

DIABETIC DISEASE PREDICTION

Submitted in partial fulfillment of the
requirements for the award of
Bachelor of Engineering degree in Computer Science and Engineering

by

**MATHAM SAICHARAN(38110481)
RUDRARAJU SAI NIRANJAN DATTA
VARMA(38110469)**



**DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING
SCHOOL OF COMPUTING**

SATHYABAMA

**INSTITUTE OF SCIENCE AND TECHNOLOGY
(DEEMED TO BE UNIVERSITY)
Accredited with Grade "A" by NAAC**

**JEPPIAAR NAGAR, RAJIV GANDHI SALAI,
CHENNAI - 600 119**

MARCH – 2022



SATHYABAMA
INSTITUTE OF SCIENCE AND TECHNOLOGY
(DEEMED TO BE UNIVERSITY)



Accredited with "A" grade by NAAC

Jeppiaar Nagar, Rajiv Gandhi Salai, Chennai - 600119

www.sathyabama.ac.in

DEPARTMENT OF COMPUTER SCIENCE ENGINEERING

BONAFIDE CERTIFICATE

This is to certify that this Project Report is the bonafide work of **MATHAM SAICHARAN (38110481)** and **RUDRARAJU SAI NIRANJAN DATTA VARMA (38110469)** who carried out the project entitled "**DIABETIC DISEASE PREDICTION** " under my supervision from November 2021 to March 2022.

Internal Guide

Dr. R. Rajalakshmi, ME., PhD.,

Head of the Department

Dr S. VIGNESHWARI, M.E, PH.D

Dr.L.Lakshmanan M.E., Ph.D. ,

Submitted for Viva voce Examination held on _____

DECLARATION

I hereby declare that the work described in this thesis, entitled **“DIABETIC DISEASE PREDICTION”** which is being submitted by me in partial fulfilment for the VII-Semester of the degree of Bachelor of Technology in the Department of **Computer Science and Engineering** to the National Institute of Technology Raipur is the result of investigations carried out by me under the guidance of **Dr.Dilip Singh Sisodia, Assistant Professor, Computer Science & Engineering Department.**

The work is original and has not been submitted for any Degree/Diploma of this or any other Institute/university.

Signature

Name of the Candidate
MATHAM SAICHARAN(38110481)
RUDRARAJU SAI NIRANJAN
DATTA VARMA(38110469)

Place:CHENNAI Date:

ACKNOWLEDGEMENT

I am pleased to acknowledge my sincere thanks to **Board of Management** of **SATHYABAMA** for their kind encouragement in doing this project and for completing it successfully. I am grateful to them.

I convey my thanks to **Dr. T.Sasikala M.E.,Ph.D., Dean**, School of Computing **Dr.S.Vigneshwari M.E., Ph.D.** and **Dr.L.Lakshmanan M.E., Ph.D.** , Heads of the Department of Computer Science and Engineering for providing us necessary support and details at the right time during the progressive reviews.

I would like to express my sincere and a deep sense of gratitude to my Project Guide **Mr.Murari Devakannan Kamalesh** for her valuable guidance, suggestions and constant encouragement paved way for the successful completion of my project work. I wish to express my thanks to all Teaching and Non-teaching staff members of the **Department of Computer Science and Engineering** who were helpful in many ways for the completion of the project

ABSTRACT

- Nowadays, eHealth service has become a booming area, which refers to computer-based health care and information delivery to improve health service locally, regionally and worldwide.
- An effective disease risk prediction model by analyzing electronic health data benefits not only to care a patient but also to provide services through the corresponding data-driven eHealth systems.
- In this paper, we particularly focus on predicting and analyzing diabetes mellitus, an increasingly prevalent chronic disease that refers to a group of metabolic disorders characterized by a high blood sugar level over a prolonged period of time.
- K-Nearest Neighbor (KNN) is one of the most popular and simplest machine learning techniques to build such a disease risk prediction model utilizing relevant health data.
- In order to achieve our goal, we present an optimal K-Nearest Neighbor (OPT-KNN) learning based prediction model based on patient's habitual attributes in various dimensions.
- This approach determines the optimal number of neighbors with low error rate for providing better prediction outcome in the resultant model.
- The effectiveness of this machine learning eHealth model is examined by conducting experiments on the real-world diabetes mellitus data collected from medical hospitals.

CONTENT

DESCRIPTION	PAGE NO.
MINOR PROJECT REPORT	1
DECLARATION	2
ACKNOWLEDGEMENT	3
ABSTRACT	4
CONTENT	5
LIST OF FIGURES	8
LIST OF TABLES	8
CHAPTER-1	9
1. INTRODUCTION	9
1.1 Overview.....	9
1.2 Objective.....	10
1.3 Scope and Motivation.....	10
CHAPTER-2.....	11
2. LITERATURE SURVEY.....	11
2.1 prediction of diabetic using classification Algorithm.....	12
2.2 A Novel Technique to predict Diabetic using Data mining.....	12
2.3 Review on prediction of diabetes Data mining Technique.....	13
2.4 prediction Technique in data mining for diabetes Mellitus.....	13
2.5 Utilization of data mining for diagnosis os diabetes.....	14
CHAPTER-3.....	15
3 . METHODOLOGY.....	15
3.1 Aim of the project.....	16
3.2 Existing system.....	16
3.2.1 Disadvantages of Existing system.....	16
3.3 Proposed System.....	16
3.3.1 Advantages of Proposed System.....	17
3.4 System Architecture.....	17
3.5 System Requirements.....	18
3.5.1 Hardware Requirements.....	18

3.5.2 Software Requirements.....	18
3.6 Modules Description.....	18
3.6.1 Data Constraints.....	18
3.6.2 Train Dataset and Test Dataset.....	18
3.6.3 Pre-processing of data.....	19
3.6.4 Feature Extraction.....	19
3.6.5 ML Algorithm : KNN.....	19
3.7 Data flow Diagram.....	19
3.7.1 Data flow diagrams.....	20
3.7.2.1 DFD-Level-0.....	20
3.7.2.2 DFD-Level-1.....	20
3.7.2.3 DFD-Level-2.....	20
3.7.3 Class Diagram.....	21
3.8 Software Environment.....	21
3.8.1 Python.....	21
3.8.1.1 History of python.....	22
3.8.1.2 Python Features.....	22
3.8.1.3 Getting Python.....	23
3.8.2 Interactive Mode Programming.....	24
3.8.2.1 Script Mode Programming.....	25
3.8.3 Flask Framework.....	25
CHAPTER-4.....	30
4. RESULT AND PERFORMANCE EVALUATON.....	30
4.1 Result	31
4.2 Requirements Analysis.....	31
4.3 Functional Requirements.....	31
4.3.1 Usability.....	31
4.3.2 Robustness.....	31
4.3.3 Security.....	31
4.3.4 Reliability.....	31
4.3.5 Compatibility.....	32

