DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING

A

PROJECT REPORT

ON

“DISEASE PREDICTION USING MACHINE LEARNING “

Submitted in partial fulfillment for the award of the degree of

BACHELOR OF ENGINEERING

IN

COMPUTER SCIENCE AND ENGINEERING

BY

Divyansh Arora
Siddharth Sharma
Shailabh Sharma
Anant Kumar

Under the guidance of

Mr. Yudhveer Singh Moudgil
, Dept. of CSE, RCE

CERTIFICATE

It is hereby certified that the project work entitled “**DISEASE PREDICTION USING MACHINE LEARNING”** is a bonafide work carried out by **Devyansh Arora** and **Siddharth Sharma** in partial fulfilment for the award of **Bachelor of Engineering** in **COMPUTER SCIENCE AND ENGINEERING** of the Roorkee college of engineering. It is certified that all corrections/suggestions indicated for Internal Assessment have been incorporated in the Report deposited in the departmental library. The project report has been approved as it satisfies the academic requirements in respect of project work prescribed for the said Degree.

Signature of Mentor
(Mr. Yudhveer Singh Moudgil)

Signature of HOD
(Mr. Yashveer Singh)

External Viva

Name of Examiner | **Signature with date**

1. …………………………………………….. | **…………………………………..**

2. …………………………………………… | **…………………………………..**

ABSTRACT

Disease 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 HTML/CSS Interface for it and also using the dataset that is available previously by the hospitals using that we will predict the disease.

.

ACKNOWLEDGEMENT

The satisfaction and euphoria that accompany the successful completion of any task would be impossible without the mention of the people who made it possible, whose constant guidance and encouragement crowned our efforts with success.

I have great pleasure in expressing my deep sense of gratitude to **CA S.K.Gupta**, Chairman of Roorkee College of Engineering for providing necessary infrastructure and creating good environment.

I would also like to thank **Mr.Yashveer Singh** Head, Department of Computer Science and Engineering, for her constant support.

I express my gratitude to**Mr.Yudhveer Singh Moudgil,** Senior Assistant Professor, my project guide, for constantly monitoring the development of the project and setting up precise deadlines. Her valuable suggestions were the motivating factors in completing the work.

Finally, a note of thanks to the teaching and non-teaching staff of Dept of Computer Science and Engineering, for their cooperation extended to me, and my friends, who helped me directly or indirectly in the course of the project work.

Devyansh Arora

Siddharth Sharma

CONTENTS

LIST OF FIGURES

INTRODUCTION

1.1 DISEASE PREDICTION

Disease Prediction using Machine Learning is a system which 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 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 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 DPUML is previously done by many other organizations but our intention is to make it different and beneficial for the users who are using this system. This Disease Prediction Using Machine Learning is completely done with the help of Machine Learning and Python Programming language with Interface for it and also using the dataset that is available previously by the hospitals using that we will predict the disease. Now a day's doctors are adopting many scientific technologies and methodology for both identification and diagnosing not only common disease, but also many fatal diseases. The successful treatment is always attributed by right and accurate diagnosis. Doctors may sometimes fail to take accurate decisions while diagnosing the disease of a patient, therefore disease prediction systems which use machine learning algorithms assist in such cases to get accurate results. 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 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. 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.

1.2 PROBLEM DEFINITION

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 PROJECT PURPOSE

The purpose of making this project called “Disease Prediction Using Machine Learning” is to predict the 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.

1.4 PROJECT FEATURES

The features of Disease Prediction Using Machine Learning are as follows.

- This Project will predict the diseases of the patients based on the symptoms.
- This is done based on the previous datasets of the hospitals so after comparing it can provide up to 80% of accurate results, and the project is still developing further to get the 100% accurate results.
- With the help of Disease prediction, it can predict the disease of the patient and can solve various problems and prevents from various aspects.

- The disease is predicted using the algorithms and the user has to enter the symptoms from the given drop-down menu, in order to get correct accuracy, the user has to enter all the symptoms.
- Here we can easily prepare the data and transform that data into algorithm, which will reduce the overall work of the project.
- To make user more application friendly rather than discussing with others for their disease.

CHAPTER 2

LITERATURE SURVEY

2.1 MACHINE LEARNING

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.

Machine learning tasks Machine learning tasks are typically classified into several broad categories:

Supervised learning: The computer is presented with example inputs and their desired outputs, given by a "teacher", and the goal is to learn a general rule that maps inputs to outputs. As special cases, the input signal can be only partially available, or restricted to special feedback.

Semi-supervised learning: The computer is given only an incomplete training signal: a training set with some (often many) of the target outputs missing.

Active learning: The computer can only obtain training labels for a limited set of instances (based on a budget), and also has to optimize its choice of objects to acquire labels for. When used interactively, these can be presented to the user for labelling.

Unsupervised learning: No labels are given to the learning algorithm, leaving it on its own to find structure in its input. Unsupervised learning can be a goal in itself (discovering hidden patterns in data) or a means towards an end (feature learning).

Reinforcement learning: Data (in form of rewards and punishments) are given only as feedback to the program's actions in a dynamic environment, such as driving a vehicle or playing a game against an opponent.

2.1.1 FEATURES OF MACHINE LEARNING

- It is nothing but automating the Automation.
- Getting computers to program themselves.
- Writing Software is bottleneck.
- Machine leaning models involves machines learning from data without the help of humans or any kind of human intervention.
- Machine Learning is the science of making of making the computers learn and act like humans by feeding data and information without being explicitly programmed.

- Machine Learning is totally different from traditionally programming, here data and output is given to the computer and in return it gives us the program which provides solution to the various problems. Below is the figure.

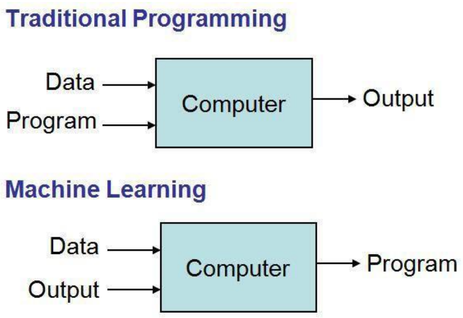


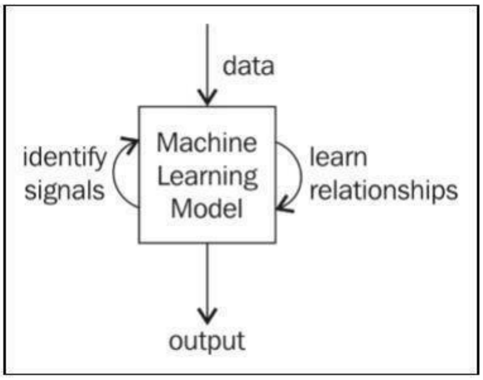
Fig 2.1.1 Traditional Programming vs Machine Learning

- Machine Learning is a combination of Algorithms, Datasets, and Programs.

- There are Many Algorithms in Machine Learning through which we will provide us the exact solution in predicting the disease of the patients.

- How Does Machine Learning Works?

- Solution to the above question is Machine learning works by taking in data, finding relationships within that data and then giving the output.



An overview of machine learning models

Fig 2.1.2 Machine Learning Model

- There are various applications in which machine learning is implemented such as Web search, computing biology, finance, e-commerce, space exploration, robotics, social networks, debugging and much more.
- There are 3 types of machine learning supervised, unsupervised, and reinforcement.

2.2 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.

2.3 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 and it predicts the appropriate and accurate disease.

2.4 SOFTWARE DESCRIPTION

2.4.1 PYTHON

Python is a multi-paradigm programming language. Object-oriented programming and structured programming are fully supported, and many of its features support functional programming and aspect-oriented programming (including by metaprogramming and meta-objects. Many other paradigms are supported via extensions, including design by contract and logic programming. Python uses dynamic typing and a combination of reference counting and a cycle-detecting garbage collector for memory management. It also features dynamic name resolution (late binding), which binds method and variable names during program execution.

Python's developers strive to avoid premature optimization, and reject patches to non-critical parts of CPython that would offer marginal increases in speed at the cost of clarity. When speed is important, a Python programmer can move time-critical functions to extension modules written in languages such as C, or use PyPy, a just-in-time compiler. Cython is also available, which translates a Python script into C and makes direct C-level API calls into the Python interpreter. An important goal of Python's developers is keeping it fun to use. Python's design offers some support for functional programming in the Lisp tradition. It has filter, map, and reduce functions, list comprehensions, dictionaries, sets, and generator expressions. The standard library has two modules (itertools and functools) that implement functional tools borrowed from Haskell and Standard ML.

2.4.2 BENEFITS OF PYTHON

- Presence of Third-Party Modules
- Extensive Support Libraries
- Open Source and Community Development
- Learning Ease and Support Available
- User-friendly Data Structures
- Productivity and Speed
- Highly Extensible and Easily Readable Language.

2.4.3 INTERFACE USING FLASK AND HTML/CSS

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

FEATURES OF FLASK

- Development server and debugger
- Integrated support for unit testing
- RESTful request dispatching
- Uses Jinja templating
- Support for secure cookies (client side sessions)
- 100% WSGI 1.0 compliant
- Unicode-based
- Extensive documentation
- Google App Engine compatibility
- Extensions available to enhance features desired

CHAPTER 3

REQUIREMENT ANALYSIS

3.1 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.

3.2 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.

