DISEASE PREDICTION USING MACHINELEARNING

***a project report submitted by***

**L.N.V.MUKESH (REG. NO: UR18CS229) M.KRISHNA SAI (REG. NO: UR18CS238)**

**S.RAKESH (REG. NO: URK18CS246)**

**S.V.THARUN KUMAR REDDY**

**(REG. NO: URK18CS247)**

***in partial fulfillment for the award of the degree of***

**BACHELOR OF TECHNOLOGY**

***in***

**COMPUTER SCIENCE AND ENGINEERING**

***under the supervision of***

**DR.R.VENKATESAN, M.TECH., Ph.D.**





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

**KARUNYA INSTITUTE OF TECHNOLOGY AND SCIENCES**

(Declared as Deemed-to-be-University under Sec-3 of the UGC Act, 1956)

**Karunya Nagar, Coimbatore - 641 114, India.**

# APRIL 2022

# BONAFIDE CERTIFICATE

Certified that this project report **“DISEASE PREDICTION USING MACHINE LEARNING”** is the bonafide work of “L.N.V.MUKESH (REG. NO:UR18CS229), MASINENI KRISHNA SAI (REG.NO:URK18CS238), S.RAKESH (REG.NO:URK18CS246), S.V.THARUN KUMAR REDDY(REG.NO:URK18CS247)**”** who carried out the project work under my supervision.

**SIGNATURE SIGNATURE**

# Dr. J. Immanuel Johnraja DR.R.Venkatesan,M.Tech.,Ph.D

**Head of the Department** Department of Computer Science and Engineering

**Supervisor**

Associate Professor

Department of Computer Science and Engineering

Submitted for the Project Viva Voce held on \_\_\_\_\_12-04-2022\_\_\_\_\_\_\_\_\_\_\_

**Examiner**

**ACKNOWLEDGEMENT**

First and foremost, we praise and thank **ALMIGHTY GOD** for giving us the will power and confidence to carry out our project.

We are grateful to our beloved founders Late **Dr. D.G.S. Dhinakaran**, **C.A.I.I.B, Ph.D.,** and **Dr. Paul Dhinakaran**, **M.B.A., Ph.D.,** for their love and always remembering us in their prayers.

We extend our thanks to **Dr. P. Mannar Jawahar, Ph.D.,** our honorable vice chancellor, **Dr. E. J. James, Ph.D.,** and **Dr. Ridling Margaret Waller, Ph.D.,**our honorable Pro-Vice Chancellor(s) and **Dr. R. Elijah Blessing, Ph.D.,** our respected registrar for giving us the opportunity to carry out this project.

We are thankful to **Dr. G. Prince Arulraj, M.E., Ph.D.,** Dean (Engineering & Technology) for his support and encouragement.

We would like to place our heart-felt thanks and gratitude to **Dr. J. Immanuel Johnraja, Ph.D.,** HOD, Department of Computer Science and Engineering for his encouragement and guidance.

We are grateful to our guide, **DR.R.Venkatesan, M.Tech., Ph.D.,** Associate Professor, Department of Computer Science and Engineering for his valuable support, advice and encouragement.

We also thank all the staff members of the Department for extending their helping hands to make this project work a success.

We would also like to thank all my friends and my parents who have prayed and helped me during the projectwork.

**ABSTRACT**

Prediction using Machine Learning is a system which predicts the disease based on the information or the symptoms he/she 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. It is a system which provides the user the tips and tricks to maintain the health system of the user and it provides a way to find out the disease using this prediction. Now a day’s health industry plays major role in curing the diseases of the patients so this is also some kind of help for the health industry to tell the user and also it is useful for the user in case he/she doesn’t want to go to the hospital or any other clinics, so just by entering the symptoms and all other useful information the user can get to know the disease he/she is suffering from and the health industry can also get benefit from this system by just asking the symptoms from the user and entering in the system and in just few seconds they can tell the exact and up to some extent the accurate diseases. This Disease Prediction Using Machine Learning is completely done with the help of Machine Learning and Python Programming language with Tkinter Interface for it and also using the dataset that is available previously by the hospitals using that we will predict the disease.

# 

# TABLE OF CONTENTS

**ACKNOWLEDGEMENT iii**

**ABSTRACT iv**

**LIST OF FIGURES v**

**LIST OF TABLES vii**

1. **INTRODUCTION**

1.1DISEASE PREDICTION 1

* 1. PROBLEM DEFINITION 2

1.3 PROJECT PURPOSE 3

1.4 CHAPTER WISE SUMMARY 3

1. **SYSTEM ANALYSIS**

* 1. EXISTING SYSTEM 4

2.2 PROPOSED SYSTEM 5

2.3 USECASE ANALYSIS 6

2.4 REQUIREMENT SPECIFICATION 7

2.4.1 FUNCTIONAL REQUIREMENTS

2.4.2 NON-FUNCTIONAL REQUIREMENTS

2.4.3 HARDWARE REQUIREMENTS

2.4.4 SOFTWARE REQUIREMENTS

**3.** **SYSTEM DESIGN**

3.1 DETAILED DESIGN (ARCHITECTURE DIAGRAM ) 9

3.2 DESIGN OF METHODOLOGY 10

* 1. MODULES 11

3.4 FRAMEWORK 12

**4. SYSTEM IMPLEMENTATION**

4.1 MODULE IMPLEMENTATION 15

4.1.1 DECISION TREE ALGORITHM 16

4.1.2 NAÏVE BAYES ALGORITHM 17

4.2 TESTING 18

**5 CONCLUSION AND FUTURE SCOPE**

5.1 CONCLUSION 19

5.2 FUTURE ENCHANCEMENT 20

**APPENDIX A – SOURCE CODE** 21

**APPENDIX B – SCREENSHOTS** 32

**LIST OF FIGURES**

**Figure No Figure Name Page No**

1.1 Use Case Analysis 6

1.2 System Architecture 10

1.3Data Flow Diagram 11

1.4 Interface and Framework Diagram 15

1.5 Decision Tree problem 16

1.6 Naïve Bayes Algorithm 17

1.7 Naïve Bayes Formula 18

1.8 Sample picture of the dataset 32

1.9 Importing the libraries 32

2.0 Training the data 33

2.1 Testing the data 33

2.2 Description of the data 34

2.3 Data Analysis Graph 34

2.4 Correlation Graph 35

**LIST OF TABLES**

**TABLE TABLE NAME PAGE NO**

1.0 Testing 18

|  |
| --- |
|  |
| **REFERENCES** |

36

|  |  |
| --- | --- |
|  |  |

**CHAPTER – 1**

**INTRODUCTION**

* 1. **DISEASE PREDICTION**

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 or the symptoms he/she 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. It is a system which provides the user the tips and tricks to maintain the health system of the user and it provides a way to find out the disease using this prediction. Now a day’s health industry plays major role in curing the diseases of the patients so this is also some kind of help for the health industry to tell the user and also it is useful for the user in case he/she doesn’t want to go to the hospital or any other clinics, so just by entering the symptoms and all other useful information the user can get to know the disease he/she is suffering from and the health industry can also get benefit from this system by just asking the symptoms from the user and entering in the system and in just few seconds they can tell the exact and up to some extent the accurate diseases.