4.3.6 Flexibility.....	32
4.3.7 Safety.....	32
4.4 Non-Functional Requirements.....	32
4.4.1 Portability.....	32
4.4.2 Performance.....	32
4.4.3 Accuracy.....	32
4.4.4 Maintainability.....	32
CHAPTER-5.....	33
5. TESTING.....	33
5.1 Input Design.....	34
5.2 Output Design.....	34
5.3 Feasibility.....	35
5.3.1 Feasibility Study.....	35
5.3.2 Economical Feasibility.....	35
5.3.3 Social Feasibility.....	36
5.3.4 System Testing.....	36
5.4 Types of Tests.....	36
5.4.1 Unit testing.....	36
5.4.2 Integration testing.....	37
5.4.3 Functional test.....	37
5.4.4 System Test.....	38
5.4.5 White Box Testing.....	38
5.4.6 Blacking Box Testing.....	38
5.5 Unit Testing.....	38
5.6 Integration Testing.....	39
CHAPTER-6.....	40
6. CONCLUSION.....	40
7.REFERENCES.....	42

FIGURE NAMES

FIGURE N0.	FIGURE NAMES	PAGE N0.
3.4	System Architecture	17
3.7.2.1	DFD-Level-0	20
3.7.2.2	DFD_Level-1	20
3.7.2.3	DFD-Level-2	21
3.7.3	Class Diagram	21
3.8.2	Interactive mode Programming	24
3.8.3	Methods and Description	26
3.8.3.1	Login Html	28
3.8.3.2	Success Url	29

CHAPTER-1

INTRODUCTION

1.1 OVERVIEW

Diabetes may be a set of metabolic problems known by high blood sugar levels over a protracted period of our time. Diabetes is outlined as a bunch of metabolic disorders in the main caused by abnormal insulin secretion and/or action. Symptoms of high aldohexose incorporate excessive voiding, continually feeling thirsty and enlarged hunger. If not treated on time, diabetes will cause serious health problems in a person like diabetic acidosis, hyperosmolar hyperglycemic state, or maybe result in death. This could result in time period complications as well as vas upset, brain stroke, failure, ulcers within the foot, and eye complications etc. Diabetes is caused once the duct gland within the body is unable to come up with insulin in enough amounts or once the cells and tissues within the body fail to utilize the insulin created. Diabetes exists in 3 forms: Diabetes Mellitus Type-1 is characterized by duct gland generating insulin but what's needed by the body, a condition conjointly referred to as "insulin-subordinated diabetes mellitus" (IDDM). Folks littered with type-1 DM need external insulin indefinite quantity to form up for the less insulin created by the duct gland. Diabetes Mellitus Type-2 is marked by the body resisting insulin because the body cells react otherwise to insulin than they traditional would. This could ultimately result in no insulin within the body. This can be otherwise referred to as "non-insulin subordinate diabetes mellitus" (NIDDM) or "adult beginning diabetes". This sort of diabetes is often found in folks with high BMI or people who lead associate degree inactive manner. Gestational diabetes is that the third principle structure that's ascertained throughout physiological state. Generally, for a traditional person, aldohexose levels vary from seventy to ninety-nine milligrams per deciliter. An individual is taken into account diabetic providing the fast aldohexose level is found to be over 126 mg/dL. Within the practice, an individual having an aldohexose concentration of a hundred to one hundred twenty-five mg/dL is taken into account as pre-diabetic. Such an individual is susceptible to the event of sort two diabetes. Over the years, it's been found that folks with the subsequent health characteristics face a larger risk against diabetes:

- A Body Mass Index worth larger than twenty five
- Members of the family littered with diabetes
- People with cholesterol concentration within the body but forty mg/dL prolonged high blood pressure having physiological condition diabetes
- People World Health Organization have suffered from polycystic ovary disorder within the past
- People happiness to ethnic teams like African American, or Native American, or Spanish American, or Asian-Pacific aged over forty five years
- Having associate degree inactive manner

1.2 OBJECTIVE

- Diabetes mellitus (DM) is one of the most prevalent chronic non-communicable diseases (NCD) around the world; about 90% of the patients who have diabetes suffer from Type 2 DM (T2DM)
- The risk of developing T2DM is strongly associated with many predispositions, behavioral, and environmental risk factors and also genetic factors

1.3 SCOPE/MOTIVATION

- About 90% of patients who have diabetes suffer from Type 2 DM (T2DM)
- Many studies suggest using the significant role of lncRNAs to improve the diagnosis of T2DM.
- Machine learning is the techniques are tools that can improve the analysis and interpretation or extraction of knowledge from the data.
- These techniques may enhance the prognosis and diagnosis associated with reducing diseases such as T2DM.

- We applied four classification models, including K-nearest neighbor (KNN)

CHAPTER-2

LITEARTURE SURVEY

2.1 Prediction of Diabetes using Classification Algorithms

Diabetes isn't a hereditary disorder however heterogeneous group of disorder which could ultimately result in an boom of glucose within the blood and lack of glucose inside the urine. Diabetes is typically resulting from genetics, way of life and surroundings. Eating an dangerous weight loss plan, being overweight play role in developing the diabetes. High blood sugar tiers can also result in kidney diseases, coronary heart illnesses. The excess of sugar in the blood can harm the tiny blood vessels in your frame. Signs of diabetes are blurry imaginative and prescient , extreme hunger, unusual weight reduction, common urination and thirsty. In this paper, parameters used within the facts set to locate the diabetes are Glucose, Blood pressure, pores and skin thickness, Insulin, Age. Huge volumes of statistics units are generated by health care industries. Those facts sets is a collection of patient information about the diabetes from the hospitals. Big records analytics is the processing which it examines the information units and exhibits the hidden information. Pima Indians Diabetes Database (PIDDD), this dataset is taken from the national Institute of Diabetes and Digestive diseases. The objective of the dataset is to predict whether or not the patient has diabetes or not, primarily based on diagnostic measurements in the dataset. Several constraints were taken from the massive database.

2.2 A Novel Technique To Predict Diabetic Disease Using Data Mining Classification Techniques

Data mining is the process of analyzing data from different perspectives and summarizing it into useful information. The software of data mining is an analytical tool for analyzing data. Data mining has become a main strategy in many industries to improve outputs and decrease costs. Now days in healthcare management this field will become very useful. Data mining techniques has became great potential for the healthcare industry to predict health deceases by using systematic data and analytics to identify inefficiencies and best practices that improve care and reduce costs. These techniques are fast in nature and take less time for the prediction

system to improve the diabetic decease with more accuracy. In this paper we are applying the various classification techniques over diabetic mellitus decease dataset for the prediction of decease and non decease person. The diabetic database is preprocessed to make the mining process more efficient. The preprocessed data is used to predict using classification algorithms like Discriminant analysis, KNN, Naïve Bayes and Support vector machine. These classifiers can be efficiently used in bioinformatics problem. We are analyzing the various classification techniques like Discriminant analysis, KNN, Naïve Bayes and Support vector machine with linear and RBF kernel function and showing their accuracy.