1. **Performance requirement:** All data entered shall be up to mark and no flaws shall be there for the performance to be 100%.
2. **Platform constraints:** The main target is to generate an intelligent system to predict the adult height.
3. **Accuracy and Precision**: Requirements are accuracy and precision of the data
4. **Modifiability:** Requirements about the effort required to make changes in the software. Often, the measurement is personnel effort (person- months).
5. **Portability:** Since mobile phone is handy so it is portable and can be carried and used whenever required.
6. **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.

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.

MAINTAINABILITY:

In software engineering, maintainability is the ease with which a software product can be modified in order to include new functionalities can be added in the project based on the user requirements just by adding the appropriate files to existing project using .net and programming languages. Since the programming is very simple, it is easier to find and correct the defects and to make the changes in the project.

SCALABILITY:

System is capable of handling increase total throughput under an increased load when resources (typically hardware) are added. System can work normally under situations such as low bandwidth and large number of users.

PORTABILITY:

Portability is one of the key concepts of high-level programming. Portability is the software code base feature to be able to reuse the existing code instead of creating new code when moving software from an environment to another. Project can be executed under different operation conditions provided it meet its minimum configurations. Only system files and dependant assemblies would have to be configured in such case.

VALIDATION:

It is the process of checking that a software system meets specifications and that it fulfils its intended purpose. It may also be referred to as software quality control. It is normally the responsibility of software testers as part of the software development lifecycle. Software validation checks that the software product satisfies or fits the intended use (high-level checking), i.e., the software meets the user requirements, not as specification artefacts or as needs of those who will operate the software only; but, as the needs of all the stakeholders.

3.3 HARDWARE REQUIREMENTS

- System : Pentium 4, Intel Core i3, i5, i7 and 2 GHz Minimum
- RAM : 1GB or above
- Hard Disk : 20 GB or above
- Input Device : Keyboard and Mouse
- Output Device : Monitor or PC

3.4 SOFTWARE REQUIREMENTS

- Operating System : Windows 7, 10 or Higher Versions
- Platform : Jupiter Notebook
- Front End : HTML/CSS
- Back End : Python Flask
- Programming Lang : Python

CHAPTER 4

DESIGN

4.1 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.

Here are the things that this system can perform.

a. Entering Symptoms

b. 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.

.

4.2 SYSTEM ARCHITECTURE

Disease prediction using machine learning predicts the presence of the disease for the user based on various symptoms such as fever, cold etc. 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.

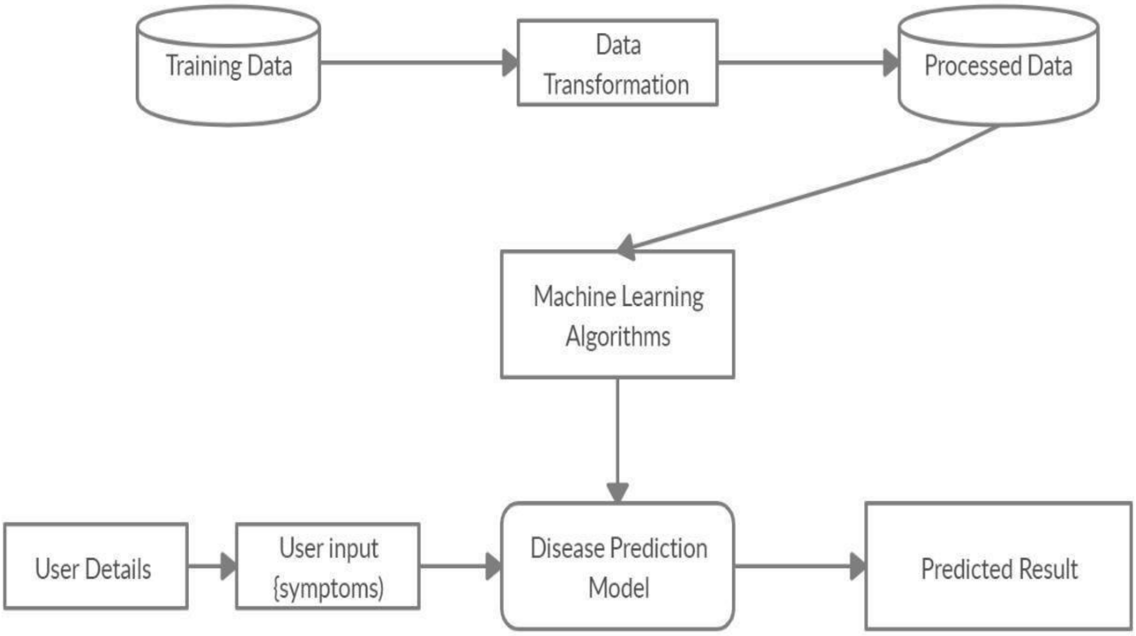


Fig 4.2 System Architecture

4.3 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.

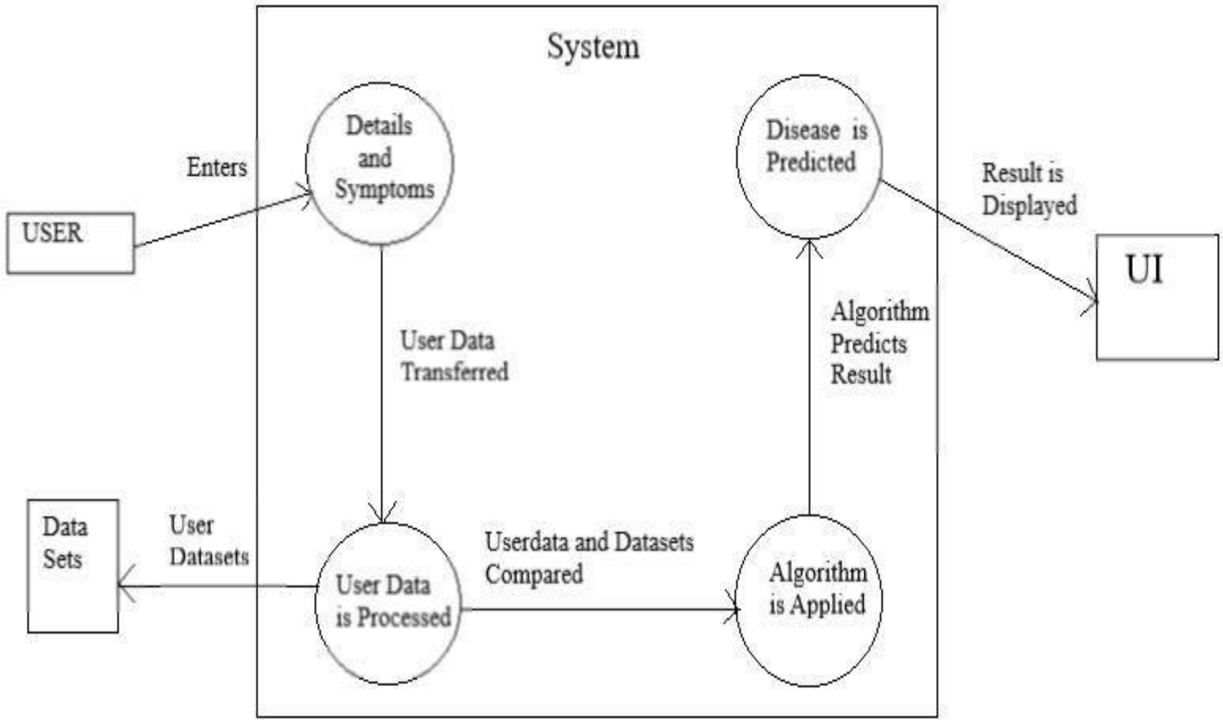


Fig 4.3 Data Flow Diagram

4.4 CLASS DIAGRAM

Disease prediction using machine learning consist of class diagram that all the other application that consists the basic class diagram, here the class diagram is the basic entity that is required in order to carry on with the project. Class diagram consist information about all the classes that is used and all the related datasets, and all the other necessary attributes and their relationships with other entities, all these information is necessary in order to use the concept of the prediction, where the user will enter all necessary information such as user name, email, phone number, and many more attributes that is required in order to login into the system and using the files concept we will store the information of the users who are registering into the system and retrieves those information later while logging into the system.

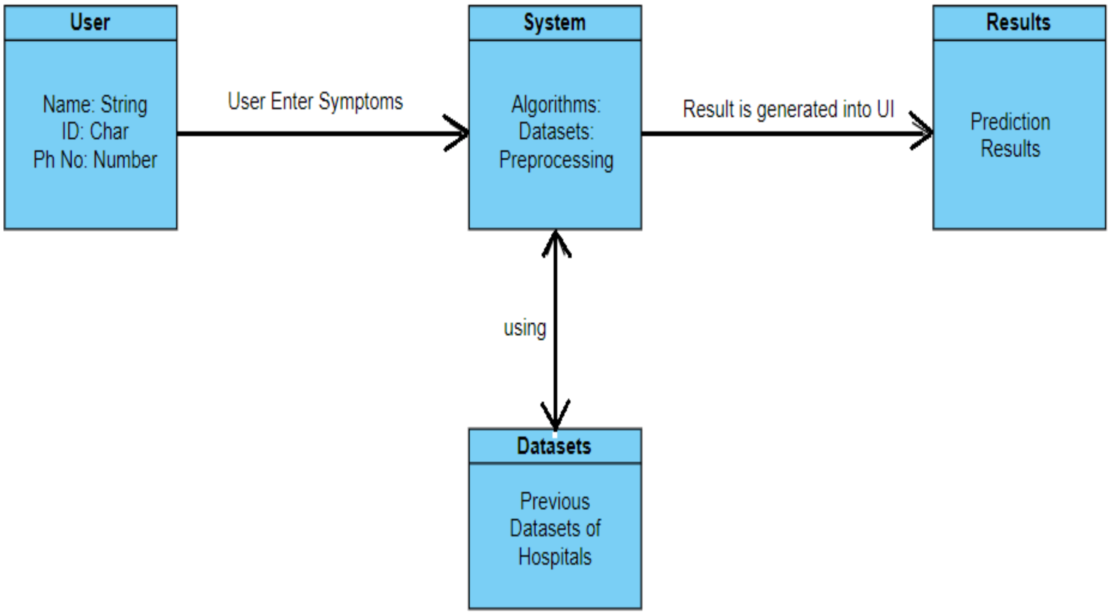


Fig 4.4 Class Diagram

4.5 SEQUENCE DIAGRAM

The Sequence diagram of the project disease prediction using machine learning consist of all the various aspects a normal sequence diagram requires. This sequence 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 sequence of all the entities are linked to each other where the user gets started with the system.