According to research there are 70% peoples in India suffers from general disease and 25%of peoples face death due to early ignorance the main motive to develop this project is that a user can sit at their convenient place and have a check-up of their health the UI is designed in such a simple way that everyone can easily operate on it and can have a check-up. The main reason of ignorance is laziness to consult a doctor and time concern the peoples have involved themselves so much that they have no time to take an appointment and consult the doctor which later results into fatal disease.

# 1.2 PROBLEM STATEMENT AND DISCUSSION

Now a day’s in Health Industry there are various problems related to machines or devices 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 the accurate predictions based on information provided by the user and also based on the datasets that are available in that machine. The health industry in information yet and knowledge poor and this industry is very vast industry which has lot of work to be done. So, with the help of all those algorithms, techniques and methodologies we have done this project which will help the peoples who are in the need. So the problem here is that many people goes to hospitals or clinic to know how is their health and how much they are improving in the given days, but they have to travel to get to know there answers and sometimes the patients may or may not get the results based on various factors such as doctor might be on leave or some whether problem so he might not have come to the hospital and many more reasons will be there so to avoid all those reasons and confusion we are making a project which will help all those person’s and all the patients who are in need to know the condition of their health, and at sometimes if the person has been observing few symptoms and he/she is not sure about the disease he/she is encountered with so this will lead to various diseases in future. So, to avoid that and get to know the disease in early stages of the symptoms this disease prediction will help a lot to the various people’s ranging from children to teenagers to adults and also the senior citizens.

# 1.3 PROBLEM PURPOSE

The purpose of making this project called “Disease Prediction Using Machine Learning” is to predict the accurate disease of the patient using all their general information’s and also the symptoms. Using this information, there we will compare with our previous datasets of the patients and predicts the disease of the patient he/she is been through. If this Prediction is done at the early stages of the disease with the help of this project and all other necessary measure the disease can be cured and in general this prediction system can also be very useful in health industry. If health industry adopts this project then the work of the doctors can be reduced and they can easily predict the disease of the patient. 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 symptoms. The health industry in information yet and knowledge poor and this industry is very vast industry which has lot of work to be done. So, with the help of all those algorithms, techniques and methodologies we have done this project which will help the peoples who are in the need.

# LITERATURE SURVEY

Tom Mitchell states machine learning as “A computer program is said to learn from experience and from some tasks and some performance on, as measured by, improves with experience”. Machine Learning is combination of correlations and relationships, most machine learning algorithms in existence are concerned with finding and/or exploiting relationship between datasets. Once Machine Learning Algorithms can pinpoint on certain correlations, the model can either use these relationships to predict future observations or generalize the data to reveal interesting patterns. In Machine Learning there are various types of algorithms such as Regression, Linear Regression, Logistic Regression, Naive Bayes Classifier, Bayes theorem, KNN (K-Nearest Neighbor Classifier), Decision Tress, Entropy, ID3, SVM (Support Vector Machines), K-means Algorithm, Random Forest and etc.,

The name machine learning was coined in 1959 by Arthur Samuel. Machine learning explores the study and construction of algorithms that can learn from and make predictions on data Machine learning is closely related to (and often overlaps with) computational statistics, which also focuses on prediction-making through the use of computers. It has strong ties to mathematical optimization, which delivers methods, theory and application domains to the field. Machine learning is sometimes conflated with data mining, where the latter subfield focuses more on exploratory data analysis and is known as unsupervised learning.

Within the field of data analytics, machine learning is a method used to devise complex models and algorithms that lend themselves to prediction; in commercial use, this is known as predictive analytics. These analytical models allow researchers, data scientists, engineers, and analysts to "produce reliable, repeatable decisions and results" and uncover "hidden insights" through learning from historical relationships and trends in the data.

* 1. **CHAPTER WISE SUMMARY**

1. INTRODUCTION

2. SYSTEM ANALYSIS

3. SYSTEM DESIGN

4. SYSTEM IMPLEMENTATION

5. CONCLUSION AND FUTURE SCOPE

**CHAPTER – 2**

**SYSTEM ANALYSIS**

**2.1 EXISTING SYSTEM**

Prediction using traditional methods and models involves various risk factors and it consists of various measures of algorithms such as datasets, programs and much more to add on. High-risk and Low-risk patient classification is done on the basis of the tests that are done in group. But these models are only valuable in clinical situations and not in big industry sector. So, to include the disease predictions in various health related industries, we have used the concepts of machine learning and supervised learning methods to build the predictions system.

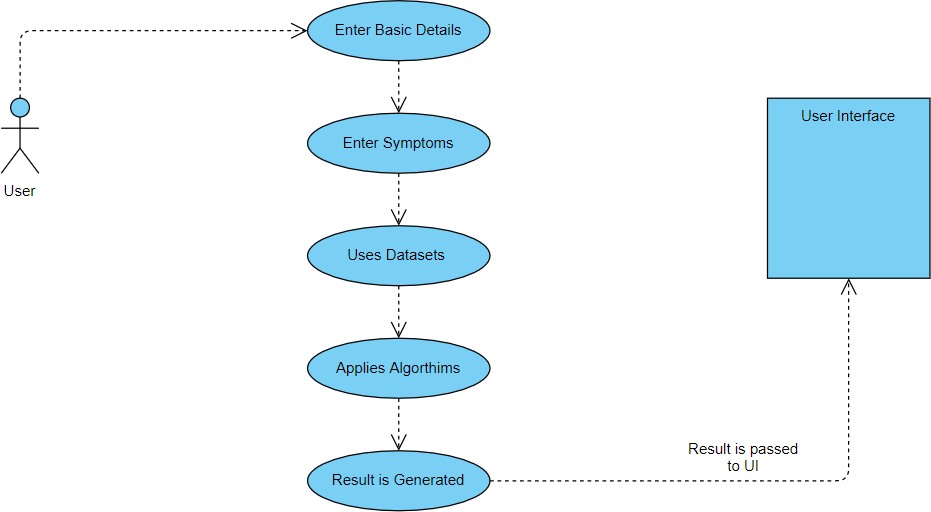
After doing the research and comparison of all the algorithms and theorems of machine learning we have come to conclusion that all those algorithms such as Decision Tree, KNN, Naïve Bayes, Regression and Random Forest Algorithm all are important in building a disease prediction system which predicts the disease of the patients from which he/she is suffering from and to do this we have used some performance measures like ROC, KAPPA Statistics, RMSE, MEA and various other tools. After using various techniques such as neural networks to make predictions of the diseases and after doing that we come to conclusion that it can predicts up to 90% accuracy rate after doing the experimentation and verifying the results. The information of patient statistics, results, disease history in recorded in EHR, which enables to identify the potential data centric solution, which reduces the cost of medical case studies. Existing system can predict the disease but not the sub type of the disease and it fails to predict the condition of the people, the predictions of disease have been indefinite and non-specific.

