Bachelor of Computer Applications

BCA 2651

Babu Banarasi Das University
Lucknow

Academic Session 2019 - 20

School of Computer Applications

I Floor, H-Block, BBDU, BBD City, Faizabad Road, Lucknow (U. P.) INDIA 226028
PHONE: HEAD: 0522-3911127, 3911321 **Dept. Adm. & Exam Cell:** 0522-3911326 **Dept. T&P Cell:** 0522-3911128; **E-Mail:** head.sca@gmail.com
www.bbdu.ac.in

PROJECT REPORT

on

Disease Prediction and Detection of Retinopathy using Machine Learning

for

Cetpa Infotech Pvt Ltd

towards partial fulfillment of the requirement
for the award of degree of

Bachelor of Computer Applications

from

Babu Banarasi Das University
Lucknow

Developed and Submitted by	**Under Guidance of**
Deepak Gupta	**Mr. Amit Kumar**
Km. Daranee Singh	
Mohd Faiz	

1170211106
1170211143
1170211172

Academic Session 2019 - 20

School of Computer Applications

I Floor, H-Block, BBDU,BBD City, Faizabad Road, Lucknow (U. P.) INDIA 226028
PHONE: HEAD: 0522-3911127,3911321Dept. Adm. & Exam Cell: 0522-3911326Dept. T&P Cell: 0522-3911128; E-Mail: head.sca@gmail.com

www.bbdu.ac.in

Babu Banarasi Das University
Lucknow

CERTIFICATE

This is to certify that Project Report entitled

Disease Prediction and Detection of Retinopathy in Prematurity using Machine Learning

being submitted by

Deepak Gupta
Km. Daranee Singh
Mohd Faiz

towards the partial fulfillment of the requirement for the award of the degree of