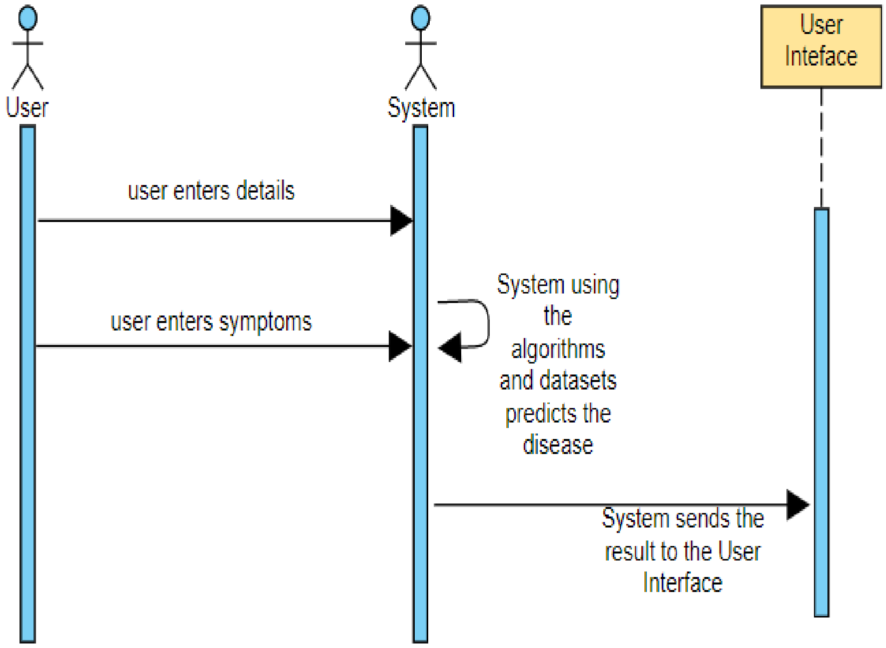

User
System
User Inteface
user enters details
user enters symptoms
System using the algorithms and datasets predicts the disease
System sends the result to the User Interface

Fig 4.5 Sequence Diagram

4.6 USE CASE DIAGRAM

The Use Case diagram 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.

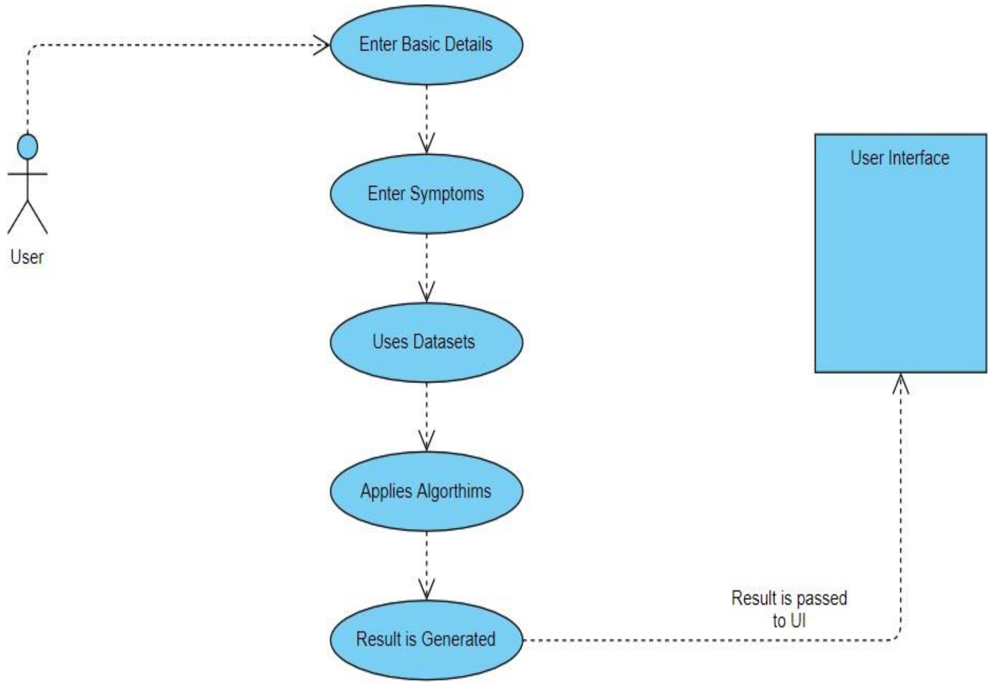


Fig 4.6 Use Case Diagram

4.7 ACTIVITY DIAGRAM

Activity diagram is another important diagram in UML to describe the dynamic aspects of the system. Activity diagram is basically a flowchart to represent the flow from one activity to another activity. The activity can be described as an operation of the system. The control flow is drawn from one operation to another. Here in this diagram the activity starts from user where the user registers into the system then login using the credentials and then the credentials are matched in the system and if its true, then the user proceeds to the prediction phase where the prediction happens. Then finally after processing the data from datasets the analysis will happen then the correct result will be displayed that is nothing but the Output.

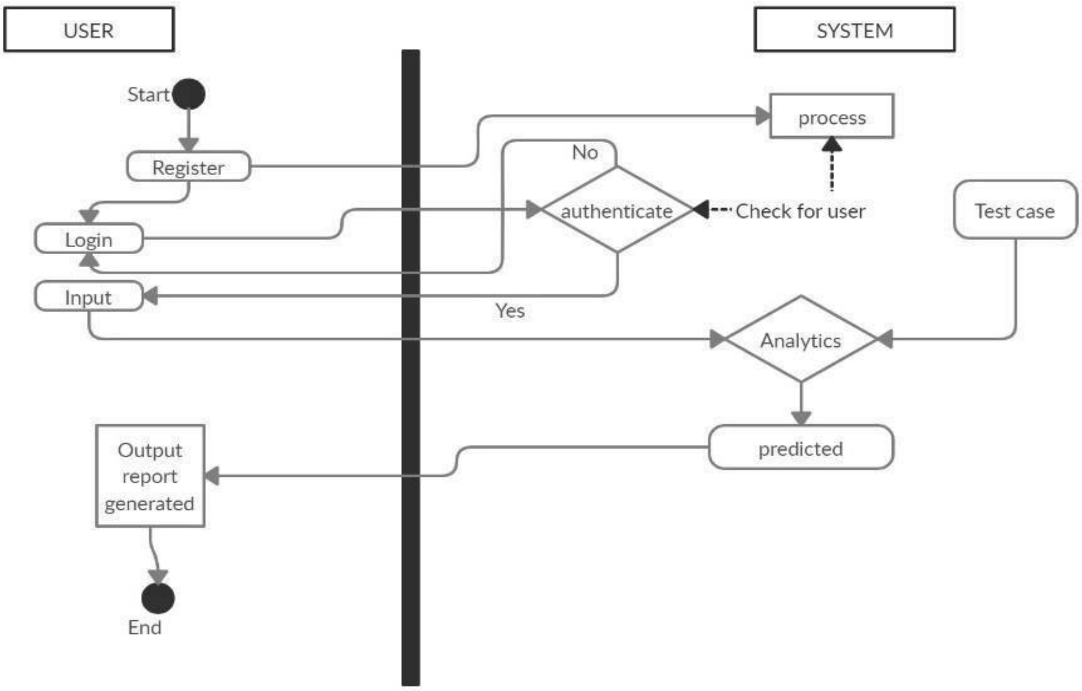


Fig 4.7 Activity Diagram

4.8 COMPONENT DIAGRAM

A component diagram, also known as a UML component diagram, describes the organization and wiring of the physical components in a system. Component diagrams are often drawn to help model implementation details and double-check that every aspect of the system's required function is covered by planned development. Here component diagram consists of all major components that is used to built a system. So, Design, Algorithm, File System and Datasets all are linked to one another. Datasets are used to compare the results and algorithm is used to process those results and give a correct accuracy and design UI is used to show the result in an appropriate way in the system and file system is used to store the user data. So, like this all components are interlinked to each other.

