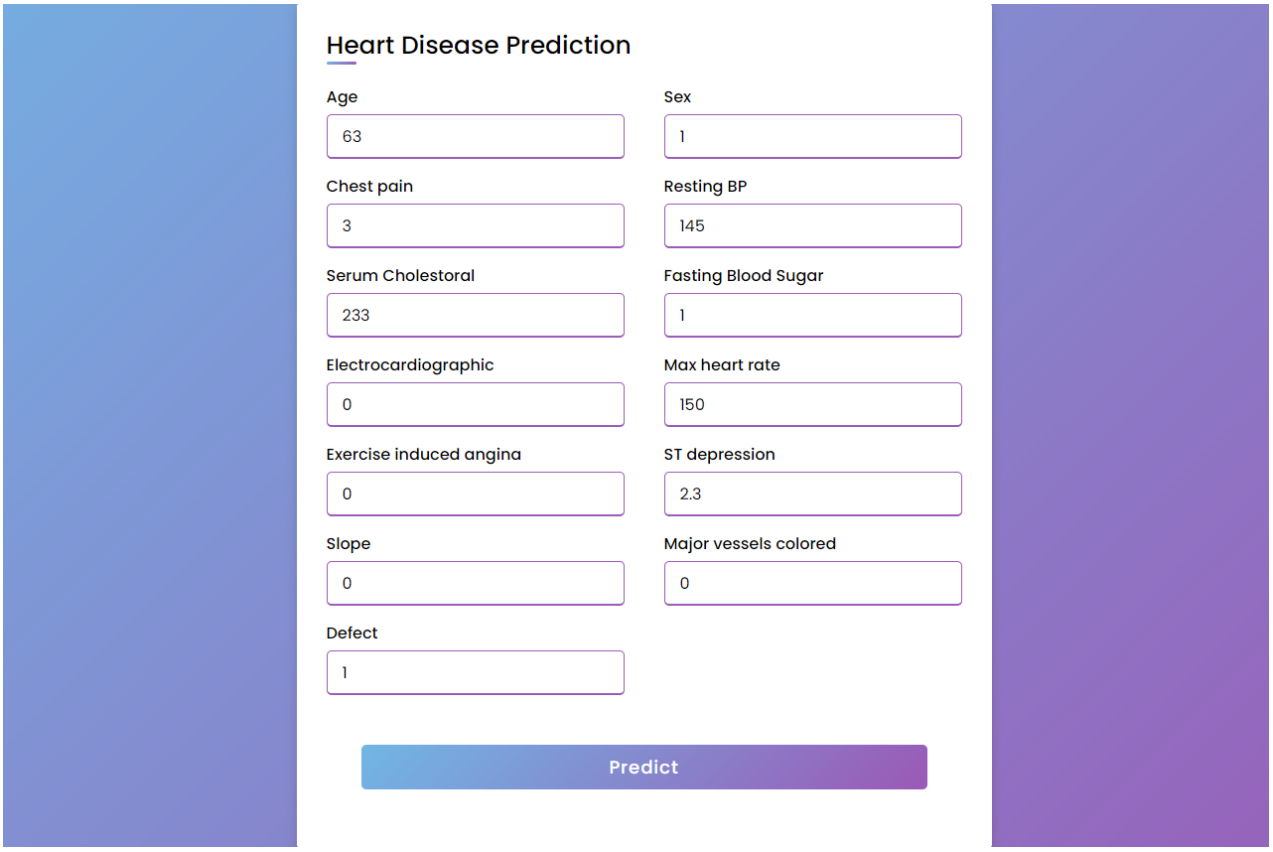

Figure 4.13 (Take Test form – Heart Disease)