Bachelor of Computer Applications

to

Babu Banarasi Das University
Lucknow

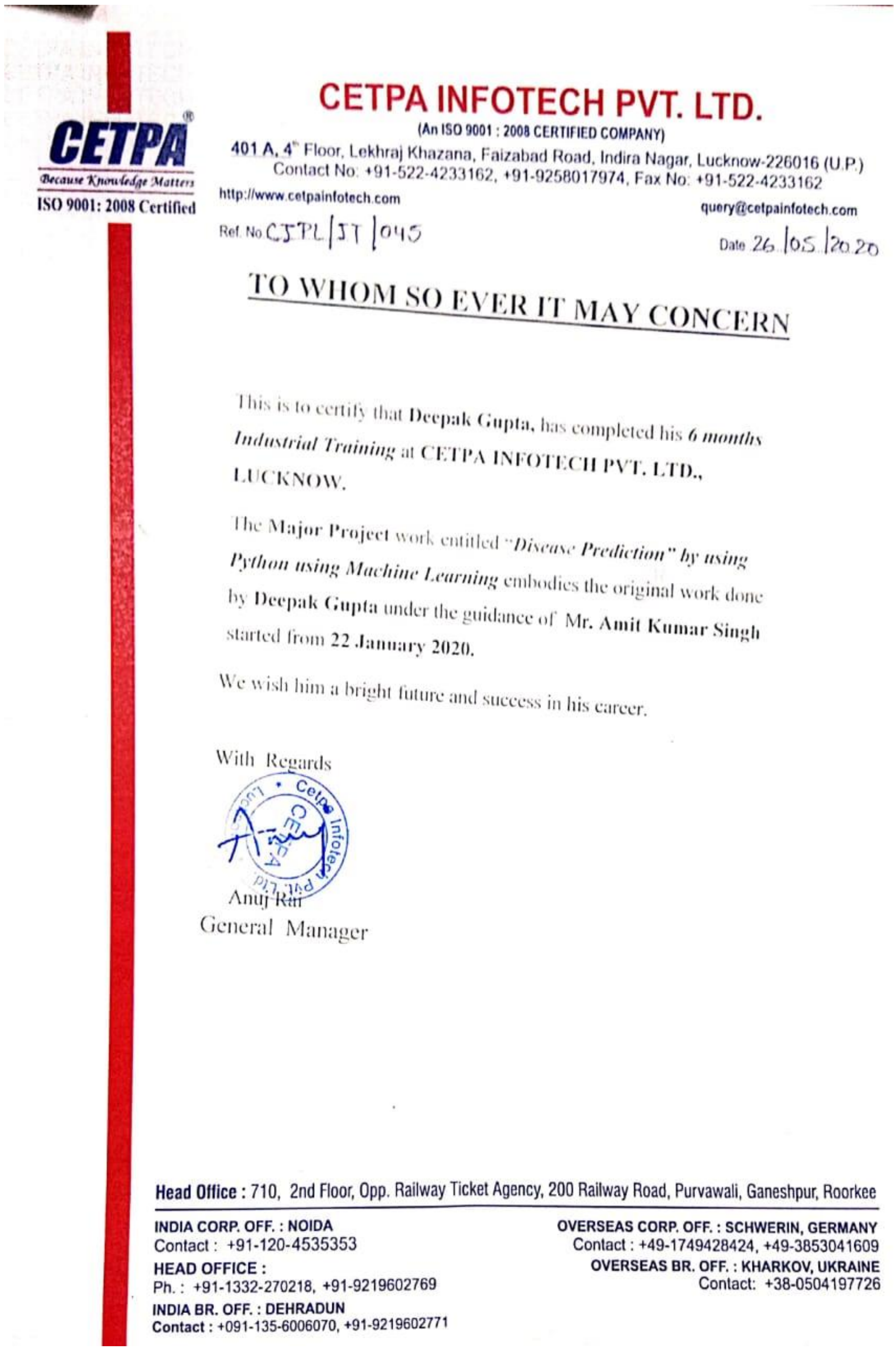
in the Academic Year 2019-20

is a record of the students own work carried out at

Cetpa Infotech Pvt Ltd

and to the best of our knowledge the work reported herein does not form a part of any other thesis or work on the basis of which degree or award was conferred on an earlier occasion to this or any other candidate.

Prabhash Ch. Pathak
HEAD (School of Computer Applications)



CETPA
Because Knowledge Matters
ISO 9001: 2008 Certified

CETPA INFOTECH PVT. LTD.

(An ISO 9001 : 2008 CERTIFIED COMPANY)

401 A, 4th Floor, Lekhraj Khazana, Faizabad Road, Indira Nagar, Lucknow-226016 (U.P.)
Contact No: +91-522-4233162, +91-9258017974, Fax No: +91-522-4233162

http://www.cetpainfotech.com
query@cetpainfotech.com

Ref. No CIPL/IT/045

Date 26/05/2020

TO WHOM SO EVER IT MAY CONCERN

This is to certify that **Deepak Gupta**, has completed his ***6 months Industrial Training*** at **CETPA INFOTECH PVT. LTD., LUCKNOW.**

The **Major Project** work entitled ***"Disease Prediction" by using Python using Machine Learning*** embodies the original work done by **Deepak Gupta** under the guidance of **Mr. Amit Kumar Singh** started from **22 January 2020.**

We wish him a bright future and success in his career.

With Regards

Cetpa Infotech Pvt. Ltd. Lucknow CETPA

Anuj Rai
General Manager

Head Office : 710, 2nd Floor, Opp. Railway Ticket Agency, 200 Railway Road, Purvawali, Ganeshpur, Roorkee