**2.2 PROPOSED SYSTEM**

The proposed system of disease prediction using machine learning is that we have used many techniques and algorithms and all other various tools to build a system which predicts the disease of the patient using the symptoms and by taking those symptoms we are comparing with the system’s dataset that is previously available. By taking those datasets and comparing with the patient’s disease we will predict the accurate percentage disease of the patient. The dataset and symptoms go to the prediction model of the system where the data is pre-processed for the future references and then the feature selection is done by the user where he will enter the various symptoms. Then the classification of those data is done with the help of various algorithms and techniques such as Decision Tree, KNN, Naïve Bayes, Random Forest anD etc. Then the data goes in the recommendation model, there it shows the risk analysis that is involved in the system and it also provides the probability estimation of the system such that it shows the various probability like how the system behaves when there are n number of predictions are done and it also does the recommendations for the patients from their final result and also from their symptoms like it can show what to use and what not to use from the given datasets and the final results. Here we have combined the overall structure and unstructured form of data for the overall risk analysis that is required for doing the prediction of the disease. Using the structured analysis, we can identify the chronic types of disease in a particular region and particular community. In unstructured analysis we select the features automatically with the help of algorithms and techniques. This system takes symptoms from the user and predicts the disease accordingly based on the symptoms that it takes and also from the previous datasets, it also helps in continuous evaluation of viral diseases, heart rate, blood pressure, sugar level and much more which is in the system and along with other external symptoms its predicts the appropriate and accurate disease.

**2.3 USECASE ANALYSIS**

The Use Case Analysis of the project disease prediction using machine learning consist of all the various aspects a normal use case diagram requires. This use case diagram shows how from starting the model flows from one step to another, like he enter into the system then enters all the information’s and all other general information along with the symptoms that goes into the system, compares with the prediction model and if true is predicts the appropriate results otherwise it shows the details where the user if gone wrong while entering the information’s and it also shows the appropriate precautionary measure for the user to follow. Here the use case diagram of all the entities are linked to each other where the user gets started with the system.

 **Fig 1.1 Use Case Analysis**

**2.4 REQUIREMENT SPECIFICATION**

**2.4.1 FUNCTIONAL REQUIREMENTS**

A Functional requirement defines a function of a system or its component. A function is described as a set of inputs, the behaviour, and outputs. Functional requirements may be calculations, technical details, data manipulation and processing and other specific functionality that define what a system is supposed to accomplish. Behavioural requirements describing all cases where the system uses the functional requirements are captured in use cases. Functional requirements are supported by non-functional requirements (also known as quality requirements), which impose constraints on the design or implementation (such as performance requirements, security, or reliability).

As defined in requirements engineering, functional requirements specify particular results of a system. This should be contrasted with non-functional requirements which specify overall characteristics such as cost and reliability. Functional requirements drive the application architecture of a system, while non-functional requirements drive the technical architecture of a system.

* Functional Requirements concerns with the specific functions delivered by the system.
* So, Functional requirements are statements of the services that the system must provide.
* The functional requirements of the system should be both complete and consistent
* Completeness means that all the services required by the user should be defined.
* Consistency means that requirements should not have any contradictory definitions.
* The requirements are usually described in a fairly abstract way. However, functional system requirements describe the system function in details, its inputs and outputs, exceptions and so on.
* Take user id and password match it with corresponding file entries. If a match is found then continue else raise an error message.

**2.4.2 NON-FUNCTIONAL REQUIREMENTS**

Non-functional Requirements refer to the constraints or restrictions on the system. They may relate to emergent system properties such as reliability, response time and store occupancy or the selection of language, platform, implementation techniques and tools.

The non-functional requirements can be built on the basis of needs of the user, budget constraints, organization policies and etc.

* + - Platform constraints**:** The main target is to generate an intelligent system to predict the adult height.
    - Accuracy and Precision: Requirements are accuracy and precision of the data
    - Modifiability: Requirements about the effort required to make changes in the software. Often, the measurement is personnel effort (person- months).
    - Portability: Since mobile phone is handy so it is portable and can be carried and used whenever required.
    - Reliability: Requirements about how often the software fails. The definition of a failure must be clear. Also, don't confuse reliability with availability which is quite a different kind of requirement. Be sure to specify the consequences of software failure, how to protect from failure, a strategy for error Prediction, and a strategy for correction.
    - Security: One or more requirements about protection of your system and its data.
    - Usability: Requirements about how difficult it will be to learn and operate the system.
    - The requirements are often expressed in learning time or similar metrics.
    1. **HARDWARE REQUIREMENTS**
* System : Pentium 4, Intel Core i3, i5, i7 and 2 GHz Minimum
* RAM : 512Mb or above
* Hard Disk : 10 GB or above
* Input Device : Keyboard and Mouse
* Output Device : Monitor or PC
  + 1. **SOFTWARE REQUIREMENTS**
* Operating System : Windows 7, 10 or Higher Versions
* Platform : Jupiter Notebook
* Front End : Python Tkinter
* Back End : Python and Files
* Programming Lang : Python
  + 1. **ACCESSIBILITY:**

Accessibility is a general term used to describe the degree to which a product, device, service, or environment is accessible by as many people as possible. In our project people who have registered with the cloud can access the cloud to store and retrieve their data with the help of a secret key sent to their email ids. User interface is simple and efficient and easy to use.

**CHAPTER – 3**

**SYSTEM DESIGN**

**3.1 DESIGN GOALS**

The Design goals consist of various design which we have implemented in our system disease prediction using machine learning. This system has built with various designs such as data flow diagram, sequence diagram, class diagram, use case diagram, component diagram, activity diagram, state chart diagram, deployment diagram. After doing these various diagrams and based on these diagrams we have done our project.

We have designed our system in such a way that whenever user log in into the system, the user has to register to the system, and new user cannot use the system without registering in the system. After that for registration the user requires basic credentials. Then the user has to login to the system using the same username and password. Here are the things that this system can perform.

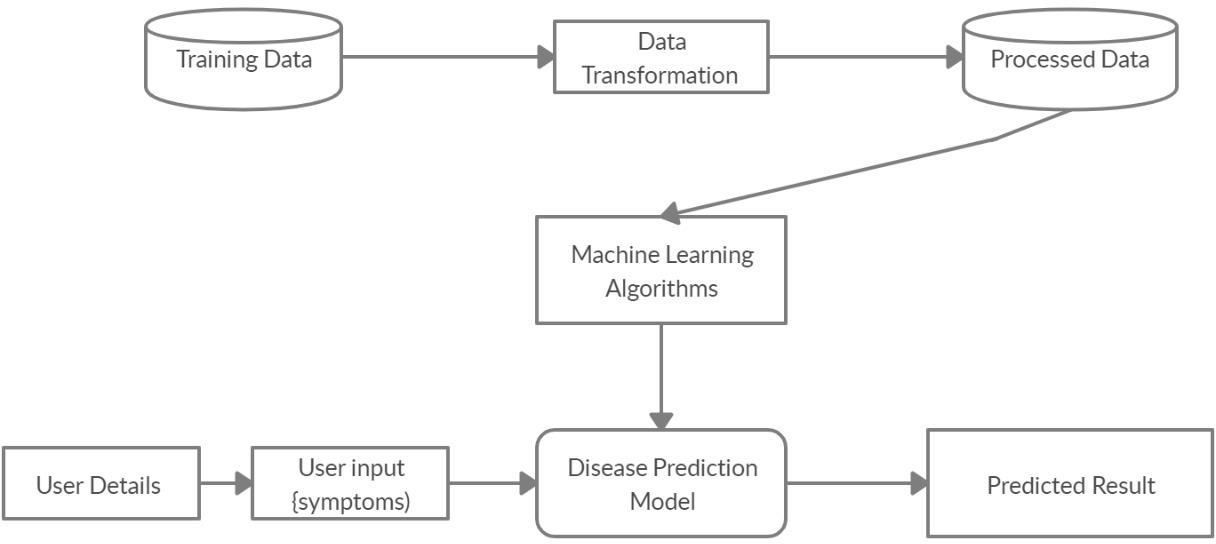
1. Entering Symptoms
2. Disease Prediction

**Entering Symptoms**: Once user successfully logged in to the system then he/she has to select the symptoms from the given drop-down menu.