2.3 Review on Prediction of Diabetes using Data Mining Technique

Diabetes mellitus is one of the world's major diseases. Millions of people are affected by the disease. The risk of diabetes is increasing day by day and is found mostly in women than men. The diagnosis of diabetes is a tedious process. So with improvement in science and technology it is made easy to predict the disease. The purpose is to diagnose whether the person is affected by diabetes or not using K Nearest Neighbor classification technique. The diabetes dataset is taken as the training data and the details of the patient are taken as testing data. The training data are classified by using the KNN classifier and secondly the target data is predicted. KNN algorithm used here would be more efficient for both classification and prediction. The results are analyzed with different values for the parameter k.

2.4 A Prediction Technique in Data Mining for Diabetes Mellitus

Diabetes mellitus is a chronic disease characterized by hyperglycemia. It may cause many complications. According to the growing morbidity in recent years, in 2040, the world's diabetic patients will reach 642 million, which means that one of the ten adults in the future is suffering from diabetes. There is no doubt that this alarming figure needs great attention. With the rapid development of machine learning, machine learning has been applied to many aspects of medical health. In this study, we used decision tree, random forest and neural network to predict diabetes mellitus. The dataset is the hospital physical examination data in

Luzhou, China. It contains 14 attributes. In this study, five-fold cross validation was used to examine the models. In order to verify the universal applicability of the methods, we chose some methods that have the better performance to conduct independent test experiments. We randomly selected 68994 healthy people and diabetic patients' data, respectively as training set. Due to the data unbalance, we randomly extracted 5 times data. And the result is the average of these five experiments. In this study, we used principal component analysis (PCA) and minimum redundancy maximum relevance (mRMR) to reduce the dimensionality. The results showed that prediction with random forest could reach the highest accuracy ($ACC = 0.8084$) when all the attributes were used.

2.5 Utilization of Data Mining Techniques for Diagnosis of Diabetes Mellitus - A Case Study

Data mining looks through a large amount of data to extract useful information. The most important and popular data mining techniques are classification, association, clustering, prediction and sequential patterns. In health concern businesses, data mining plays an important role in early prediction of diseases. In general to detect a disease numerous tests must be conducted in a patient. The usage of data mining techniques in disease prediction is to reduce the test and increase the accuracy of rate of detection. One of the most common diseases among young adult is Diabetes mellitus. This develops at a middle age and more common in obese children and adolescents. In order to reduce the population with diabetes mellitus it should be detected at an earlier stage, hence a quick and efficient detection mechanism has to be discovered. The principle of this study is to apply various data mining techniques which are noteworthy to prediction of diabetes mellitus and extract hidden patterns from the PIMA Indian diabetes dataset available at UCI Machine Learning Repository.

CHAPTER-3

METHODOLOGY

3.1 AIM OF THE PROJECT

- The primary aim of the present study was to implement four models to predict DT2M applying data mining techniques based on the lncRNA variables. The research objectives of our study were:
- Implementing data mining techniques for prediction of the DT2M.
- Comparing the applied methods.
- Selecting the best model for the T2DM prediction.
- We used the variables for predicting T2DM and comparing the performance of the various data mining techniques. For the implementation of the algorithm

3.2 EXISTING SYSTEM

The existing system was taking in order to meets the demands of this system and solve the problems of the existing system by implementing the naïve beyes classifier.

3.2.1 DISADVANTAGES OF EXISTING SYSTEM

- The system is not fully automated, it needs data from user for full diagnosis.

3.3 PROPOSED SYSTEM

The proposed diabetes prediction system has two main stages that work together to achieve the desired results. The first stage of the proposed system is the data preparation, and the second one is the classification. However, the input into the system is the dataset and the output will be one class which represent the healthy or the diabetic. The implementation of

this new system will help to reduce the stressful process, the result of the experiment shows that the proposed system has a better prediction in terms of accuracy. We have applied K-Nearest neighbor algorithm on the training and test sample data and obtained results for different values of K which is number of nearest neighbors.

3.3.1 ADVANTAGES OF PROPOSED SYSTEM

- User can diagnose their diabetes and get instant result.
- K- Nearest neighbor Algorithm is a fast, highly scalable algorithm.

3.4 SYSTEM ARCHITECTURE

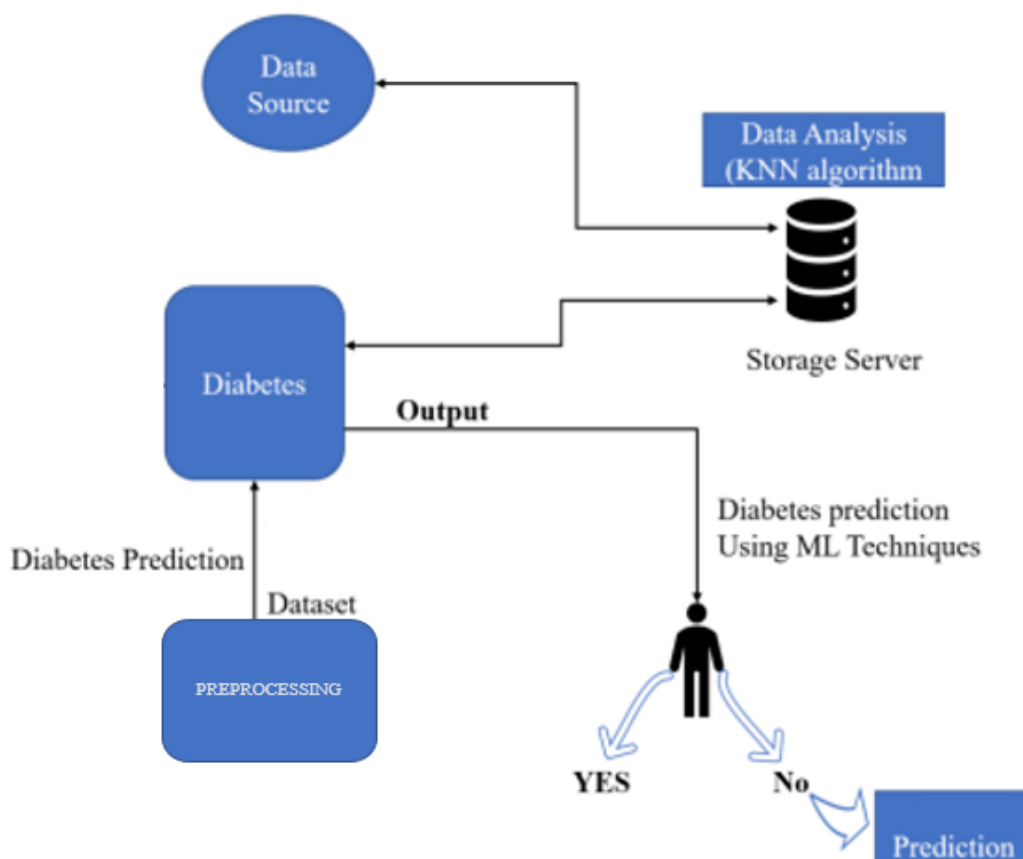


Fig no 3.4 :system Architecture

3.5 SYSTEM REQUIREMENTS :

3.5.1 HARDWARE REQUIREMENTS :

- System : Pentium Dual Core.
- Hard Disk : 120 GB.
- Monitor : 15” LED
- Input Devices : Keyboard, Mouse
- Ram : 4 GB.