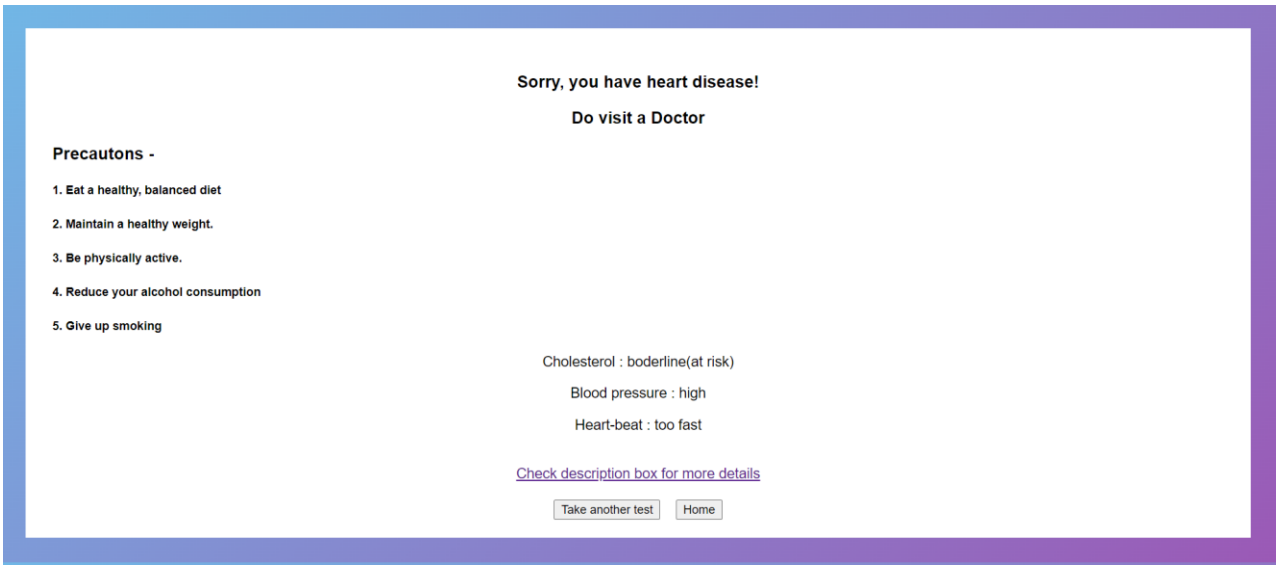

Figure 4.14 (Output – Heart Disease)