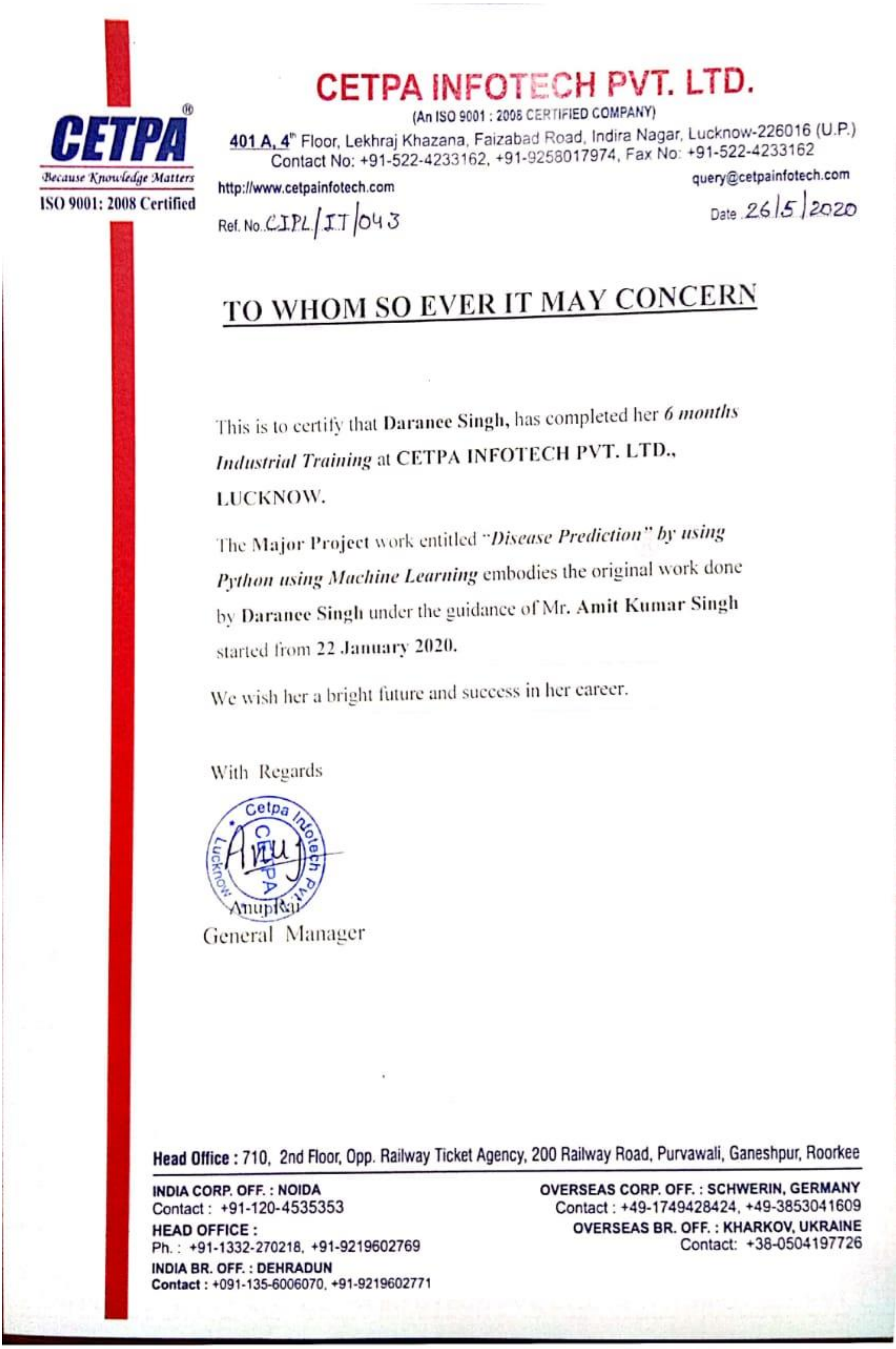
INDIA CORP. OFF. : NOIDA
Contact : +91-120-4535353

HEAD OFFICE :
Ph. : +91-1332-270218, +91-9219602769

INDIA BR. OFF. : DEHRADUN
Contact : +091-135-6006070, +91-9219602771

OVERSEAS CORP. OFF. : SCHWERIN, GERMANY
Contact : +49-1749428424, +49-3853041609

OVERSEAS BR. OFF. : KHARKOV, UKRAINE
Contact: +38-0504197726



CETPA

Because Knowledge Matters

ISO 9001: 2008 Certified

CETPA INFOTECH PVT. LTD.

(An ISO 9001 : 2008 CERTIFIED COMPANY)

401 A, 4th Floor, Lekhraj Khazana, Faizabad Road, Indira Nagar, Lucknow-226016 (U.P.)
Contact No: +91-522-4233162, +91-9258017974, Fax No: +91-522-4233162

http://www.cetpainfotech.com

query@cetpainfotech.com

Ref. No. CIPL/IT/043

Date 26/5/2020

TO WHOM SO EVER IT MAY CONCERN

This is to certify that **Daranee Singh,** has completed her ***6 months Industrial Training*** at CETPA INFOTECH PVT. LTD., LUCKNOW.