**Disease prediction**: The predictive model predicts the disease of a person he might have, based on the user entered symptoms

**3.1.1 SYSTEM ARCHITECTURE**

Disease prediction using machine learning predicts the presence of the disease for the user based on various symptoms and the information the user gives such as sugar level, haemoglobin level and many more such general information through the symptoms. The architecture of the system disease prediction using machine learning consist of various datasets through which we will compare the symptoms of the user and predicts it, then the datasets are transformed into the smaller sets and from there it gets classified based on the classification algorithms later on the classified data is then processed into the machine learning technologies through which the data gets processed and goes in to the disease prediction model using all the inputs from the user that is mentioned above. Then after user entering the above information and overall processed data combines and compares in the prediction model of the system and finally predicts the disease. An architecture diagram is a graphical representation of a set of concepts, that are part of an architecture, including their principles, elements and components. The diagram explains about the system software in perception of overview of the system.

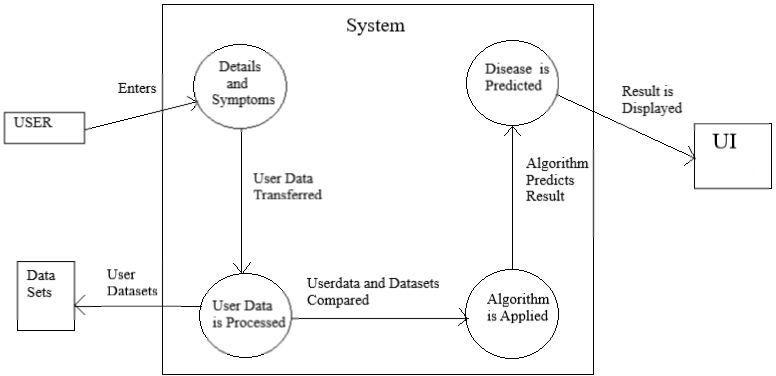


**Fig 1.2 System Architecture**

**3.2 DESIGN OF METHODLOGY**

**3.2.1 DATA FLOW DIAGRAM**

The dataflow diagram of the project disease prediction using machine learning consist of all the various aspects a normal flow diagram requires. This dataflow diagram shows how from starting the model flows from one step to another, like he enter into the system then enters all the information’s and all other general information along with the symptoms that goes into the system, compares with the prediction model and if true is predicts the appropriate results otherwise it shows the details where the user if gone wrong while entering the information.



**Fig 1.3 Data Flow Diagram**

**3.3 MODULES**

**3.3.1 DATASET**

For predicting diseases, the user needs to enter the details of patient and symptoms. A dataset is a collection of data in which data is arranged in some order. A dataset can contain any data from a series of an array to a database table. The key to success in the field of machine learning or to become a great data scientist is to practice with different types of datasets. But discovering a suitable dataset for each kind of machine learning project is a difficult task. So, in this topic, we will provide the details of the sources from which you can easily get the dataset according to your project.  To work with machine learning projects, we need a huge amount of data, because, without the data, one cannot train ML models. Collecting and preparing the dataset is one of the most crucial parts while creating an ML project. The technology applied to any ML project cannot work properly if the dataset is not well prepared and pre-processed.

**3.3.2 PREPROCESSING OF DATA**

Data preprocessing is a process of preparing the raw data and making it suitable for a machine learning model. It is the first and crucial step while creating a machine learning model. Data Preprocessingincludes the steps we need to follow to transform or encode data so that it may be easily  parsed by the machine. The main agenda for a model to be accurate and precise in predictions is that the  algorithm should be able to easily interpret the data's features.

**3.3.4 TRAINING AND TESTING DATA**

The Dataset will be then split into 80 percent of data for Training and 20 percent of data for Testing This type of data build up the machine learning algorithm. The data scientist feeds the algorithm input data,which corresponds to an expected output. The model evaluates the data repeatedly to learn more about the data’s behaviour and then adjusts itself to serve its intended purpose. After the model is built, testing data once again validates that it can make accurate predications. If training and validation data includes labels to monitor performance metrics of the model, the testing data should be unlabelled. Test data provides a final, real-world check of an unseen dataset to confirm that the ML algorithm was trained effectively..

**3.4 FRAMEWORK**

**TKINTER:**

Tkinter is a Python binding to the Tk GUI toolkit. It is the standard Python interface to the Tk GUI toolkit and is Python's de facto standard GUI. Tkinter is included with standard Linux, Microsoft Windows and Mac OS X installs of Python. The name Tkinter comes from Tk interface. Tkinter was written by Fredrik Lundh. Tkinter is free software released under a Python license.

As with most other modern Tk bindings, Tkinter is implemented as a Python wrapper around a complete Tool Command Language (TCL) interpreter embedded in the Python interpreter. Tkinter calls are translated into Tcl commands which are fed to this embedded interpreter, thus making it possible to mix Python and TCL in a single application. In Tkinter, the Frame widget is the basic unit of organization for complex layouts. A frame is a rectangular area that can contain other widgets. When any widget is created, a parent- child relationship is created. For example, if you place a text label inside a frame, the frame is the parent of the label.

Python offers multiple options for developing GUI (Graphical User Interface). Out of all the GUI methods, tkinter is most commonly used method. It is a standard Python interface to the Tk GUI toolkit shipped with Python. Python with tkinter outputs the fastest and easiest way to create the GUI applications.

**To create a Tkinter:**

Importing the module – tkinter Create the main window (container)

Add any number of widgets to the main window Apply the event Trigger on the widgets.

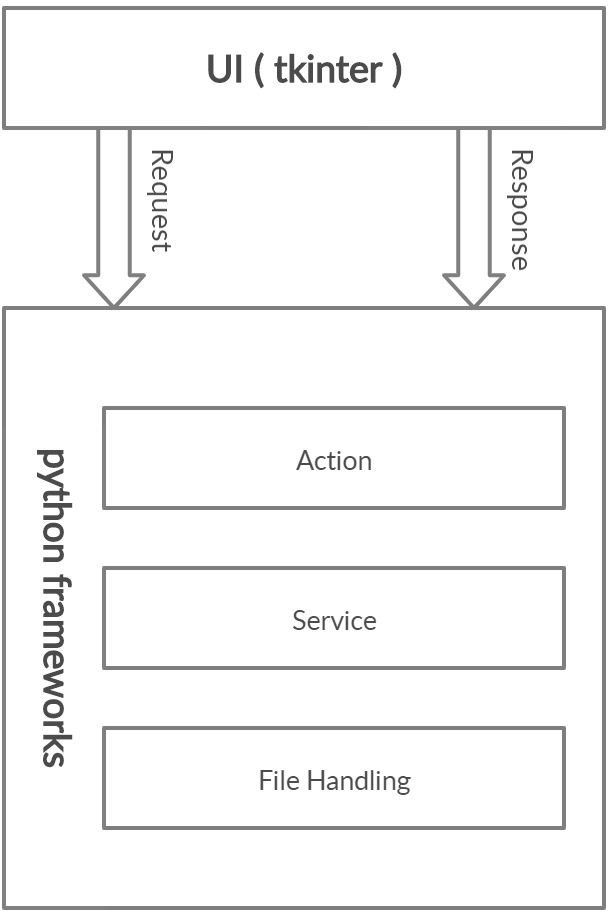
Importing tkinter is same as importing any other module in the python code.