3.5.2 SOFTWARE REQUIREMENTS:

- Operating system: Windows 7/10.
- Coding Language :Python
- Tool : python 3.6.1

3.6 MODULES DESCRIPTION

- Data Constraints
- Train Dataset and Test Dataset
- Pre-processing of data
- Feature Extraction
- ML Algorithm: KNN
- Result

3.6.1 Data Constraints

Data is a collection global dataset. IN this system use Pima Indian data set is used for training a model.

3.6.2 Train Dataset and Test Dataset

The training data is a initial set of data which is used to understand the program. This is the one in which we have to train the model first because to set the feature and this data is available on system. This data is used to teach the machine for do different actions. It is the data in which model can learn with algorithm to teach the model and doing work automatic. Testing data is the input given to a software. It shows the data affects when the execution of the module that specifying and this is basically used for testing.

3.6.3 Pre-processing of data

Data preprocessing is a process in which that is actual use for converting the basic data into the clean data set. It is the step in which the data transform or an encode to the state that the machine can be easily parse. The major task of data preprocessing in learning process is to remove the unwanted data and filling the missed value. So that it help to machine can be trained easily.

3.6.4 Feature Extraction

Feature Extraction is the method in which it used for alter the key data for features of outcomes. This, trait square is used to compute the characteristics of designs given that facilitate in different amid the class of key pattern details. This method involving to decrease the counts of resource required to describe the huge set of data. Feature extraction is an attribute reduction process. This is also used to increasing the speed and effectiveness of supervised learning.

3.6.5 ML Algorithm: KNN

The k-nearest neighbor's is a ML algorithm is the non-parametric method proposed by Thomas Cover used for Regression and Classification. This algorithm is mainly used for the classification of problems in the industry. KNN algorithm is a type of instance-based learning method. This algorithm relies on the distance for objects classification, training data normalizing to the improve its accuracy dramatically. The neighbors are derived from the set of things for which classes or object property values are known. It can be thought of as a training set for the algorithm, although no explicit training steps are required.

3.7 DATA FLOW DIAGRAM:

1. The DFD is also called as bubble chart. It is a simple graphical formalism that can be used to represent a system in terms of input data to the system, various processing carried out on this data, and the output data is generated by this system.
2. The data flow diagram (DFD) is one of the most important modeling tools. It is used to model the system components. These components are the system process, the data used by the process, an external entity that interacts with the system and the information flows in the system.
3. DFD shows how the information moves through the system and how it is modified by a series of transformations. It is a graphical technique that depicts information flow and the transformations that are applied as data moves from input to output.
4. DFD is also known as bubble chart. A DFD may be used to represent a system at any level of abstraction. DFD may be partitioned into levels that represent increasing information flow and functional detail.