The **Major Project** work entitled ***"Disease Prediction" by using Python using Machine Learning*** embodies the original work done by **Daranee Singh** under the guidance of Mr. **Amit Kumar Singh** started from **22 January 2020.**

We wish her a bright future and success in her career.

With Regards

Cetpa Infotech Pvt. Lucknow CETPA

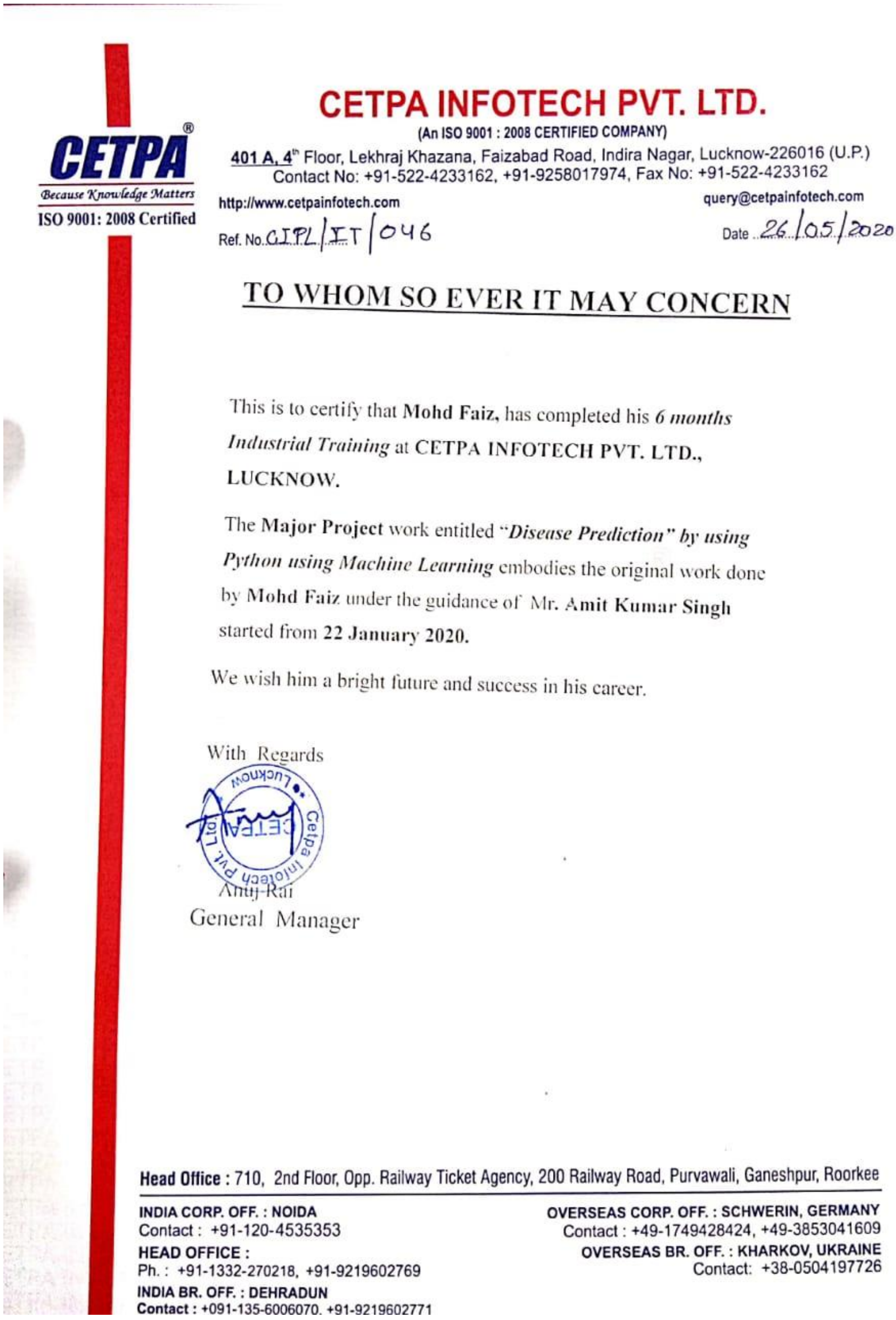
Anuj

Anup Rai
General Manager

Head Office : 710, 2nd Floor, Opp. Railway Ticket Agency, 200 Railway Road, Purvawali, Ganeshpur, Roorkee