Entering Details :

User can select the symptoms from the given drop-down menu.

Disease Prediction:

The Predictive model predicts the disease of a person he might have based on the user entered symptoms.



**Fig 1.4 Interface and Framework Diagram**

* + - After the User Interface it consist of the framework in which the system works accordingly using all the technologies, algorithms and various tools in which the project works accordingly.
    - The framework consists of all the modules starting from the data preparation, data building and assessment stage.
    - All these three factors are then going into the data collection phase, where the data is classified accordingly using the appropriate models and algorithms such as decision tree, naïve bayes, random forest.
    - Then all those algorithms use the datasets and it forms the sets where all the previous data is stored, then using that data it compares with the new data and result is generated.
    - Then pre-processing work will happen to reduce and analyze the data that is present in the system.
    - Then with the help of UI the data is transferred into the main screen.
    - Then later all those data are analyzed and validated then the final result is generated.
    - Finally after user enters the symptoms, all backend mechanisms works and the predicted result is displayed in the User Interface.

**CHAPTER – 4**

**IMPLEMENTATION**

**4.1 MODULE IMPLEMENTATION**

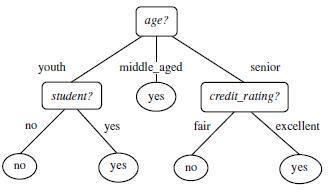
The project Disease Prediction using Machine Learning is developed to overcome general disease in earlier stages as we all know in competitive environment of economic development the mankind has involved so much that he/she is not concerned about health according to research there are 40% peoples how ignores about general disease which leads to harmful disease later. The Project “Disease Prediction using Machine Learning” is implemented using python completely. Even the interface of this project is done using python’s library interface called Tkinter. Here first the user needs to register into the system in order to use the prediction, user needs to register with username, email-id, phone, age and password. All these values are stored into the file system respectively, then user has option to move forward or leave, then user needs to login to the system using the username and password which he/she provided during the time of registration. If he/she enter incorrect username and correct password then the error message will prompt stating incorrect username and if he/she enters incorrect password and correct username then the error message will prompt stating incorrect password, so both username and password is necessary in order to login to the system. After logging in the user needs to the name and needs to select the symptoms from given drop-down menu, for more accurate result the user needs to enter all the given symptoms, then the system will provide the accurate result. This prediction is basically done with the help of 3 algorithms of machine learning such as Decision Tree, Random Forest and Naïve Bayes. When user enter all the symptoms then he needs to press the buttons of respective algorithm, for example there are 3 buttons for 3 algorithms, if user enters all symptoms and presses only Random forest’s button then the result will be provided only calculating using that algorithm, like this we have used 3 algorithms to provide more clear picture of the results and user needs to be satisfied with his predicted result.

**4.1.1 IMPORTING MODULES**

**ALOGARITHMS USED**

**DECISION TREE ALOGARITHM:**

Decision tree induction is the learning of decision trees from class-labelled training tuples. A decision tree is flowchart-like tree structure,



**Fig 1.5 Decision Tree problem**

* + - Decision tree induction is a non-parametric approach for building classification models.
    - Finding an optimal decision tree is an NP-complete problem
    - Techniques developed for constructing decision trees are computationally inexpensive, making it possible to construct models even when the training set size is very large.
    - Decision trees, especially smaller-sized trees, are relatively easy to interpret.
    - Decision tree provide an expressive representation for learning discrete- valued functions.
    - Decision tree algorithms are quite robust to the presence of noise, especially when methods for avoiding overfitting.
    - Decision tree induction is a non-parametric approach for building classification models.
    - Finding an optimal decision tree is an NP-complete problem
    - Techniques developed for constructing decision trees are computationally inexpensive, making it possible to construct models even when the training set size is very large.
    - Decision trees, especially smaller-sized trees, are relatively easy to interpret.
    - Decision tree provide an expressive representation for learning discrete- valued functions.
    - Decision tree algorithms are quite robust to the presence of noise, especially when methods for avoiding overfitting.

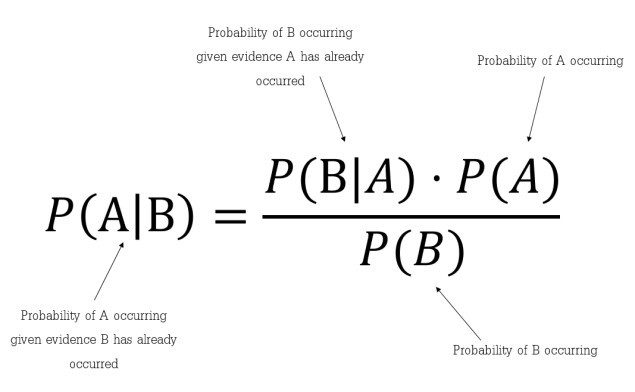
# 

**Fig 1.6 Decision tree Example**

Decision trees classify instances by sorting them down the tree the root to some leaf node, which provides the classification of the instance. An instance is classified by starting at the root node of the tree, testing the attribute specified by this node, then moving down the tree branch corresponding to the value of the attribute as shown in the above figure. This process is then repeated for the subtree rooted at the new node.

**NAIVE BAYES ALGORITHM:**

A classification technique based on Bayes' Theorem with an assumption of independence among predictors. It’s NAIVE as it assumes that features of a measurement are independent of each other.Bayes theorem-A way of calculating posterior probability P(c|x) from P(c), P(x) and P(x|c) P(c|x)=P(x|c)P(c)/P(x) Naive Bayes model is easy to build and particularly useful for very large data sets. Along with simplicity, Naive Bayes is known to outperform even highly sophisticated classification methods.



**Fig 1.7 Naïve Bayes formula**

* It classifies the class data based on the training set and the values in a classifying attribute and uses it in classifying new data.
* It is a two-step process Model Construction and Model Usage.
* This Bayes theorem is named after Thomas Bayes and it is statistical method for classification and supervised learning method.
* It can solve both categorical and continuous values attributes.
* Bayes theorem finds the probability of an event occurring given the probability of another event that has already occurred. Bayes theorem is stated mathematically as the following equation.

**4.2 TESTING**

|  |  |
| --- | --- |
| Serial Number of Test Case | TC-01 |
| Module Under Test | Prediction Result |
| Description | User needs to enter the name and symptoms to get the prediction result. |
| Input | Name and Symptoms |
| Output | If user enters all 5 correct symptoms then the accuracy will be high. If user enters only few symptoms then accuracy will be low. |
| Remarks | Test Successfull |

**Table 1.0 Testing**

**CHAPTER – 5**

**CONCLUSION AND FUTURE ENCHANCEMENT**

**5.1 CONCLUSION**

So, Finally I conclude by saying that, this project Disease prediction using machine learning is very much useful in everyone’s day to day life and it is mainly more important for the healthcare sector, because they are the one that daily uses these systems to predict the diseases of the patients based on their general information and there symptoms that they are been through. Now a day’s health industry plays major role in curing the diseases of the patients so this is also some kind of help for the health industry to tell the user and also it is useful for the user in case he/she doesn’t want to go to the hospital or any other clinics, so just by entering the symptoms and all other useful information the user can get to know the disease he/she is suffering from and the health industry can also get benefit from this system by just asking the symptoms from the user and entering in the system and in just few seconds they can tell the exact and up to some extent the accurate diseases. If health industry adopts this project then the work of the doctors can be reduced and they can easily predict the disease of the patient. The 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.