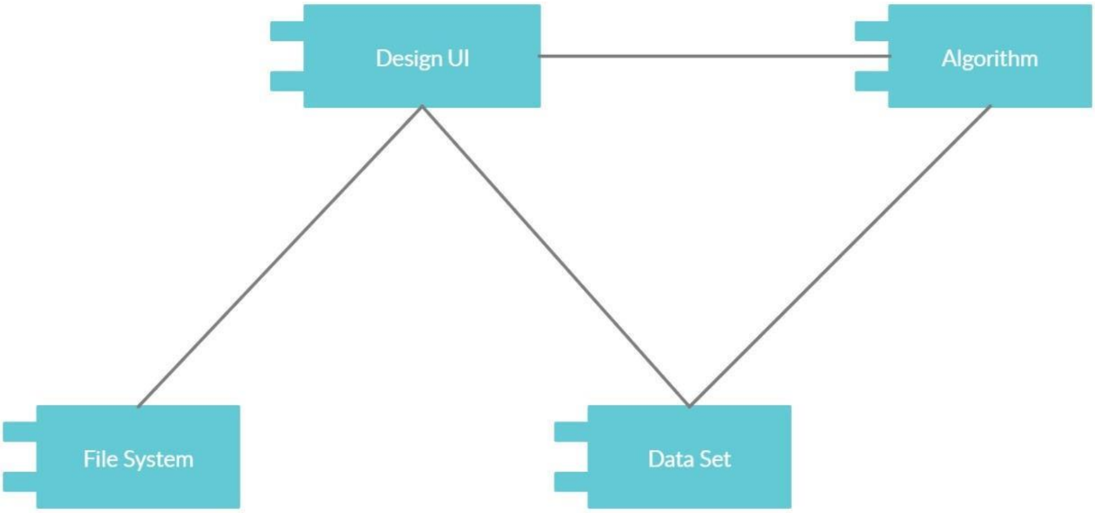


Fig 4.8 Component Diagram

4.9 STATE CHART DIAGRAM

A State chart diagram describes the behaviour of a single object in response to a series of events in a system. Sometimes it's also known as a Harel state chart or a state machine diagram. This UML diagram models the dynamic flow of control from state to state of a particular object within a system. It is similar to activity diagram but here there are only few rules like how it starts and how it end all are denoted with the help of the symbol given below, the system starts with the registration and then login comes, if the login is successful then it will go to the next step and if login is incorrect then comes back to same page stating incorrect details. After the successful login the user needs to enter the symptoms and then press the prediction button, at the same time the backend process will do their work and the correct result is predicted.

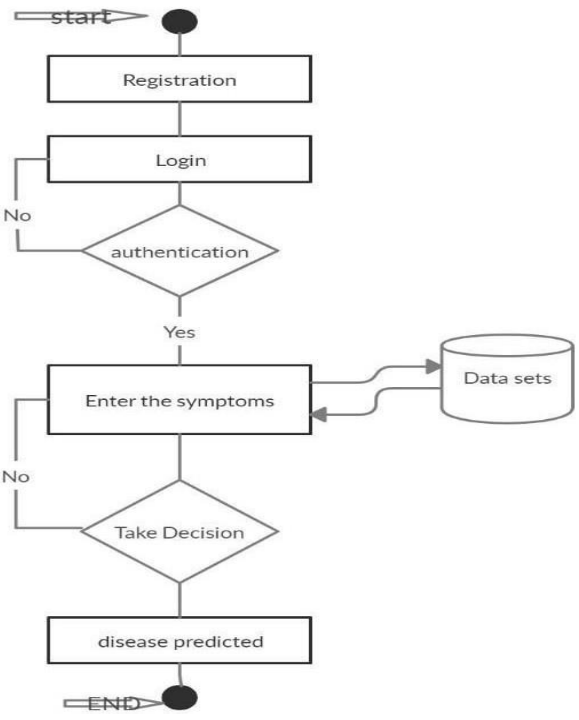


Fig 4.9 State Chart Diagram

4.10 COLLABORATION DIAGRAM

A collaboration diagram, also known as a communication diagram, is an illustration of the relationships and interactions among software objects in the Unified Modelling Language (UML). These diagrams can be used to portray the dynamic behaviour of a particular use case and define the role of each object. Here this diagram shows how all the models are connected to show the correct result starting from user, where he opens the system then using the system he does registration and that registration data is saved into file system and the using those data user logs in to the system and then he provides all the necessary information in order to get the accurate result, then system evaluates the user entered information and finally gives the correct result.

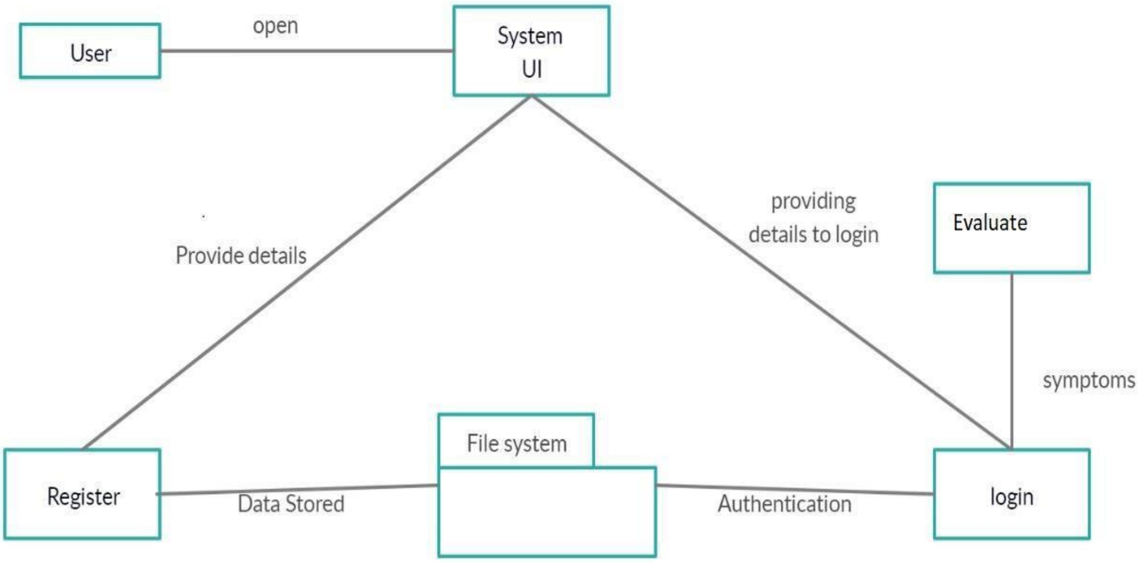


Fig 4.10 Collaboration Diagram

4.11 DEPLOYMENT DIAGRAM

A deployment diagram shows the configuration of run time processing nodes and the components that live on them. Deployment diagrams is a kind of structure diagram used in modelling the physical aspects of an object-oriented system. Here the deployment diagram show the final stage of the project and it also shows how the model looks like after doing all the processes and deploying in the machine. Starting from the system how it processes the user entered information and then comparing that information with the help of datasets, then training and testing those data using the algorithms such as decision tree, naïve Bayes, random forest. Then finally processing all those data and information the system gives the desired result in the interface.

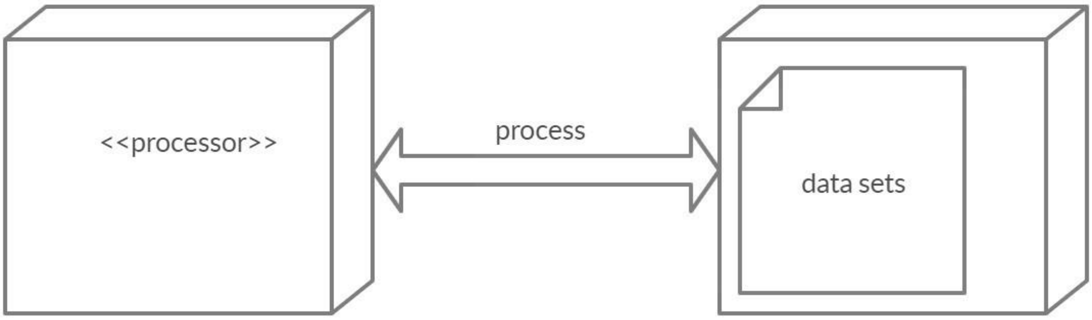


Fig 4.11 Deployment Diagram

4.12 INTERFACE AND FRAMEWORK DIAGRAM

- Below is the structure which we will use in our project Disease Prediction using Machine learning.
- The User Interface of this system consists of HTML/CSS with backend of Python's library Flask.
- Then it goes into the framework model where all the actions and services are combined and then the result is processed.
- It also consists of file system where all the user related information is stored such as name.
- Below is the structure of the User Interface along with necessary implementations.

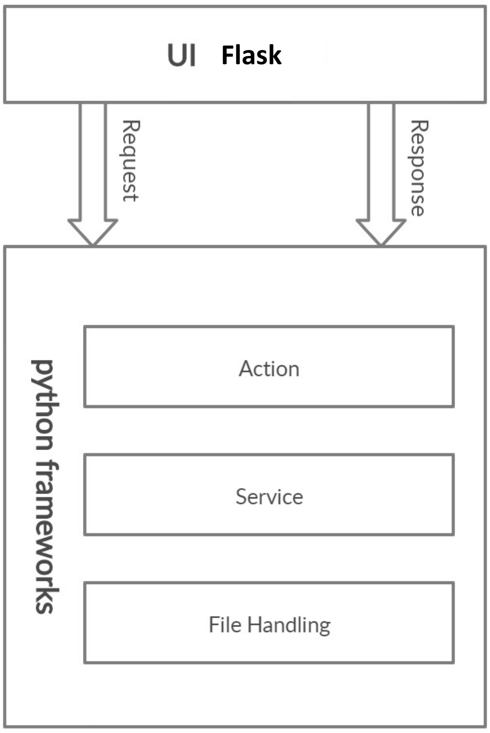


Fig 4.12 Interface and Framework Diagram

- 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 5

IMPLEMENTATION

5.1 OVERVIEW

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. 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 Random Forest. 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.