INDIA CORP. OFF. : NOIDA
Contact : +91-120-4535353
HEAD OFFICE :
Ph. : +91-1332-270218, +91-9219602769
INDIA BR. OFF. : DEHRADUN
Contact : +091-135-6006070, +91-9219602771

OVERSEAS CORP. OFF. : SCHWERIN, GERMANY
Contact : +49-1749428424, +49-3853041609
OVERSEAS BR. OFF. : KHARKOV, UKRAINE
Contact: +38-0504197726



CETPA®

Because Knowledge Matters

ISO 9001: 2008 Certified

CETPA INFOTECH PVT. LTD.

(An ISO 9001 : 2008 CERTIFIED COMPANY)

401 A, 4th Floor, Lekhraj Khazana, Faizabad Road, Indira Nagar, Lucknow-226016 (U.P.)
Contact No: +91-522-4233162, +91-9258017974, Fax No: +91-522-4233162

http://www.cetpainfotech.com query@cetpainfotech.com

Ref. No. CIPL/IT/046 Date 26/05/2020

TO WHOM SO EVER IT MAY CONCERN

This is to certify that **Mohd Faiz**, has completed his ***6 months Industrial Training*** at CETPA INFOTECH PVT. LTD., LUCKNOW.

The **Major Project** work entitled "***Disease Prediction" by using Python using Machine Learning*** embodies the original work done by **Mohd Faiz** under the guidance of **Mr. Amit Kumar Singh** started from **22 January 2020.**

We wish him a bright future and success in his career.

With Regards

Lucknow Cetpa Infotech Pvt. Ltd. CETPA

Anuj Rai
General Manager

Head Office : 710, 2nd Floor, Opp. Railway Ticket Agency, 200 Railway Road, Purvawali, Ganeshpur, Roorkee

INDIA CORP. OFF. : NOIDA
Contact : +91-120-4535353
HEAD OFFICE :
Ph. : +91-1332-270218, +91-9219602769
INDIA BR. OFF. : DEHRADUN
Contact : +091-135-6006070, +91-9219602771

OVERSEAS CORP. OFF. : SCHWERIN, GERMANY
Contact : +49-1749428424, +49-3853041609
OVERSEAS BR. OFF. : KHARKOV, UKRAINE
Contact: +38-0504197726

TABLE OF CONTENT

S. NO. **CONTENTS**

INTRODUCTION

Disease prediction using machine learning and health data by applying data mining and machine learning techniques is ongoing struggle for the past decades. Many works have been applied data mining techniques to pathological data or medical profiles for prediction of specific diseases. These approaches tried to predict the reoccurrence of disease. The recent success of deep learning in disparate areas of machine learning has driven a shift towards machine learning models that can learn rich, hierarchical representations of raw data with little pre-processing and produce more accurate results. With the development of data analytics technology, more attention has been paid to disease prediction from the perspective of big data analysis; various researches have been conducted.

The main focus is on to use machine learning in healthcare tosupplement patient care for better results. Machine learning has made easier to identify different diseases and diagnosis correctly. Predictive analysis with the help of efficient multiple machine learning algorithms helps to predict the disease more correctly and help treat patients.

Medical facilities need to be advanced so that better decisions for patient diagnosis and treatment options can be made. Machine learning in healthcare aids the humans to process huge and complex medical datasets and then analyze them into clinical insights.

- Disease Predictor is a application that predicts the disease of the user with respect to the symptoms by the user. This system has data set collected from different health related data.

- With the help of Disease Predictor the user will be able to know the probability of the disease with the given Symptoms.