3.7.1 DATA FLOW DIAGRAMS

3.7.2.1 DFD-LEVEL-0

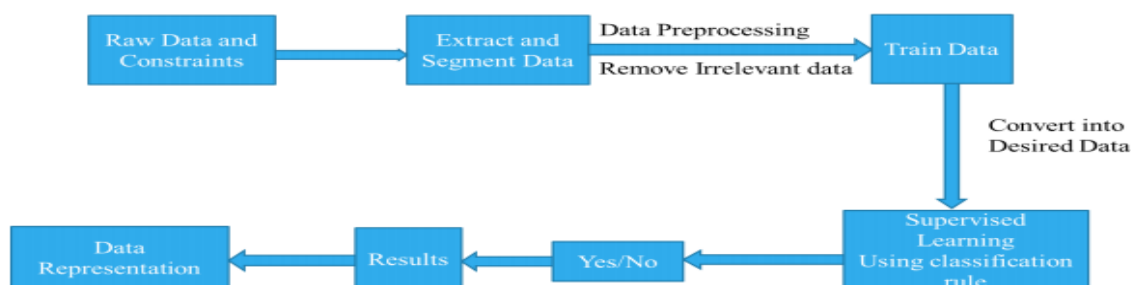


Fig no.3.7.2.1:DFD-level-0

3.7.2.2 DFD-LEVEL-1

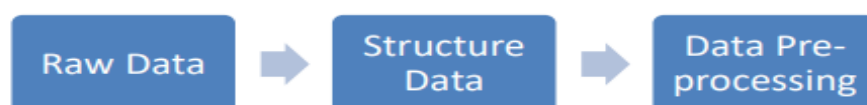


Fig no..3.7.2.2:DFD-level-1

3.7.2.3 DFD-LEVEL-2

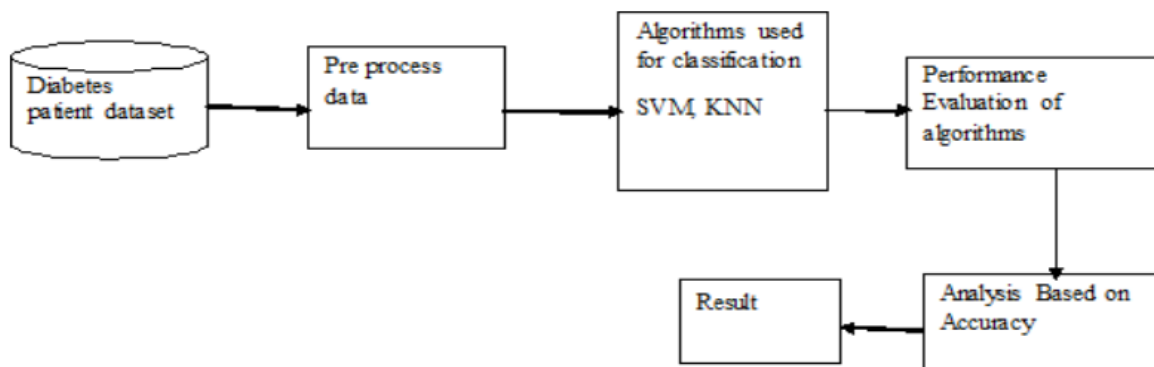


Fig no.3.7.2.3:DFD-level-2

3.7.3 CLASS DIAGRAM

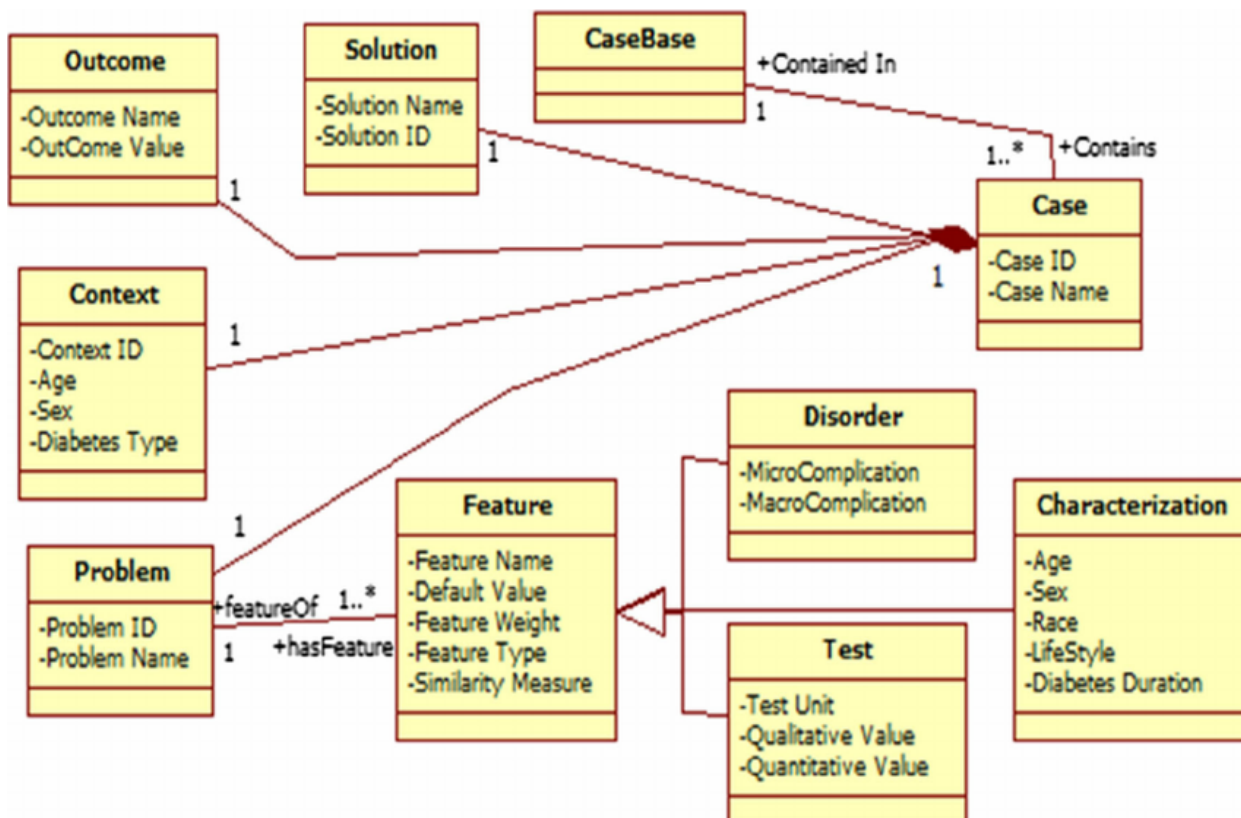


FIG NO.3.7.3:CLASS DIAGRAM

3.8 SOFTWARE ENVIRONMENT

3.8.1 Python:

Python is a high-level, interpreted, interactive and object-oriented scripting language. Python is designed to be highly readable. It uses English keywords frequently where as other languages use punctuation, and it has fewer syntactical constructions than other languages.

- **Python is Interpreted** – Python is processed at runtime by the interpreter. You do not need to compile your program before executing it. This is similar to PERL and PHP.
- **Python is Interactive** – You can actually sit at a Python prompt and interact with the interpreter directly to write your programs.
- **Python is Object-Oriented** – Python supports Object-Oriented style or technique of programming that encapsulates code within objects.
- **Python is a Beginner's Language** – Python is a great language for the beginner-level programmers and supports the development of a wide range of applications from simple text processing to WWW browsers to games.

3.8.1.1 History of Python

Python was developed by Guido van Rossum in the late eighties and early nineties at the National Research Institute for Mathematics and Computer Science in the Netherlands.

Python is derived from many other languages, including ABC, Modula-3, C, C++, Algol-68, SmallTalk, and Unix shell and other scripting languages.

Python is copyrighted. Like Perl, Python source code is now available under the GNU General Public License (GPL).

Python is now maintained by a core development team at the institute, although Guido van Rossum still holds a vital role in directing its progress.

3.8.1.2 Python Features

Python's features include –

- **Easy-to-learn** – Python has few keywords, simple structure, and a clearly defined syntax. This allows the student to pick up the language quickly.
- **Easy-to-read** – Python code is more clearly defined and visible to the eyes.

- **Easy-to-maintain** – Python's source code is fairly easy-to-maintain.
- **A broad standard library** – Python's bulk of the library is very portable and cross-platform compatible on UNIX, Windows, and Macintosh.
- **Interactive Mode** – Python has support for an interactive mode which allows interactive testing and debugging of snippets of code.
- **Portable** – Python can run on a wide variety of hardware platforms and has the same interface on all platforms.
- **Extendable** – You can add low-level modules to the Python interpreter. These modules enable programmers to add to or customize their tools to be more efficient.
- **Databases** – Python provides interfaces to all major commercial databases.
- **GUI Programming** – Python supports GUI applications that can be created and ported to many system calls, libraries and windows systems, such as Windows MFC, Macintosh, and the X Window system of Unix.
- **Scalable** – Python provides a better structure and support for large programs than shell scripting.

Apart from the above-mentioned features, Python has a big list of good features, few are listed below –

- It supports functional and structured programming methods as well as OOP.
- It can be used as a scripting language or can be compiled to byte-code for building large applications.
- It provides very high-level dynamic data types and supports dynamic type checking.
- It supports automatic garbage collection.
- It can be easily integrated with C, C++, COM, ActiveX, CORBA, and Java.

Python is available on a wide variety of platforms including Linux and Mac OS X. Let's understand how to set up our Python environment.

3.8.1.3 Getting Python

The most up-to-date and current source code, binaries, documentation, news, etc., is available on the official website of Python <https://www.python.org>.

Windows Installation

Here are the steps to install Python on Windows machine.

- Open a Web browser and go to <https://www.python.org/downloads/>.
- Follow the link for the Windows installer python-XYZ.msifile where XYZ is the version you need to install.
- To use this installer python-XYZ.msi, the Windows system must support Microsoft Installer 2.0. Save the installer file to your local machine and then run it to find out if your machine supports MSI.
- Run the downloaded file. This brings up the Python install wizard, which is really easy to use. Just accept the default settings, wait until the install is finished, and you are done.

The Python language has many similarities to Perl, C, and Java. However, there are some definite differences between the languages.

First Python Program

Let us execute programs in different modes of programming.

3.8.2 Interactive Mode Programming

Invoking the interpreter without passing a script file as a parameter brings up the following prompt –

```
$ python
```

```
Python2.4.3(#1,Nov112010,13:34:43)
```

```
[GCC 4.1.220080704(RedHat4.1.2-48)] on linux2
```

```
Type "help", "copyright", "credits" or "license" for more information.
```

```
>>>
```

Fig no.3.8.2:Interactive Mode Programming

Type the following text at the Python prompt and press the Enter –

```
>>>print"Hello, Python!"
```

If you are running new version of Python, then you would need to use print statement with parenthesis as in **print ("Hello, Python!");**. However in Python version 2.4.3, this produces the following result –

```
Hello, Python!
```

3.8.2.1 Script Mode Programming

Invoking the interpreter with a script parameter begins execution of the script and continues until the script is finished. When the script is finished, the interpreter is no longer active.