The project is designed user friendly and also secure to use ever user requires a authentication to enter into the system after which it provides the result based on the user input let me explain the complete implementation and working of project step wise below

- Once user open the system to login user needs to register by clicking on register/signup button
- After which user needs to provide some basic details of signup and then the details of user are saved in system
- Then user needs to login to have a checkup of his/her health
- When user tries to login if he provides wrong user name the system will provide a prompt message stating that the user is not found
- And if user tries to enter the wrong password the system will prompt stating that password is in correct hence the user needs to enter the correct user id and password to get in to the system
- After user enters the system user has to provide the symptoms which he/she is going through based on which we have several algorithms which predict the disease and also displays the percentage of accuracy
- The user needs to enter all the columns of symptoms to get the accurate result.
- Data collection and dataset preparation This will involve collection of medical information from various sources like hospitals, then pre-processing is applied on dataset which will remove all the unnecessary data and extract important features from data.
- Developing a probabilistic model and deep learning approach (RNN) for Disease Prediction in this step probabilistic model and deep learning approach based on RNN is to be developed it will run effectively on extensive databases of healthcare. And generate decision tree also it can deal with a huge number of information variables without variable deletion.
- Training and experimentation on datasets The Disease Prediction model will be trained on the dataset of diseases to do the prediction accurately and produce Confusion matrix. In this project 3 different algorithms were used –

- Decision Tree Algorithm
- Random Forest Algorithm
- Naïve Bayes Algorithm
- Deployment and analysis on real life scenario the trained and tested prediction model will be deployed in a real-life scenario made by the human experts & will be leveraged for further improvement in the methodology.
- The working and basic explanation of those 3 algorithms Random Forest, Decision Tree and Naïve Bayes is given below.

5.2 DECISION TREE ALGORITHM

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

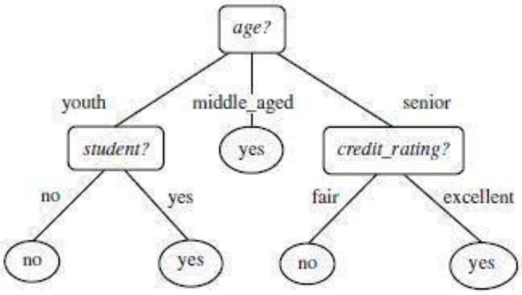
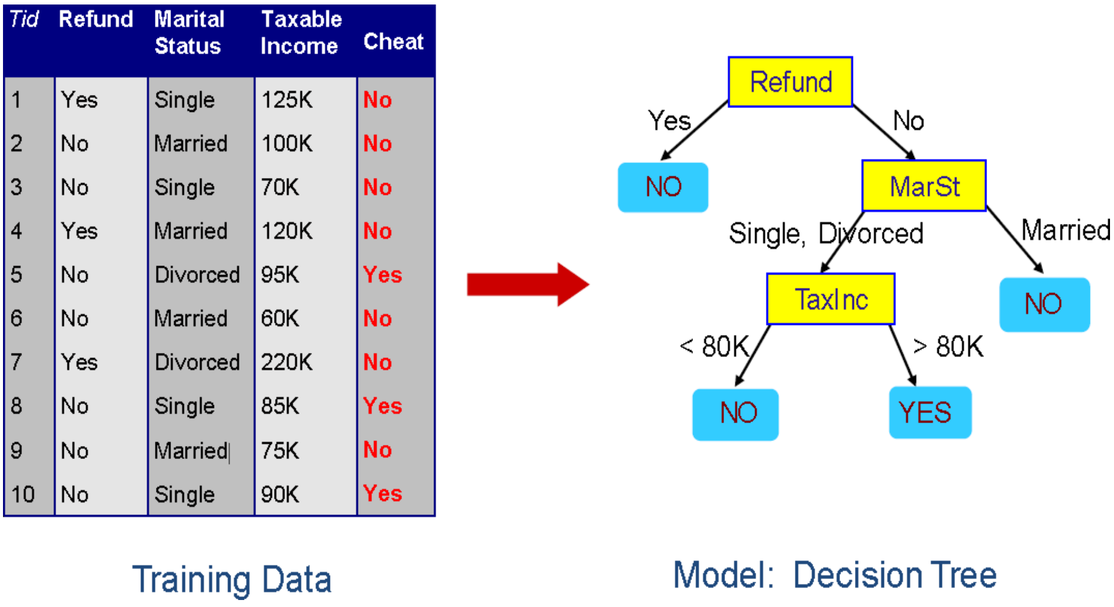


Fig 5.2.1 Decision Tree problem

- 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.



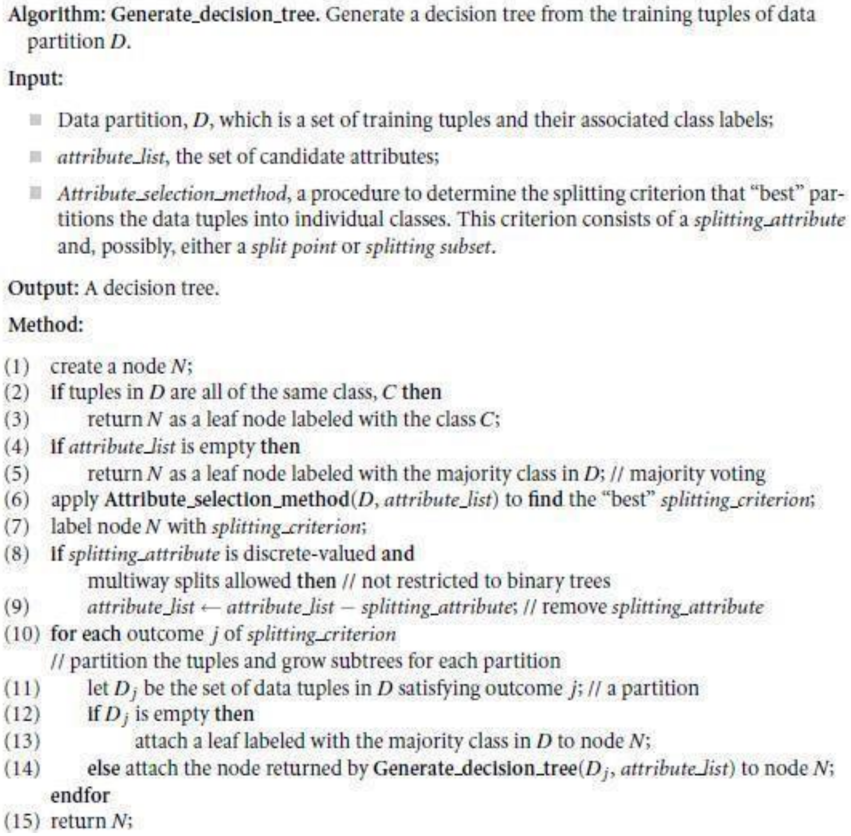
Tid	Refund	Marital Status	Taxable Income	Cheat
1	Yes	Single	125K	No
2	No	Married	100K	No
3	No	Single	70K	No
4	Yes	Married	120K	No
5	No	Divorced	95K	Yes
6	No	Married	60K	No
7	Yes	Divorced	220K	No
8	No	Single	85K	Yes
9	No	Married	75K	No
10	No	Single	90K	Yes

Training Data

Model: Decision Tree

Fig 5.2.2 Decision Tree Example

- The presence of redundant attributes does not adversely affect the accuracy of decision tree.
- The construction of decision tree classifiers does not require any domain knowledge or parameter setting, and therefore I appropriate for exploratory knowledge discovery. Decision trees can handle high dimensional data.
- Their representation of acquired knowledge in tree form is intuitive and generally easy to assimilate by humans.
- The learning and classification steps of decision tree induction are simple and fast.
- In general, decision tree classifiers have good accuracy.
- Decision tree induction algorithm shave been used for classification in many application areas, such as medicine, manufacturing and production, financial analysis, astronomy, and molecular biology.



Algorithm: Generate_decision_tree. Generate a decision tree from the training tuples of data partition *D*.

Input:

- Data partition, *D*, which is a set of training tuples and their associated class labels;
- *attribute_list*, the set of candidate attributes;
- *Attribute_selection_method*, a procedure to determine the splitting criterion that "best" partitions the data tuples into individual classes. This criterion consists of a *splitting_attribute* and, possibly, either a *split point* or *splitting subset*.

Output: A decision tree.

Method:

```
create a node N;
if tuples in D are all of the same class, C then
    return N as a leaf node labeled with the class C;
if attribute_list is empty then
    return N as a leaf node labeled with the majority class in D; // majority voting
apply Attribute_selection_method(D, attribute_list) to find the "best" splitting_criterion;
label node N with splitting_criterion;
if splitting_attribute is discrete-valued and
    multiway splits allowed then // not restricted to binary trees
    attribute_list ← attribute_list − splitting_attribute; // remove splitting_attribute
for each outcome j of splitting_criterion
// partition the tuples and grow subtrees for each partition
    let $D_j$ be the set of data tuples in D satisfying outcome j; // a partition
    if $D_j$ is empty then
        attach a leaf labeled with the majority class in D to node N;
    else attach the node returned by Generate_decision_tree($D_j$, attribute_list) to node N;
endfor
return N;
```

Fig 5.2.3 Decision Tree Algorithm

5.3 RANDOM FOREST ALGORITHM

• It is an ensemble classifier using many decision trees models; it can be used for regression as well as classification.

• Accuracy and variable importance information can be provided with the results.