* 1. **FUTURE ENCHANCEMENT**
     + Facility for modifying user detail.
     + More interactive user interface.
     + Facilities for Backup creation.
     + Can be done as Web page.
     + Can be done as Mobile Application.
     + More Details and Latest Diseases

# Appendix A

**Sample Source Code:**

**Source code:**

from tkinter import \*

import numpy as np

import pandas as pd

# from gui\_stuff import \*

l1=['back\_pain','constipation','abdominal\_pain','diarrhoea','mild\_fever','yellow\_urine',

'yellowing\_of\_eyes','acute\_liver\_failure','fluid\_overload','swelling\_of\_stomach',

'swelled\_lymph\_nodes','malaise','blurred\_and\_distorted\_vision','phlegm','throat\_irritation',

'redness\_of\_eyes','sinus\_pressure','runny\_nose','congestion','chest\_pain','weakness\_in\_limbs',

'fast\_heart\_rate','pain\_during\_bowel\_movements','pain\_in\_anal\_region','bloody\_stool',

'irritation\_in\_anus','neck\_pain','dizziness','cramps','bruising','obesity','swollen\_legs',

'swollen\_blood\_vessels','puffy\_face\_and\_eyes','enlarged\_thyroid','brittle\_nails',

'swollen\_extremeties','excessive\_hunger','extra\_marital\_contacts','drying\_and\_tingling\_lips',

'slurred\_speech','knee\_pain','hip\_joint\_pain','muscle\_weakness','stiff\_neck','swelling\_joints',

'movement\_stiffness','spinning\_movements','loss\_of\_balance','unsteadiness',

'weakness\_of\_one\_body\_side','loss\_of\_smell','bladder\_discomfort','foul\_smell\_of urine',

'continuous\_feel\_of\_urine','passage\_of\_gases','internal\_itching','toxic\_look\_(typhos)',

'depression','irritability','muscle\_pain','altered\_sensorium','red\_spots\_over\_body','belly\_pain',

'abnormal\_menstruation','dischromic \_patches','watering\_from\_eyes','increased\_appetite','polyuria','family\_history','mucoid\_sputum',

'rusty\_sputum','lack\_of\_concentration','visual\_disturbances','receiving\_blood\_transfusion',

'receiving\_unsterile\_injections','coma','stomach\_bleeding','distention\_of\_abdomen',

'history\_of\_alcohol\_consumption','fluid\_overload','blood\_in\_sputum','prominent\_veins\_on\_calf',

'palpitations','painful\_walking','pus\_filled\_pimples','blackheads','scurring','skin\_peeling',

'silver\_like\_dusting','small\_dents\_in\_nails','inflammatory\_nails','blister','red\_sore\_around\_nose',

'yellow\_crust\_ooze']

disease=['Fungal infection','Allergy','GERD','Chronic cholestasis','Drug Reaction',

'Peptic ulcer diseae','AIDS','Diabetes','Gastroenteritis','Bronchial Asthma','Hypertension',

' Migraine','Cervical spondylosis',

'Paralysis (brain hemorrhage)','Jaundice','Malaria','Chicken pox','Dengue','Typhoid','hepatitis A',

'Hepatitis B','Hepatitis C','Hepatitis D','Hepatitis E','Alcoholic hepatitis','Tuberculosis',

'Common Cold','Pneumonia','Dimorphic hemmorhoids(piles)',

'Heartattack','Varicoseveins','Hypothyroidism','Hyperthyroidism','Hypoglycemia','Osteoarthristis',

'Arthritis','(vertigo) Paroymsal Positional Vertigo','Acne','Urinary tract infection','Psoriasis',

'Impetigo']

l2=[]

for x in range(0,len(l1)):

l2.append(0)

# TESTING DATA df -------------------------------------------------------------------------------------

df=pd.read\_csv("Training.csv")

df.replace({'prognosis':{'Fungal infection':0,'Allergy':1,'GERD':2,'Chronic cholestasis':3,'Drug Reaction':4,

'Peptic ulcer diseae':5,'AIDS':6,'Diabetes ':7,'Gastroenteritis':8,'Bronchial Asthma':9,'Hypertension ':10,

'Migraine':11,'Cervical spondylosis':12,

'Paralysis (brain hemorrhage)':13,'Jaundice':14,'Malaria':15,'Chicken pox':16,'Dengue':17,'Typhoid':18,'hepatitis A':19,

'Hepatitis B':20,'Hepatitis C':21,'Hepatitis D':22,'Hepatitis E':23,'Alcoholic hepatitis':24,'Tuberculosis':25,

'Common Cold':26,'Pneumonia':27,'Dimorphic hemmorhoids(piles)':28,'Heart attack':29,'Varicose veins':30,'Hypothyroidism':31,

'Hyperthyroidism':32,'Hypoglycemia':33,'Osteoarthristis':34,'Arthritis':35,

'(vertigo) Paroymsal Positional Vertigo':36,'Acne':37,'Urinary tract infection':38,'Psoriasis':39,

'Impetigo':40}},inplace=True)

# print(df.head())

X= df[l1]

y = df[["prognosis"]]

np.ravel(y)

# print(y)

# TRAINING DATA tr --------------------------------------------------------------------------------

tr=pd.read\_csv("Testing.csv")

tr.replace({'prognosis':{'Fungal infection':0,'Allergy':1,'GERD':2,'Chronic cholestasis':3,'Drug Reaction':4,

'Peptic ulcer diseae':5,'AIDS':6,'Diabetes ':7,'Gastroenteritis':8,'Bronchial Asthma':9,'Hypertension ':10,

'Migraine':11,'Cervical spondylosis':12,

'Paralysis (brain hemorrhage)':13,'Jaundice':14,'Malaria':15,'Chicken pox':16,'Dengue':17,'Typhoid':18,'hepatitis A':19,

'Hepatitis B':20,'Hepatitis C':21,'Hepatitis D':22,'Hepatitis E':23,'Alcoholic hepatitis':24,'Tuberculosis':25,

'Common Cold':26,'Pneumonia':27,'Dimorphic hemmorhoids(piles)':28,'Heart attack':29,'Varicose veins':30,'Hypothyroidism':31,

'Hyperthyroidism':32,'Hypoglycemia':33,'Osteoarthristis':34,'Arthritis':35,

'(vertigo) Paroymsal Positional Vertigo':36,'Acne':37,'Urinary tract infection':38,'Psoriasis':39,

'Impetigo':40}},inplace=True)

X\_test= tr[l1]

y\_test = tr[["prognosis"]]

np.ravel(y\_test)

# ------------------------------------------------------------------------------------------------------

def DecisionTree():

from sklearn import tree

clf3 = tree.DecisionTreeClassifier() # empty model of the decision tree

clf3 = clf3.fit(X,y)