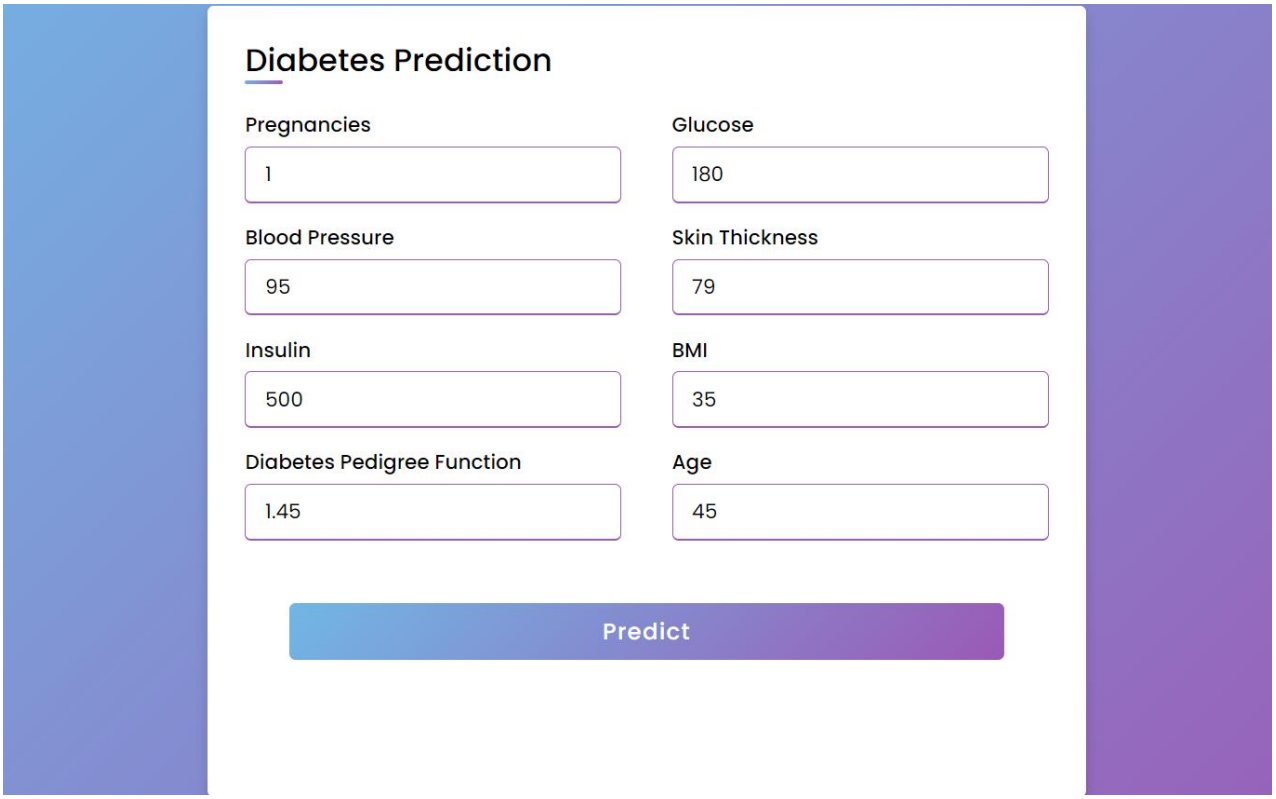

Figure 4.15 (Take Test form – Diabetes)

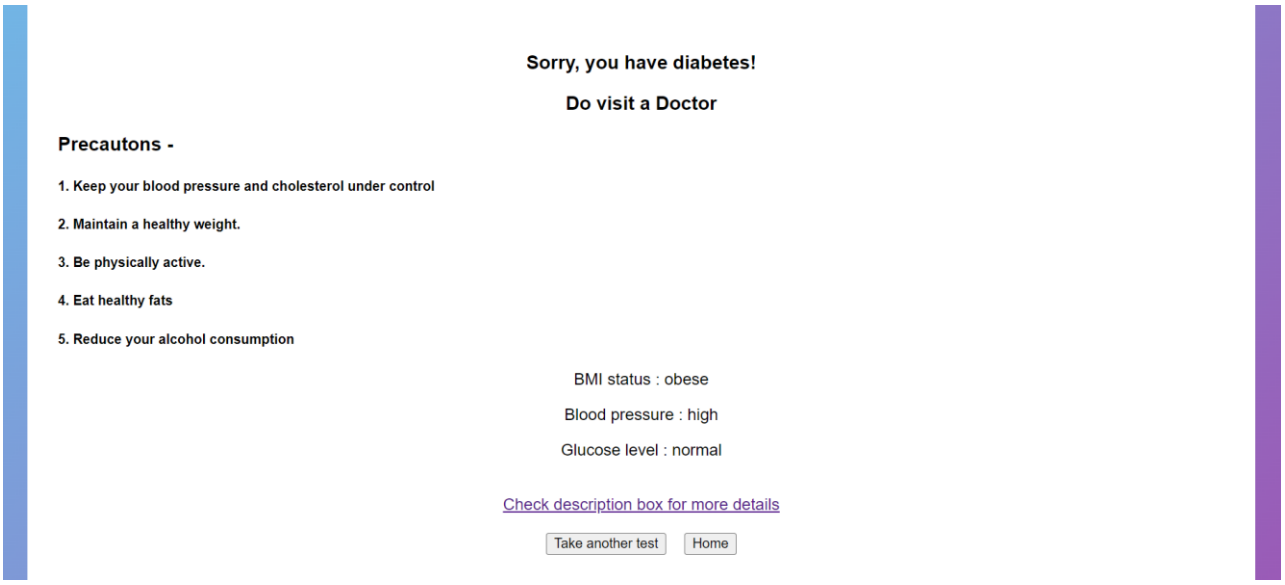

Figure 4.16 (Output – Diabetes)