• A random forest is the classifier consisting of a collection of tree structured classifiers k, where the k is independently, identically distributed random trees and each random tree consist of the unit of vote for classification of input.

• Random forest uses the Gini index for the classification and determining the final class in each tree.

• The final class of each tree is aggregated and voted by the weighted values to construct the final classifier.

• The working of random forest is, A random seed is chosen which pulls out at a random, a collection of samples from the training datasets while maintaining the class distribution.

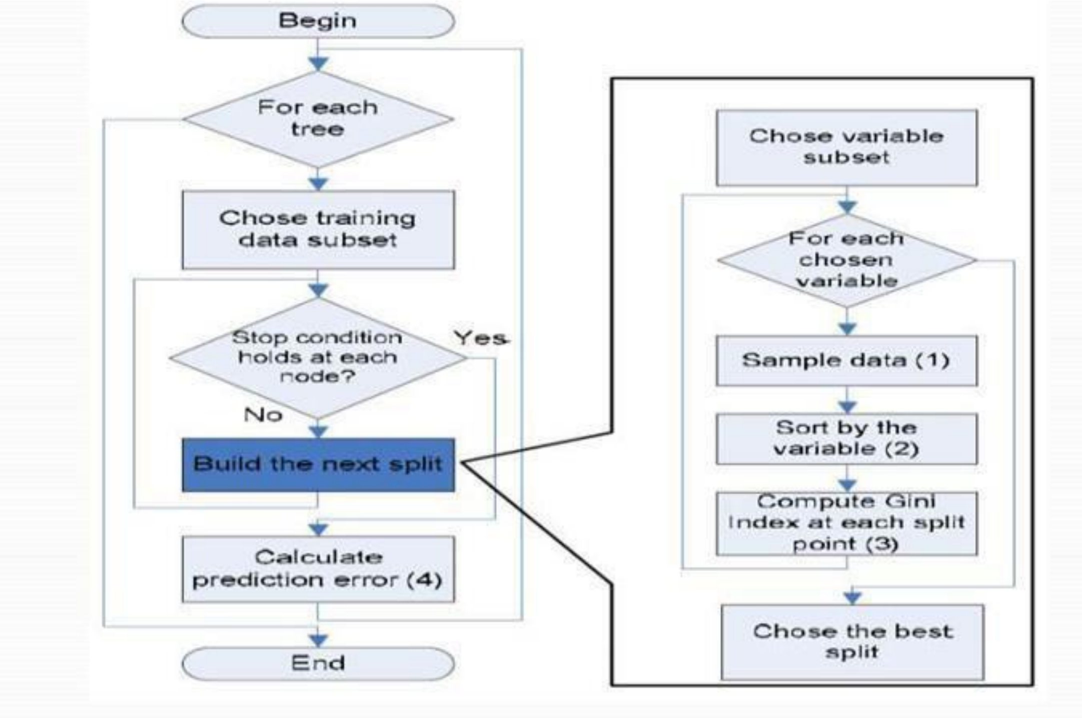


Fig 5.3 Random Forest Example

5.4 NAÏVE BAYES ALGORITHM

•It is used to predict the categorical class labels.

•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.

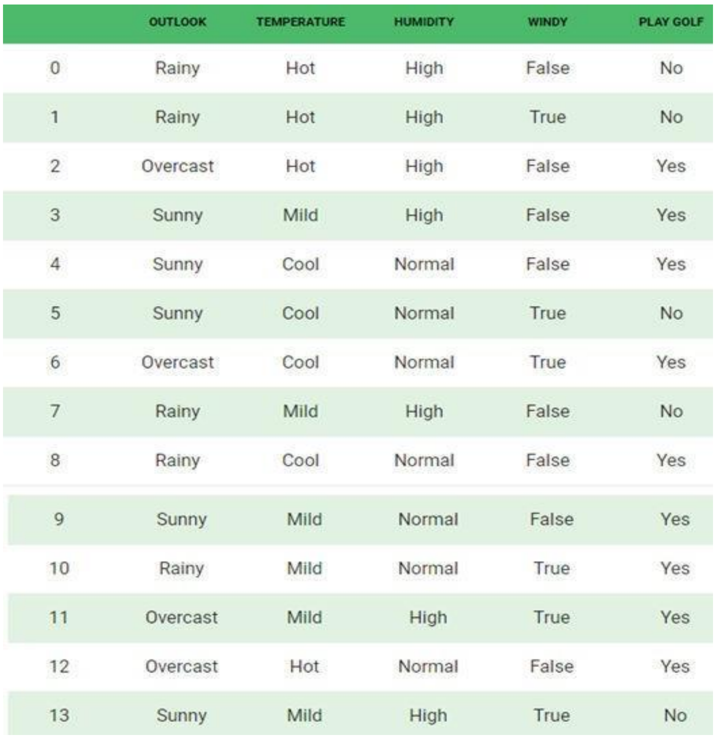
• This Bayes theorem is named after Thomas Bayes and it is statistical method for classification and supervised learning method.

• 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.

- P(A/B) = P(B|A) P(A)/P(B)

- Below is the example how this algorithm/theorem works with the dataset.



	OUTLOOK	TEMPERATURE	HUMIDITY	WINDY	PLAY GOLF
0	Rainy	Hot	High	False	No
1	Rainy	Hot	High	True	No
2	Overcast	Hot	High	False	Yes
3	Sunny	Mild	High	False	Yes
4	Sunny	Cool	Normal	False	Yes
5	Sunny	Cool	Normal	True	No
6	Overcast	Cool	Normal	True	Yes
7	Rainy	Mild	High	False	No
8	Rainy	Cool	Normal	False	Yes
9	Sunny	Mild	Normal	False	Yes
10	Rainy	Mild	Normal	True	Yes
11	Overcast	Mild	High	True	Yes
12	Overcast	Hot	Normal	False	Yes
13	Sunny	Mild	High	True	No

Fig 5.4 Naïve Bayes Dataset

- The given dataset is divided into two parts namely feature matrix and response vector.

- Feature matrix contains all the vectors means rows of the dataset in which each vector consists of the values of dependent features. In the above dataset features are outlook, temperature, humidity and windy.

- Response vector consist of values of class variables for each row of feature matrix. In the above dataset the class variable name is play golf.

- The fundamental naïve based assumption is that each feature makes an independent and equal contribution to the outcome.

CHAPTER 6

TESTING

TYPES OF TESTS

6.1 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.

6.2 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.

6.3 VALIDATION TESTING

An engineering validation test (EVT) is performed on first engineering prototypes, to ensure that the basic unit performs to design goals and specifications. It is important in identifying design problems, and solving them as early in the design cycle as possible, is the key to keeping projects on time and within budget. Too often, product design and performance problems are not detected until late in the product development cycle — when the product is ready to be shipped. The old adage holds true: It costs a penny to make a change in engineering, a dime in production and a dollar after a product is in the field.

Verification is a Quality control process that is used to evaluate whether or not a product, service, or system complies with regulations, specifications, or conditions imposed at the start of a development phase. Verification can be in development, scale-up, or production. This is often an internal process.

Validation is a Quality assurance process of establishing evidence that provides a high degree of assurance that a product, service, or system accomplishes its intended requirements. This often involves acceptance of fitness for purpose with end users and other product stakeholders.

The testing process overview is as follows:

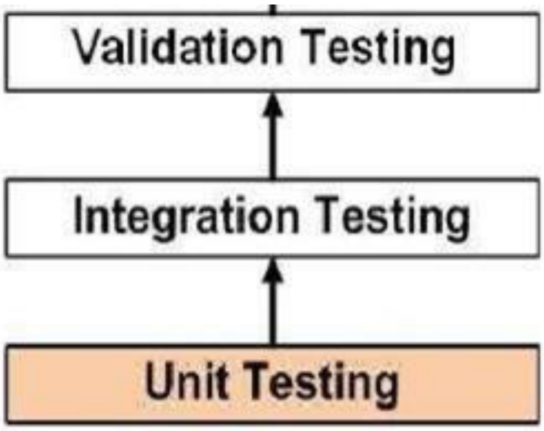


Fig 6.1 The Testing Process

6.4 SYSTEM TESTING

System testing of software or hardware is testing conducted on a complete, integrated system to evaluate the system's compliance with its specified requirements. System testing falls within the scope of black box testing, and as such, should require no knowledge of the inner design of the code or logic.

As a rule, system testing takes, as its input, all of the "integrated" software components that have successfully passed integration testing and also the software system itself integrated with any applicable hardware system. System testing is a more limited type of testing; it seeks to detect defects both within the "inter-assemblages" and also within the system as a whole. System testing is performed on the entire system in the context of a Functional Requirement Specification (FRS) or System Requirement Specification (SRS).

6.5 TESTING OF INITIALIZATION AND UICOMPONENTS

Serial Number of Test Case	TC 01
Description	A user enters their details for registering themselves to the System
Input	Details of Users such as username, email, phone, age, password.
Output	If the user's details are correct, user is registered. If the user's details are incorrect, Displays error message. If the user is already registered, Displays error message.
Remarks	Test Successful.

Table 6.5.1 Test Case for User Registration

Serial Number of Test Case	TC 02
Description	When the user tries to log in, details of user are verified in the system
Input	Username and Password.
Output	If the login details are correct, the user is logged in and user page is displayed. If the login details are incorrect, Displays error message.
Remarks	Test Successful.