- Machine Learning allows building models to quickly analyze data and deliver results, leveraging both historical and real-time data.
- With machine learning, healthcare service providers can make better decisions on patient’s diagnoses and treatment options, which leads to the overall improvement of healthcare services.

- This thesis mainly focuses on the prediction of diabetic retinopathy and analysis performed of different algorithm for the prediction. Machine learning algorithms such as KNN, RF, SVM, NNET etc. can be trained by providing training datasets to them and then these algorithms can predict the data by comparing the provided data with the training datasets. Our objective is to train our algorithm by providing training datasets to it and our goal is to detect diabetic retinopathy using different types of classification algorithms.

PROBLEM STATEMENT

Healthcare industry has become big business. The healthcare industry produces large amounts of health-care data daily that can be used to extract information for predicting disease that can happen to a patient in future while using the treatment history and health data. This hidden information in the healthcare data will be later used for affective decision making for patient's health. Also, this area need improvement by using the informative data in healthcare.

Major challenge is how to extract the information from these data because the amount is very large so some data analytics and machine learning techniques can be used. Also, the expected outcome and scope of this project is that if disease can be predicted than early treatment can be given to the patients which can reduce the risk of life and save life of patients and cost to get treatment of diseases can be reduced up to some extent by early recognition.

Advancements in healthcare technology have allowed physicians to diagnose and treat their patients better. Thanks to the continuous emerging technology in the healthcare industry, countless lives have been saved, and the overall quality of life continues to improve.

The rapid adoption of electronic health records has created a wealth of new data about patients, which is a goldmine for improving the understanding of human health. The above method is used to predict diseases using patient treatment history and health data.

Here are the top 5 problems in healthcare that technology can solve to remain fit Making remote access to telemedicine is a substantial challenge. Telemedicine, together with mobile healthcare, aim to raise the level of awareness in an age where wearable and smart phones are used to track and improve health

With healthcare application, telemedicine aims to raise the level of healthcare with wearable and smartphones to track health. Telemedicine offers enrolment of patients in remotemonitoring projects, use of mobile apps, video conferencing, and emails for the counseling and treatment.Healthcare reform has been one of the most disruptive political issues right now. Electronic Health Record (EHR) systems are one of those healthcare technology trends that the industry is ready to embrace in order to simplify health record-keeping as well as Medicare and Medicaid filing.

SYSTEM ANALYSIS

1. Need of Identification

The purpose of bringing this project is that sometimes we do not have doctors available. Due to which we have a lot of difficulty in taking treatment and we do not know the treatment on time due to which the patient is killed even.After bringing this project, we will be able to give timely treatment to the patient and prevent death.Sometimes it happens that we do not know the right doctor and right treatment for the disease.With the help of this application, we will be able to find the right cure for the disease and the right doctor.we have combined the structure and unstructured data in healthcare fields that let us assess the risk of disease. The approach of the latent factor model for reconstructing the missing data in medical records which are collected from the hospital. And by using statistical knowledge, we could determine the major chronic diseases in a particular region and in particular community. To handle structured data, we consult hospital experts to know useful features. In the case of unstructured text data, we select the features automatically with the help of k-mean algorithm. We propose a naïve bayes, k-nearest neighbors algorithm and logistic regresion for both structured and unstructured data

- With the help of this application, we will be able to get treatment at the right time without a doctor.
- Will be able to save the patient from dying.
- The right treatment for the disease will be available.
- Will be able to find the right doctor for the disease.
- Manytimes we have such a disease which we are shy to tell to the doctor.With the help of this application, we will be able to cure the disease by ourselves.
- We will be able to identify the disease from the symptoms without test reports.
- With the help of this application, we will be able to detect disease from anywhereand find a doctor.
- With the help of this application, we will also save money going after the wrong doctors.

2. EXISTING SYSTEM:-

Prediction using traditional disease risk model usually involves a machine learning and supervised learning algorithm which uses training data with the labels for the training of the models. High-risk and Low-risk patient classification is done in groups test sets. But these models are only valuable in clinical situations and are widely studied. A system for sustainable health monitoring using smart clothing by Chen et.al. He thoroughly studied heterogeneous systems and was able to achieve the best results for cost minimization on the tree and simple path cases for heterogeneous systems.