Chapter- 5

Conclusion

Chapter – 5
Conclusion

The main objective of this project was to create a system that would predict more than one disease and do so with high accuracy. Because of this project the user doesn't need to traverse different websites which saves time as well. Diseases if predicted early can increase your life expectancy as well as save you from financial troubles. As the system is based on the web application, the user can use this system from anywhere and at any time. In conclusion, for disease risk modeling, the accuracy of risk prediction depends on the diversity feature of the hospital data. This systematic review aims to determine the performance, limitations, and future use of Software in health care. Findings may help inform future developers of Disease Predictability Software and promote personalized patient care. The program predicts Patient Diseases. Disease Prediction is done through User Symbols. For this purpose, we have used various machine learning algorithms like Random Forest, and K nearest neighbor (KNN) to achieve maximum accuracy.

Future Scope

• In the future, we can add more diseases to the existing API.

• We can try to improve the accuracy of prediction in order to decrease the mortality rate

REFRENCES

[1] https://www.kaggle.com/datasets/uciml/breast-cancer-wisconsin-data

[2] K.P. Bennett and O.L. Mangasarian: "Robust Linear Programming Discrimination of Two Linearly Inseparable Sets", Optimization Methods and Software 1, 1992, 23-34.

[3] https://archive.ics.uci.edu/ml/datasets/Breast+Cancer+Wisconsin+%28Diagnostic%29

[4]https://www.kaggle.com/code/tentotheminus9/what-causes-heart-disease-explaining-the-model

[5] https://www.kaggle.com/datasets/akshaydattatraykhare/diabetes-dataset

[6] R. Tina Patil, S.S. Sherekar, Performance analysis of Naive bayes and J48 classification algorithm for data classification, Int. J. Comput. Sci. Appl. 6 (2) (2013) 256–261.

[7] Shruti Ratnakar, K. Rajeswari, Rose Jacob, Prediction of heart disease using genetic algorithm for selection of optimal reduced set of attributes, Int. J. Adv. Comput. Eng. Netw. 1 (2) (2013) 51–55.

[8] S. Grampurohit, C. Sagarnal, Disease prediction using machine learning algorithms, 2020 https://doi. org/10.1109/INCET49848.2020.9154130.

[9] R.J.P. Princy, S. Parthasarathy, P.S. Hency Jose, A. Raj Lakshminarayanan, S. Jeganathan, Prediction of Cardiac Disease using Supervised Machine Learning Algorithms, in: 2020 4th International Conference on Intelligent Computing and Control Systems (ICICCS), 2020, pp. 570– 575, https://doi.org/10.1109/ ICICCS48265.2020.9121169.

[10] P. Deepika, S. Sasikala. Enhanced Model for Prediction and Classification of Cardiovascular Disease using Decision Tree with Particle Swarm Optimization, 2020 4th International Conference on Electronics, Communication and Aerospace Technology (ICECA), 2020, pp. 1068-1072, doi: 10.1109/ ICECA49313.2020.9297398.

Appendix A: Glossary

● Application: a software or program which is designed for use by a user for specific operations.

● User Interface (UI): anything that the user of an application may see, interact with, or experience

● Web Application: A computer program that utilizes web browsers and web technology to perform tasks over the Internet

● Machine Learning (ML): Machine learning (ML) is a type of artificial intelligence (AI) that allows software applications to become more accurate at predicting outcomes without being explicitly programmed to do so. Machine learning algorithms use historical data as input to predict new output values.