Table 6.5.2 Test Case for User Login

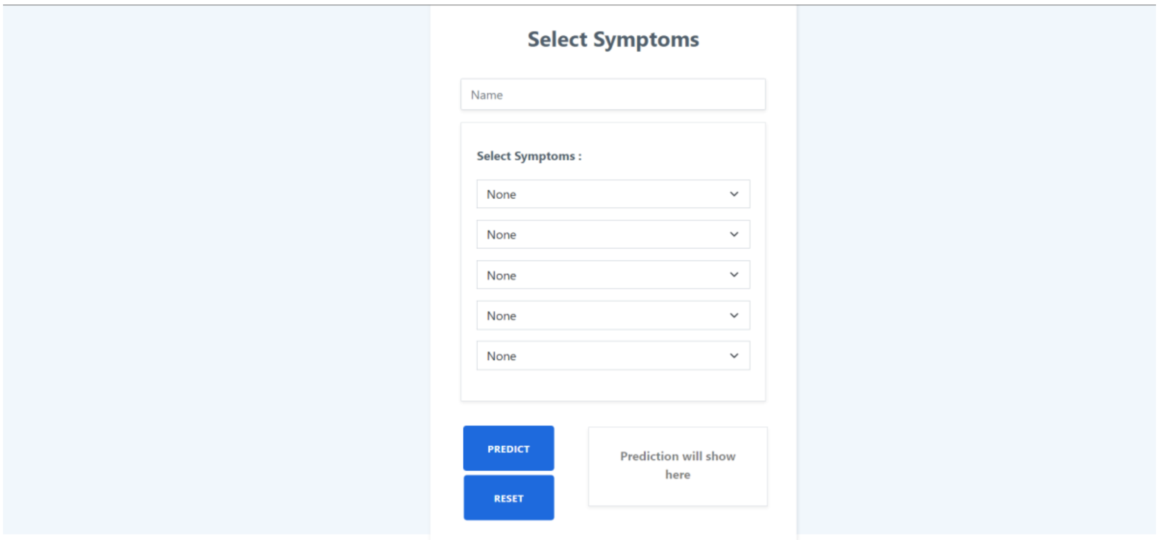
Serial Number of Test Case	TC 03
Module Under Test	Prediction Result
Description	User needs to enter the name and symptoms to get the prediction result.
Input	Name and Symptoms
Output	If user enters all 5 correct symptoms then the accuracy will be high. If user enters only few symptoms then accuracy will be low.
Remarks	Test Successful.

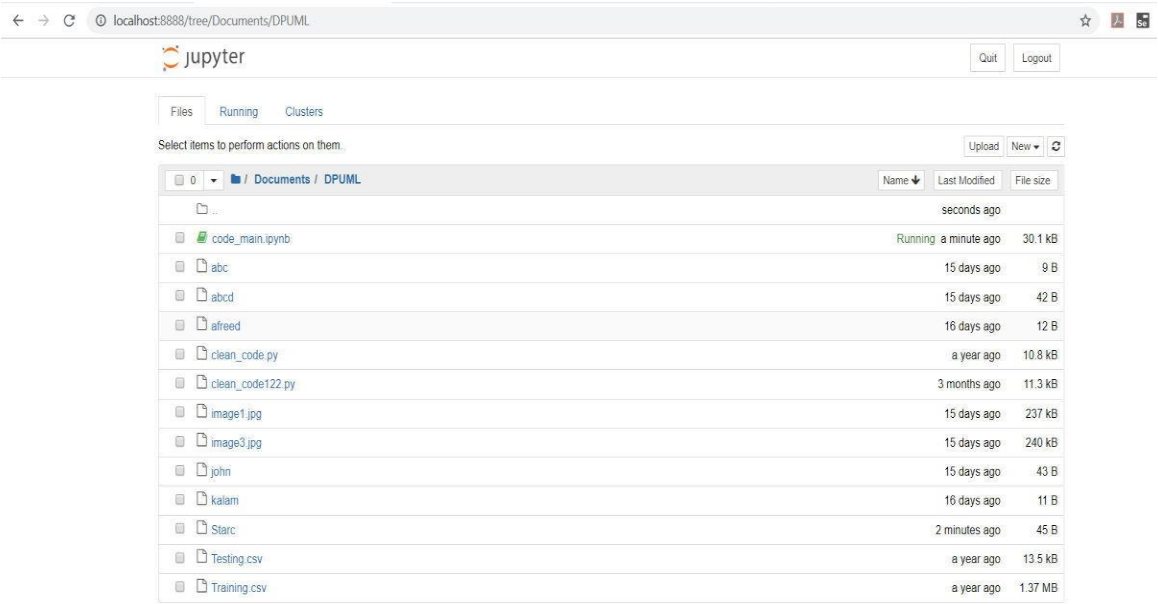
Table 6.5.3 Test Case for Prediction Result

CHAPTER 7

SNAPSHOTS

Fig 7.1 Main Page



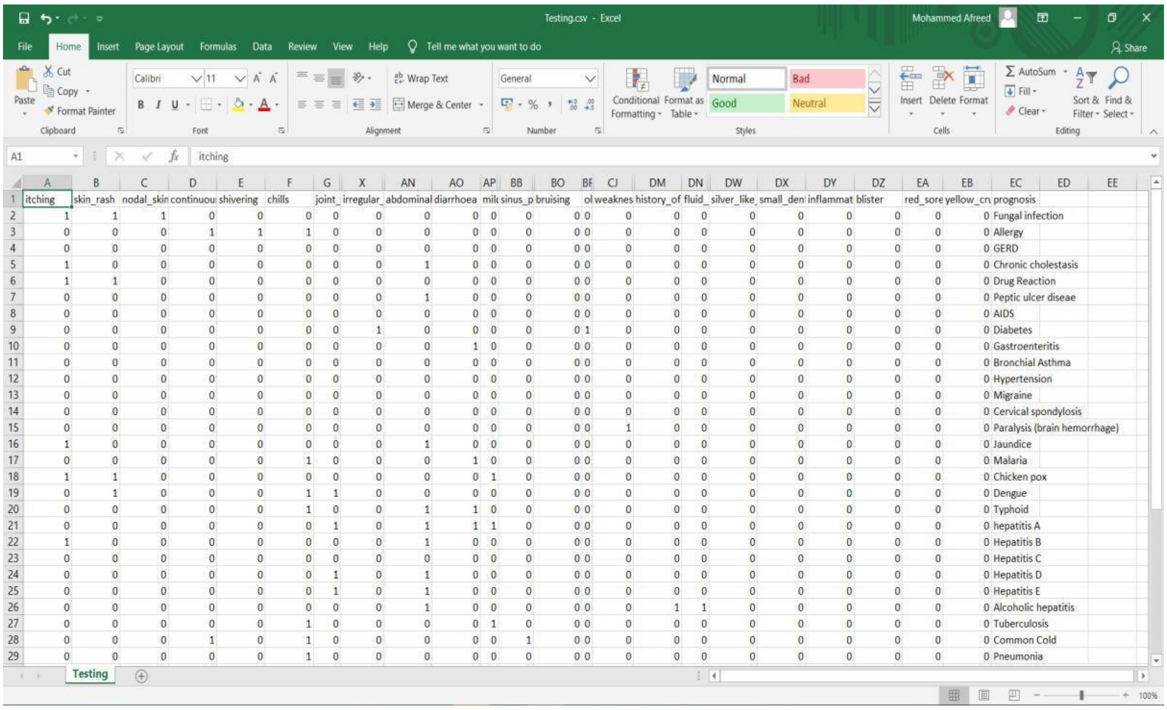


Select items to perform actions on them.

/ Documents / DPUML

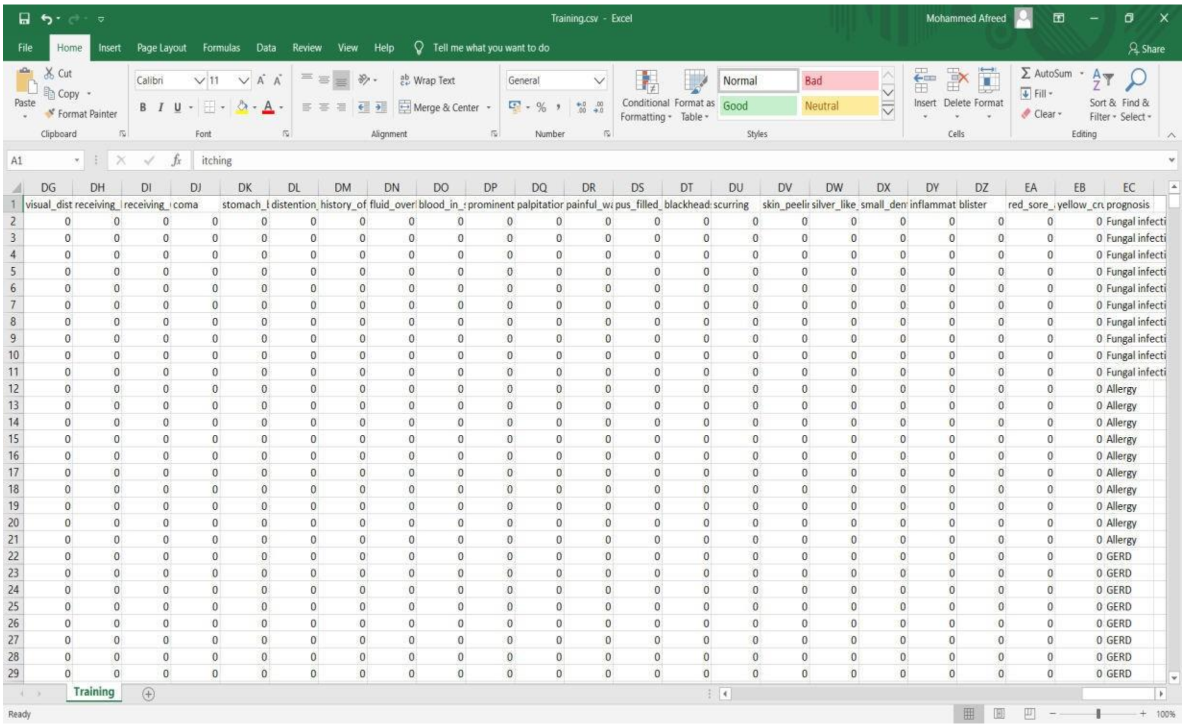
Name	Last Modified	File size
..	seconds ago	
code_main.ipynb	Running a minute ago	30.1 kB
abc	15 days ago	9 B
abcd	15 days ago	42 B
afreed	16 days ago	12 B
clean_code.py	a year ago	10.8 kB
clean_code122.py	3 months ago	11.3 kB
image1.jpg	15 days ago	237 kB
image3.jpg	15 days ago	240 kB
john	15 days ago	43 B
kalam	16 days ago	11 B
Starc	2 minutes ago	45 B
Testing.csv	a year ago	13.5 kB
Training.csv	a year ago	1.37 MB