# calculating accuracy-------------------------------------------------------------------

from sklearn.metrics import accuracy\_score

y\_pred=clf3.predict(X\_test)

print(accuracy\_score(y\_test, y\_pred))

print(accuracy\_score(y\_test, y\_pred,normalize=False))

# -----------------------------------------------------

psymptoms = [Symptom1.get(),Symptom2.get(),Symptom3.get(),Symptom4.get(),Symptom5.get()]

for k in range(0,len(l1)):

# print (k,)

for z in psymptoms:

if(z==l1[k]):

l2[k]=1

inputtest = [l2]

predict = clf3.predict(inputtest)

predicted=predict[0]

h='no'

for a in range(0,len(disease)):

if(predicted == a):

h='yes'

break

if (h=='yes'):

t1.delete("1.0", END)

t1.insert(END, disease[a])

else:

t1.delete("1.0", END)

t1.insert(END, "Not Found")

def NaiveBayes():

from sklearn.naive\_bayes import GaussianNB

gnb = GaussianNB()

gnb=gnb.fit(X,np.ravel(y))

# calculating accuracy-------------------------------------------------------------------

from sklearn.metrics import accuracy\_score

y\_pred=gnb.predict(X\_test)

print(accuracy\_score(y\_test, y\_pred))

print(accuracy\_score(y\_test, y\_pred,normalize=False))

# -----------------------------------------------------

psymptoms = [Symptom1.get(),Symptom2.get(),Symptom3.get(),Symptom4.get(),Symptom5.get()]

for k in range(0,len(l1)):

for z in psymptoms:

if(z==l1[k]):

l2[k]=1

inputtest = [l2]

predict = gnb.predict(inputtest)

predicted=predict[0]

h='no'

for a in range(0,len(disease)):

if(predicted == a):

h='yes'

breaK

if (h=='yes'):

t3.delete("1.0", END)

t3.insert(END, disease[a])

else:

t3.delete("1.0", END)

t3.insert(END, "Not Found")

# gui\_stuff------------------------------------------------------------------------------------

root = Tk()

root.configure(background='blue')

# entry variables

Symptom1 = StringVar()

Symptom1.set(None)

Symptom2 = StringVar()

Symptom2.set(None)

Symptom3 = StringVar()

Symptom3.set(None)

Symptom4 = StringVar()

Symptom4.set(None)

Symptom5 = StringVar()

Symptom5.set(None)

Name = StringVar()

# Heading

w2 = Label(root, justify=LEFT, text="Disease Prediction using Machine Learning", fg="white", bg="blue")

w2.config(font=("Elephant", 30))

w2.grid(row=1, column=0, columnspan=2, padx=100)

w2 = Label(root, justify=LEFT, text="", fg="white", bg="blue")

w2.config(font=("Aharoni", 30))

w2.grid(row=2, column=0, columnspan=2, padx=100)

# labels

NameLb = Label(root, text="Name of the Patient", fg="yellow", bg="black")

NameLb.grid(row=6, column=0, pady=15, sticky=W)

S1Lb = Label(root, text="Symptom 1", fg="yellow", bg="black")

S1Lb.grid(row=7, column=0, pady=10, sticky=W)

S2Lb = Label(root, text="Symptom 2", fg="yellow", bg="black")

S2Lb.grid(row=8, column=0, pady=10, sticky=W)

S3Lb = Label(root, text="Symptom 3", fg="yellow", bg="black")

S3Lb.grid(row=9, column=0, pady=10, sticky=W)

S4Lb = Label(root, text="Symptom 4", fg="yellow", bg="black")

S4Lb.grid(row=10, column=0, pady=10, sticky=W)

S5Lb = Label(root, text="Symptom 5", fg="yellow", bg="black")

S5Lb.grid(row=11, column=0, pady=10, sticky=W)

lrLb = Label(root, text="DecisionTree", fg="white", bg="red")

lrLb.grid(row=15, column=0, pady=10,sticky=W)

ranfLb = Label(root, text="NaiveBayes", fg="white", bg="red")

ranfLb.grid(row=19, column=0, pady=10, sticky=W)

# entries

OPTIONS = sorted(l1)

NameEn = Entry(root, textvariable=Name)

NameEn.grid(row=6, column=1)

S1En = OptionMenu(root, Symptom1,\*OPTIONS)

S1En.grid(row=7, column=1)

S2En = OptionMenu(root, Symptom2,\*OPTIONS)

S2En.grid(row=8, column=1)

S3En = OptionMenu(root, Symptom3,\*OPTIONS)

S3En.grid(row=9, column=1)

S4En = OptionMenu(root, Symptom4,\*OPTIONS)

S4En.grid(row=10, column=1)

S5En = OptionMenu(root, Symptom5,\*OPTIONS)

S5En.grid(row=11, column=1)

dst = Button(root, text="DecisionTree", command=DecisionTree,bg="green",fg="yellow")

dst.grid(row=8, column=3,padx=10)

lr = Button(root, text="NaiveBayes", command=NaiveBayes,bg="green",fg="yellow")

lr.grid(row=10, column=3,padx=10)

#textfileds

t1 = Text(root, height=1, width=40,bg="orange",fg="black")

t1.grid(row=15, column=1, padx=10)

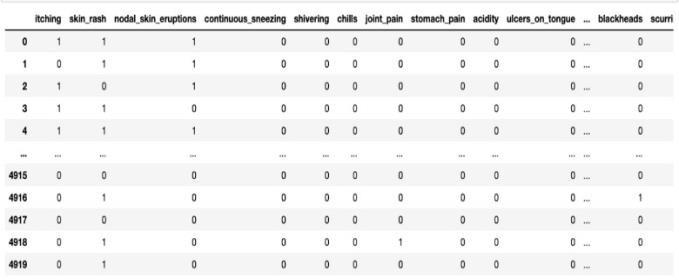
t3 = Text(root, height=1, width=40,bg="orange",fg="black")

t3.grid(row=19, column=1 , padx=10)

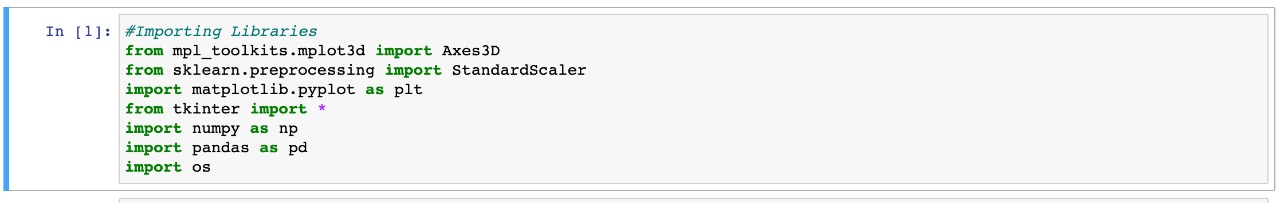
root.mainloop()