Let us write a simple Python program in a script. Python files have extension **.py**. Type the following source code in a test.py file –

```
print"Hello, Python!"
```

We assume that you have Python interpreter set in PATH variable. Now, try to run this program as follows –

```
$ python test.py
```

This produces the following result –

```
Hello, Python!
```

3.8.3 Flask Framework:

Flask is a web application framework written in Python. Armin Ronacher, who leads an international group of Python enthusiasts named Pocco, develops it. Flask is based on Werkzeug WSGI toolkit and Jinja2 template engine. Both are Pocco projects.

Http protocol is the foundation of data communication in world wide web. Different methods of data retrieval from specified URL are defined in this protocol.

The following table summarizes different http methods –

Sr.No	Methods & Description
1	GET Sends data in unencrypted form to the server. Most common method.
2	HEAD Same as GET, but without response body
3	POST Used to send HTML form data to server. Data received by POST method is not cached by server.
4	PUT Replaces all current representations of the target resource with the uploaded content.
5	DELETE Removes all current representations of the target resource given by a URL

Fig no.3.8.3:Methods and Description

By default, the Flask route responds to the **GET** requests. However, this preference can be altered by providing methods argument to **route()** decorator.

In order to demonstrate the use of **POST** method in URL routing, first let us create an HTML form and use the **POST** method to send form data to a URL.

Save the following script as login.html

```
<html>

<body>

<form action="http://localhost:5000/login" method="post">

<p>Enter Name:</p>

<p><input type="text" name="nm"/></p>

<p><input type="submit" value="submit"/></p>

</form>

</body>

</html>
```

Now enter the following script in Python shell.

```
from flask import Flask, redirect, url_for, request

app=Flask(__name__)

@app.route('/success/<name>')

def success(name):

return 'welcome %s'% name

@app.route('/login', methods=['POST', 'GET'])

def login():

if request.method=='POST':
```

```
user=request.form['nm']

return redirect(url_for('success',name= user))

else:

user=request.args.get('nm')

return redirect(url_for('success',name= user))

if __name__ == '__main__':

app.run(debug =True)
```

After the development server starts running, open **login.html** in the browser, enter name in the text field and click **Submit**.

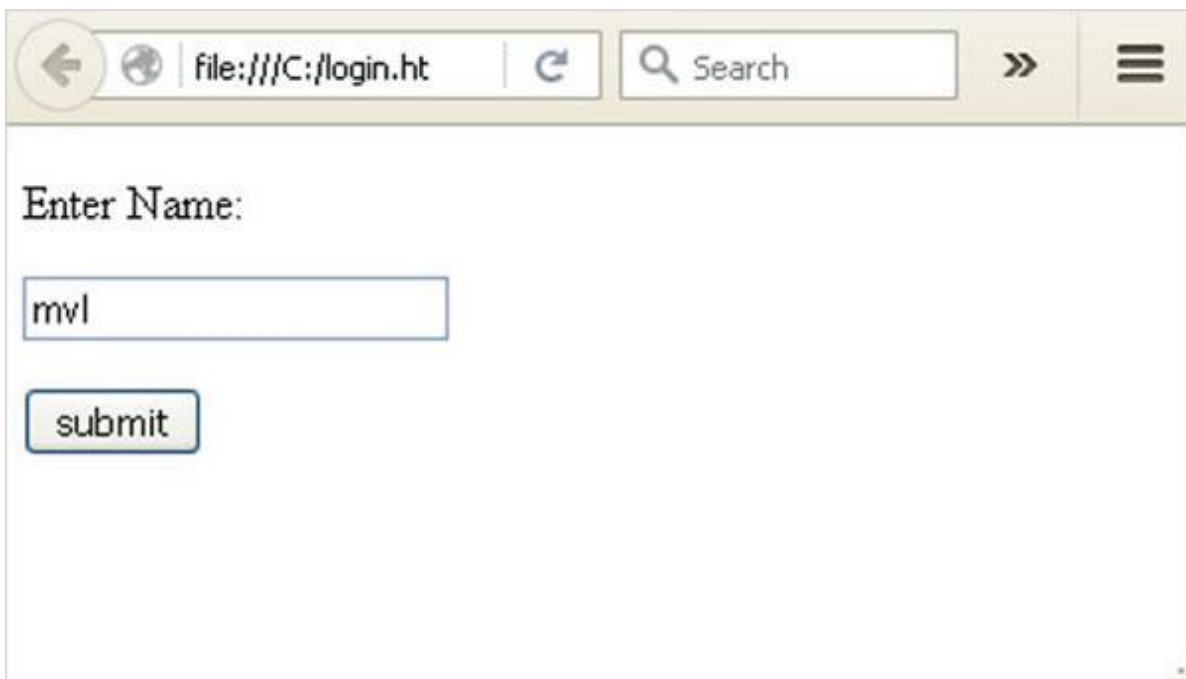
A screenshot of a web browser window. The address bar shows the file path 'file:///C:/login.ht'. The page content includes a label 'Enter Name:', a text input field with the value 'mvl', and a button labeled 'submit'.

Fig no.3.8.3.1:login html

Form data is POSTed to the URL in action clause of form tag.

http://localhost/login is mapped to the **login()** function. Since the server has received data by **POST** method, value of 'nm' parameter obtained from the form data is obtained by –

```
user = request.form['nm']
```

It is passed to **‘/success’** URL as variable part. The browser displays a **welcome** message in

the window.