Fig 7.8 Notebook



	A	B	C	D	E	F	G	X	AN	AO	AP	BB	BO	BF	CJ	DM	DN	DW	DX	DY	DZ	EA	EB	EC	ED	EE
1	itching	skin_rash	nodal_skin	continuou	shivering	chills	joint_	irregular_	abdominal	diarrhoea	mil	sinus_p	bruising	ol	weaknes	history_of	fluid_	silver_like_	small_den	inflammat	blister	red_sore	yellow_cru	prognosis		
2	1	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	Fungal infection		
3	0	0	0	1	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	Allergy		
4	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	GERD		
5	1	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	Chronic cholestasis		
6	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	Drug Reaction		
7	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	Peptic ulcer diseae		
8	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	AIDS		
9	0	0	0	0	0	0	0	1	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	Diabetes		
10	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	Gastroenteritis		
11	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	Bronchial Asthma		
12	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	Hypertension		
13	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	Migraine		
14	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	Cervical spondylosis		
15	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	Paralysis (brain hemorrhage)		
16	1	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	Jaundice		
17	0	0	0	0	0	1	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	Malaria		
18	1	1	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	Chicken pox		
19	0	1	0	0	0	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	Dengue		
20	0	0	0	0	0	1	0	0	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	Typhoid		
21	0	0	0	0	0	0	1	0	1	1	1	0	0	0	0	0	0	0	0	0	0	0	0	hepatitis A		
22	1	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	Hepatitis B		
23	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	Hepatitis C		
24	0	0	0	0	0	0	1	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	Hepatitis D		
25	0	0	0	0	0	0	1	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	Hepatitis E		
26	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	1	1	0	0	0	0	0	0	Alcoholic hepatitis		
27	0	0	0	0	0	1	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	Tuberculosis		
28	0	0	0	1	0	1	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	Common Cold		
29	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	Pneumonia		

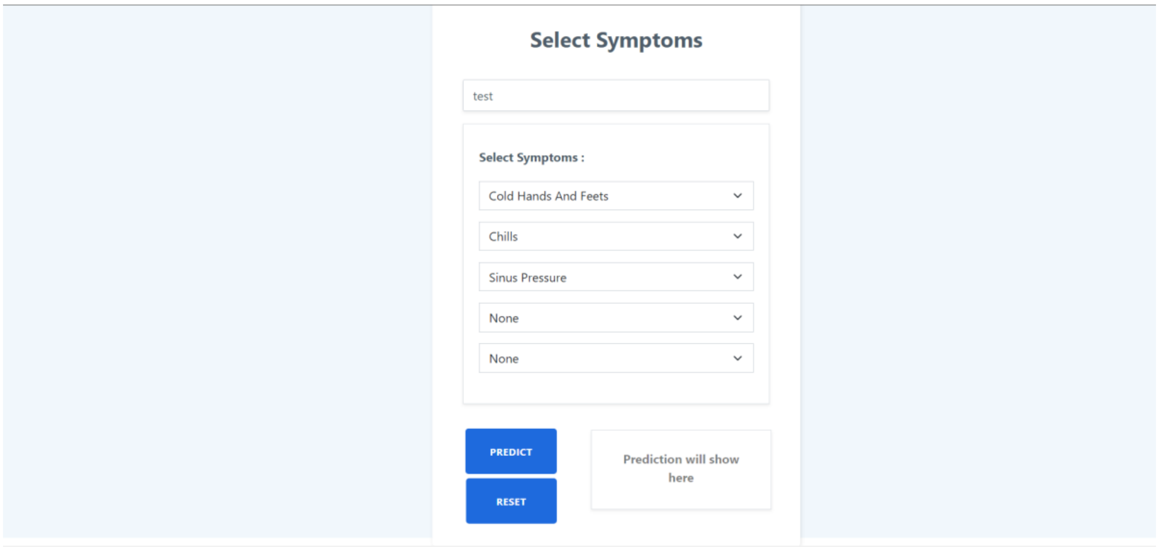
Fig 7.9 Tested Data



	DG	DH	DI	DJ	DK	DL	DM	DN	DO	DP	DQ	DR	DS	DT	DU	DV	DW	DX	DY	DZ	EA	EB	EC
1	visual_dist	receiving_	receiving_	coma	stomach_	distention_	history_of	fluid_over	blood_in_	prominent	palpitatior	painful_wa	pus_filled_	blackhead	scurring	skin_peelir	silver_like_	small_den	inflammat	blister	red_sore_	yellow_cru	prognosis
2	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	Fungal infecti
3	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	Fungal infecti
4	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	Fungal infecti
5	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	Fungal infecti
6	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	Fungal infecti
7	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	Fungal infecti
8	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	Fungal infecti
9	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	Fungal infecti
10	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	Fungal infecti
11	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	Fungal infecti
12	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	Allergy
13	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	Allergy
14	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	Allergy
15	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	Allergy
16	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	Allergy
17	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	Allergy
18	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	Allergy
19	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	Allergy
20	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	Allergy
21	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	Allergy
22	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	GERD
23	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	GERD
24	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	GERD
25	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	GERD
26	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	GERD
27	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	GERD
28	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	GERD
29	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	GERD

Fig 7.10 Trained Data

Fig 7.12 Predicted Result1 Page


Select Symptoms

test

Select Symptoms :

Cold Hands And Feets

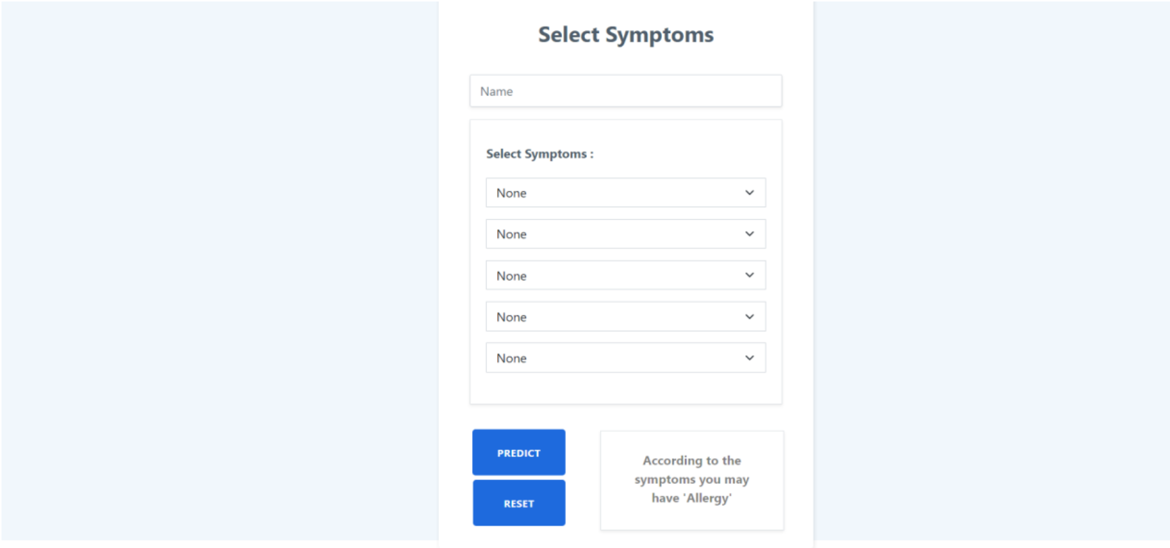
Chills

Sinus Pressure

None

None

Prediction will show here


Select Symptoms

Name

Select Symptoms :

None

None

None

None

None

According to the symptoms you may have 'Allergy'

Fig 7.13 Predicted Result2 Page

CHAPTER 8

CONCLUSION AND FUTURE ENHANCEMENT

8.1 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.

8.2 FUTURE ENHANCEMENT

- More interactive user interface.
- Facilities for Backup creation.
- Can be done as Mobile Application.
- More Details and Latest Diseases.

REFERENCES

1. Disease Prediction and Doctor Recommendation System by www.irjet.net

2. Disease Prediction Based on Prior Knowledge by www.hcup- us.ahrq.gov/nisoverview.jsp

3. GDPS - General Disease Prediction System by 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

8. https://www.researchgate.net/publication/325116774_disease_prediction_using_machine_learning_technique