**Appendix B**

**SCREENSHOTS**



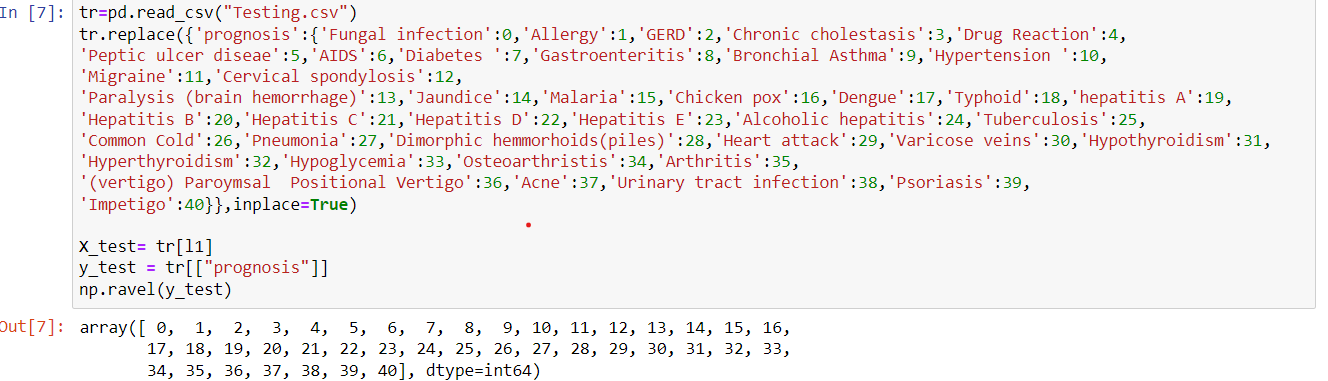
**Fig 1.8 sample picture of the dataset**

****

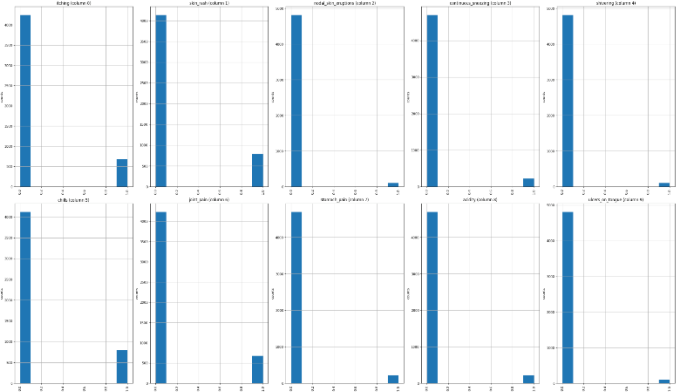
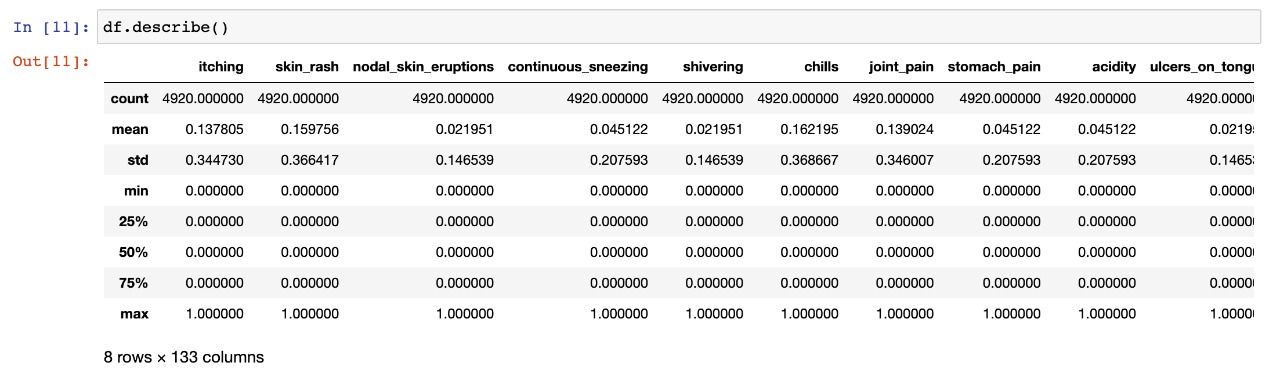
**Fig 1.9 Importing the libraries**

****

**Fig 2.0 Training the data**

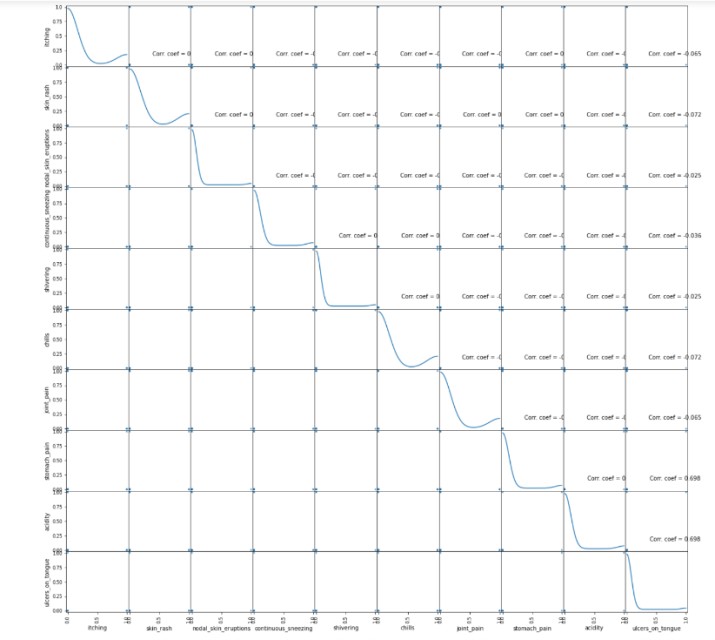
****

**Fig 2.1 Testing the data**

****

**Fig 2.2 Description of the data**

**Fig 2.3 Data Analysis Graph**



**Fig 2.4 Correlation graph**

**User page (input):**

****

**After entering symptoms:**

****

**Disease Prediction:**



**REFERENCES**

[1] Disease Prediction and Doctor Recommendation System by [www.irjet.net](http://www.irjet.net/)

[2] Disease Prediction Based on PriorKnowledge by www.hcup- us.ahrq.gov/nisoverview.jsp

[3] GDPS - General Disease Prediction System by [www.irjet.net](http://www.irjet.net/)

[4] Disease Prediction Using Machine Learning by International Research Journal of Engineering and Technology (IRJET).

[5] Kaveeshwar, S.A., and Cornwall, J., 2014, “The current state of disease mellitus in

India”. AMJ, 7(1), pp. 45-48.

[6] Dean, L., McEntyre, J., 2004, “The Genetic Landscape of Disease [Internet]. Bethesda (MD): National Center for Biotechnology Information (US); Chapter 1, Introduction to Disease. 2004 Jul 7.

[7] Machine Learning Methods Used in Disease by [www.wikipedia.com](http://www.wikipedia.com/)

[8] <https://www.researchgate.net/publication/325116774_disease_prediction_using_machine_learning_techniques>

[9] <https://ieeexplore.ieee.org/document/8819782>/disease\_prediction

[10] Algorithms Details from [www.dataspirant.com](http://www.dataspirant.com/)

[11] [https:/ en.wikipedia.org /machine\_learning\_algorithms](https://www.wikipedia.com/machine-learning-algorithms)

[12] <https://en.wikipedia.org/wiki/Python_(programming_language)>

[13] <https://wiki.python.org/TkInter>

[14] <https://creately.com/lp/uml-diagram-tool/>