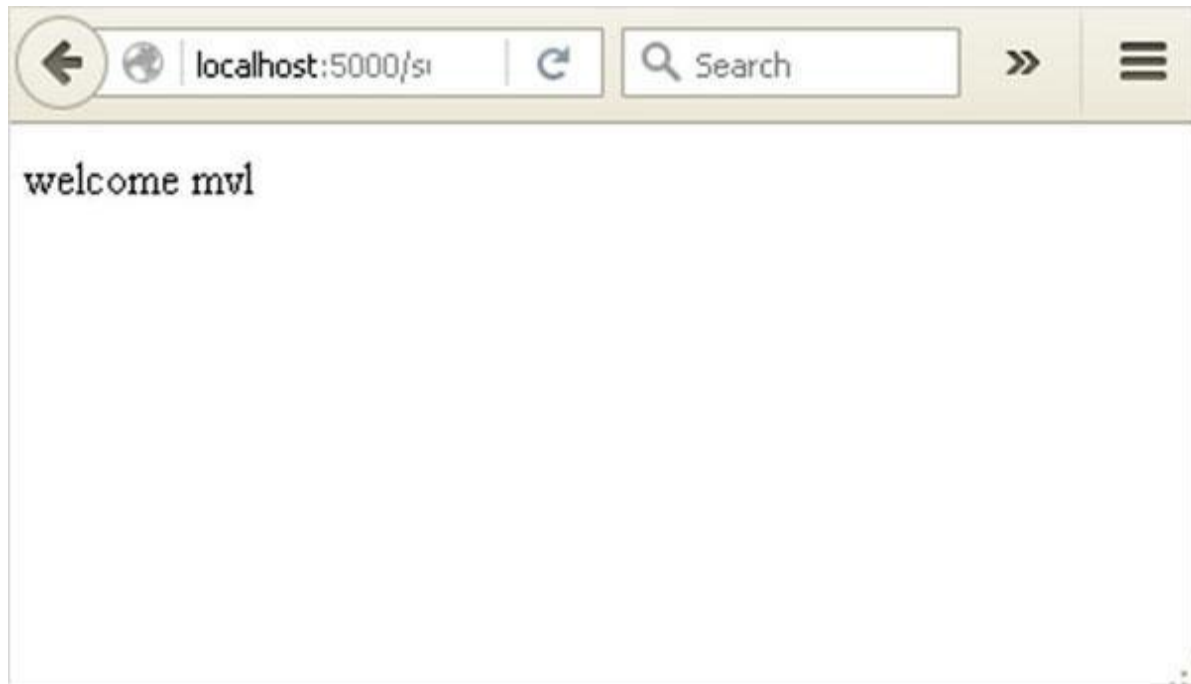


Fig no.3.8.3.2:Success url

Change the method parameter to '**GET**' in **login.html** and open it again in the browser. The data received on server is by the **GET** method. The value of 'nm' parameter is now obtained by –

```
User = request.args.get('nm')
```

Here, **args** is dictionary object containing a list of pairs of form parameter and its corresponding value. The value corresponding to 'nm' parameter is passed on to '/success' URL as before.

CHAPTER-4

RESULT AND PERFORMANCE EVALUATION

4.1 Result

After taking that input data from the system will able to divine the statistics by appeal the ML algorithm & also provided the foremost output in the devise of different in between to detection the most accurate to treatment to diabetes millets.

4.2 REQUIREMENT ANALYSIS

Requirement analysis, also called requirement engineering, is the process of determining user expectations for a new modified product. It encompasses the tasks that determine the need for analysing, documenting, validating and managing software or system requirements. The requirements should be documentable, actionable, measurable, testable and traceable related to identified business needs or opportunities and define to a level of detail, sufficient for system design.

4.3 FUNCTIONAL REQUIREMENTS

It is a technical specification requirement for the software products. It is the first step in the requirement analysis process which lists the requirements of particular software systems including functional, performance and security requirements. The function of the system depends mainly on the quality hardware used to run the software with given functionality.

4.3.1 Usability

It specifies how easy the system must be use. It is easy to ask queries in any format which is short or long, porter stemming algorithm stimulates the desired response for user.

4.3.2 Robustness

It refers to a program that performs well not only under ordinary conditions but also under unusual conditions. It is the ability of the user to cope with errors for irrelevant queries during execution.

4.3.3 Security

The state of providing protected access to resource is security. The system provides good security and unauthorized users cannot access the system there by providing high security.

4.3.4 Reliability

It is the probability of how often the software fails. The measurement is often expressed in MTBF (Mean Time Between Failures). The requirement is needed in order to ensure that the processes work correctly and completely without being aborted. It can handle any load and survive and survive and even capable of working around any failure.

4.3.5 Compatibility

It is supported by version above all web browsers. Using any web servers like localhost makes the system real-time experience.

4.3.6 Flexibility

The flexibility of the project is provided in such a way that it has the ability to run on different environments being executed by different users.

4.3.7 Safety

Safety is a measure taken to prevent trouble. Every query is processed in a secured manner without letting others to know one's personal information.

4.4 NON- FUNCTIONAL REQUIREMENTS

4.4.1 Portability

It is the usability of the same software in different environments. The project can be run in any operating system.

4.4.2 Performance

These requirements determine the resources required, time interval, throughput and everything that deals with the performance of the system.

4.4.3 Accuracy

The result of the requesting query is very accurate and high speed of retrieving information. The degree of security provided by the system is high and effective.

4.4.4 Maintainability

Project is simple as further updates can be easily done without affecting its stability. Maintainability basically defines that how easy it is to maintain the system. It means that how easy it is to maintain the system, analyse, change and test the application. Maintainability of this project is simple as further updates can be easily done without affecting its stability.

CHAPTER-5

TESTING

SYSTEM DESIGN AND TESTING PLAN

5.1 INPUT DESIGN

The input design is the link between the information system and the user. It comprises the developing specification and procedures for data preparation and those steps are necessary to put transaction data in to a usable form for processing can be achieved by inspecting the computer to read data from a written or printed document or it can occur by having people keying the data directly into the system. The design of input focuses on controlling the amount of input required, controlling the errors, avoiding delay, avoiding extra steps and keeping the process simple. The input is designed in such a way so that it provides security and ease of use with retaining the privacy. Input Design considered the following things:

What data should be given as input?

How the data should be arranged or coded?

The dialog to guide the operating personnel in providing input.

Methods for preparing input validations and steps to follow when error occur.

5.2 OUTPUT DESIGN

A quality output is one, which meets the requirements of the end user and presents the information clearly. In any system results of processing are communicated to the users and to other system through outputs. In output design it is determined how the information is to be displaced for immediate need and also the hard copy output. It is the most important and direct source information to the user. Efficient and intelligent output design improves the system's relationship to help user decision-making.

The output form of an information system should accomplish one or more of the following objectives.

Convey information about past activities, current status or projections of the

Future.

Signal important events, opportunities, problems, or warnings.

Trigger an action.

Confirm an action.

5.3 SYSTEM STUDY

5.3.1 FEASIBILITY STUDY

The feasibility of the project is analyzed in this phase and business proposal is put forth with a very general plan for the project and some cost estimates. During system analysis the feasibility study of the proposed system is to be carried out. This is to ensure that the proposed system is not a burden to the company. For feasibility analysis, some understanding of the major requirements for the system is essential.

Three key considerations involved in the feasibility analysis are

- ◆ ECONOMICAL FEASIBILITY
- ◆ TECHNICAL FEASIBILITY
- ◆ SOCIAL FEASIBILITY

5.3.2 ECONOMICAL FEASIBILITY

This study is carried out to check the economic impact that the system will have on

the organization. The amount of fund that the company can pour into the research and development of the system is limited. The expenditures must be justified. Thus the developed system as well within the budget and this was achieved because most of the technologies used are freely available. Only the customized products had to be purchased.

TECHNICAL FEASIBILITY

This study is carried out to check the technical feasibility, that is, the technical requirements of the system. Any system developed must not have a high demand on the available technical resources. This will lead to high demands on the available technical resources. This will lead to high demands being placed on the client. The developed system must have a modest requirement, as only minimal or null changes are required for implementing this system.

5.3.3 SOCIAL FEASIBILITY

The aspect of study is to check the level of acceptance of the system by the user. This includes the process of training the user to use the system efficiently. The user must not feel threatened by the system, instead must accept it as a necessity. The level of acceptance by the users solely depends on the methods that are employed to educate the user about the system and to make him familiar with it. His level of confidence must be raised so that he is also able to make some constructive criticism, which is welcomed, as he is the final user of the system.

5.3.4 SYSTEM TESTING

The purpose of testing is to discover errors. Testing is the process of trying to discover every conceivable fault or weakness in a work product. It provides a way to check the functionality of components, sub assemblies, assemblies and/or a finished product. It is the process of exercising software with the intent of ensuring that the Software system meets its requirements and user expectations and does not fail in an

unacceptable manner. There are various types of test. Each test type addresses a specific testing requirement.

5.4 TYPES OF TESTS

5.4.1 Unit testing

Unit testing involves the design of test cases that validate that the internal program logic is functioning properly, and that program inputs produce valid outputs. All decision branches and internal code flow should be validated. It is the testing of individual software units of the application .it is done after the completion of an individual unit before integration. This is a structural testing, that relies on knowledge of its construction and is invasive. Unit tests perform basic tests at component level and test a specific business process, application, and/or system configuration. Unit tests ensure that each unique path of a business process performs accurately to the documented specifications and contains clearly defined inputs and expected results.

5.4.2 Integration testing

Integration tests are designed to test integrated software components to determine if they actually run as one program. Testing is event driven and is more concerned with the basic outcome of screens or fields. Integration tests demonstrate that although the components were individually satisfaction, as shown by successfully unit testing, the combination of components is correct and consistent. Integration testing is specifically aimed at exposing the problems that arise from the combination of components.

5.4.3 Functional test

Functional tests provide systematic demonstrations that functions tested are available as specified by the business and technical requirements, system documentation, and user manuals.

Functional testing is centered on the following items:

Valid Input : identified classes of valid input must be accepted.
Invalid Input : identified classes of invalid input must be rejected.
Functions : identified functions must be exercised.
Output : identified classes of application outputs must be exercised.
Systems/Procedures: interfacing systems or procedures must be invoked.

Organization and preparation of functional tests is focused on requirements, key functions, or special test cases. In addition, systematic coverage pertaining to identify Business process flows; data fields, predefined processes, and successive processes must be considered for testing. Before functional testing is complete, additional tests are identified and the effective value of current tests is determined.

5.4.4 System Test

System testing ensures that the entire integrated software system meets requirements. It tests a configuration to ensure known and predictable results. An example of system testing is the configuration oriented system integration test. System testing is based on process descriptions and flows, emphasizing pre-driven process links and integration points.

5.4.5 White Box Testing

White Box Testing is a testing in which the software tester has knowledge of the inner workings, structure and language of the software, or at least its purpose. It is purpose. It is used to test areas that cannot be reached from a black box level.

5.4.6 Black Box Testing

Black Box Testing is testing the software without any knowledge of the inner workings, structure or language of the module being tested. Black box tests, as most other kinds of tests, must be written from a definitive source document, such as specification or requirements document, such as specification or requirements document. It is a testing in which the software under test is treated, as a black box .you cannot “see” into it. The test provides inputs and responds to outputs without considering how the software works.

5.5 Unit Testing:

Unit testing is usually conducted as part of a combined code and unit test phase of the software lifecycle, although it is not uncommon for coding and unit testing to be conducted as two distinct phases.

Test strategy and approach

Field testing will be performed manually and functional tests will be written in detail.

Test objectives

- All field entries must work properly.
- Pages must be activated from the identified link.
- The entry screen, messages and responses must not be delayed.

Features to be tested

- Verify that the entries are of the correct format
- No duplicate entries should be allowed
- All links should take the user to the correct page.

5.6 Integration Testing

Software integration testing is the incremental integration testing of two or more integrated software components on a single platform to produce failures caused by interface defects.

The task of the integration test is to check that components or software applications, e.g. components in a software system or – one step up – software applications at the company level – interact without error.

Test Results: All the test cases mentioned above passed successfully. No defects encountered.

5.7 Acceptance Testing

User Acceptance Testing is a critical phase of any project and requires significant participation by the end user. It also ensures that the system meets the functional requirements.

Test Results: All the test cases mentioned above passed successfully. No defects encountered.

CHAPTER-6

CONCLUSION

CONCLUSION:

There is a concern among physicians how to detect diabetes at its infancy stage. This study had made an effort in designing the system in predicting the diabetes. The experimental work implemented ML algorithms and the evaluation was done on various measures. The experiment was carried on the Diabetes data set and the results confirmed the designed system had an accuracy of 79.17% using the supplying regression classification formula. The designed system using this ML algorithm can also be customized in predicting other alternative diseases. The research can be further enhanced in implementing other ML algorithm in improving the prediction of diabetes.

REFERENCES

- [1] Deepti Sisodiaa, Dilip Singh Sisodiab, 2018, Prediction of Diabetes using Classification Algorithms, International Conference on Computational Intelligence and Data Science - ICCIDS 2018 ,Science Direct Procedia Computer Science 132 (2018) 1578–1585.
- [2] G. Krishnaveni*, T. Sudha,” A Novel Technique To Predict Diabetic Disease Using Data Mining Classification Techniques” in International Conference on Innovative Applications in Engineering and Information Technology (ICIAEIT2017), vol. 3, Issue 1, pp. 5-11, 2017.
- [3] Vrushali B., and Rakhi W., “Review on Prediction of Diabetes using Data Mining Technique”, International Journal of Research and Scientific Innovation (IJRSI), Volume IV, Issue IA, pp. 43-46, January 2017.
- [4] Harleen and Dr. Pankaj B.,”A Prediction Technique in Data Mining for Diabetes Mellitus,” Journal of Management Sciences and Technology, vol. 4, Issue 1, pp. 1-12, 2016.
- [5] Thirumal P., and Nagarajan N.,” Utilization of Data Mining Techniques for Diagnosis of Diabetes Mellitus - A Case Study”, ARPN Journal of Engineering and Applied Sciences, Vol. 10, No. 1, pp. 8-13, January 2015.
- [6] Iyer A., Jeyalatha S., Sumbaly R., “Diagnosis of diabetes using classification mining techniques,” International Journal of Data Mining & Knowledge Management Process (IJDKP), vol. 5, no. 1, 2015.
- [7] Perveen, S., Shahbaz, M.,Guergachi,A.,Keshavjee,K.,2016.PerformanceAnalysis of Data MiningClassificationTechniques to Predict Diabetes. Procedia Computer Science 82, 115–121. doi:10.1016/j.procs.2016.04.016.
- [8] Orabi, K.M.,Kamal, Y.M.,Rabah,T.M.,2016.EarlyPredictive SystemforDiabetes MellitusDisease,in:IndustrialConference on Data Mining, Springer. Springer. pp. 420–427.