Machine can predict diseases but cannot predict the sub types of the diseases caused by occurrence of one disease. It fails to predict all possible conditions of the people. Existing system handles only structured data. The prediction system are broad and ambiguous. In current past, countless disease estimate classifications have been advanced and in procedure. The standing organizations arrange a blend of machine learning algorithms which are judiciously exact in envisaging diseases. However the restraint with the prevailing systems are speckled. First, the prevailing systems are dearer only rich people could pay for to such calculation systems. And also, when it comes to folks, it becomes even higher. Second, the guess systems are non-specific and indefinite so far. So that, a machine can envisage a positive disease but cannot expect the sub types of the diseases and diseases caused by the existence of one bug. For occurrence, if a group of people are foreseen with Diabetes, doubtless some of them might have complex risk for Heart viruses due to the actuality of Diabetes. The remaining schemes fail to foretell all possible surroundings of the tolerant.

The information of patient's statistics, test results, and disease history is recorded in EHR which enables to identify potential data-centric solutions which reduce the cost ofmedical case studies. Bates et al. propose six applications of big data in the healthcare field. Existing systems can predict the diseases but not the subtype of diseases. It fails to predict the condition of people.The predictions of diseases have been non-specific and indefinite.

a) Preliminary Investigation

- Evaluation of project request is major purpose of preliminary investigation.
- It is the collecting information that helps committee members to evaluate merits of the project request and make judgment about the feasibility of the purposed projects.
- To answer the above questions, system analysis discuss with different category of person to collect facts about their business and their operations.
- When the request is made, the first activity is the preliminary investigation begins.
- The next step is to determine the requirements met by the system. Many requests from employees and users in the organizations are not clearly defined. Therefore, it become necessary that project request must examine and clarified properly before considering system investigation.
- Information related to different needs of the System can be obtained by different users of the system. This can be done by reviewing different organization's documents such as current method of storing sales data, complaint data etc. By observing the onsite activities the analyst can get close information related to system.

b) Feasibility Study

The feasibility report of the project holds the advantages and flexibility of the project. This is divided into three sections:

- Economic Feasibility
- Technical Feasibility
- Operational Feasibility

a) **Economic feasibility**:-

Economic feasibility is a method of studying economic processes, which consists in considering the relationships between the various elements of these processes. A system financial benefit must exceed the cost of developing that system. i.e., a new system being developed should be a good investment for the organization.

The proposed system is economically feasible because:

- The application requires very less time factors.
- The application reduces paper cost.
- The application will have GUI interface and very less user-training is required to learn it.
- The application will predict fast and efficient results.
- The application will provide fast and efficient automated environment instead of slow and error prone manual system, thus reducing both time and man power spent in finding the diseases.
- With the help of this application, we will also save money going after the wrong doctors.

b) **Technical Feasibility**:-

Technical feasibility centers around the existing computer system (hardware and software) whether it can support the addition of proposed system, if not, to what extent it can support and the organization's capacity to acquire additional components.

Our proposed system is technically feasible because:-

- The hardware and software required are easy to install and handle.
- The necessary hardware configuration and software platform is already there.
- The system supports interactivity with the user through GUI.

Proposed system is technically feasible because of the following reasons:-

- It's required less system resources.
- Expandability will be maintained in the new system. New modules can be added later on the application, if required in the future.
- The application will have User-friendly Forms and screens, all validation checks. So the new system guarantees accuracy, reliable, ease of access and data security.

c) **Behavioral Feasibility**:-

Behavioral feasibility determines how much effort will go in the proposed information system, and in educating and training the users on the new system, along with the new ways of conducting the business. Behavioral study strives on ensuring that the equilibrium of the organization and status que in the organization neither are nor disturbed and changes are readily accepted by the users.

The proposed system is behaviorally feasible because of the following:-

- The users will accept it because they are already acquainted with computers.
- This system is also meant for the general user. Nowadays the Internet is almost familiar to everyone. So, it is not difficult for the user to use the system, in fact they feel comfortable in using this system.
- Most of the users are familiar with the web browser and the process of booking the auditorium will be simplified for the users. The organization is definitely ready to welcome the computerized system.

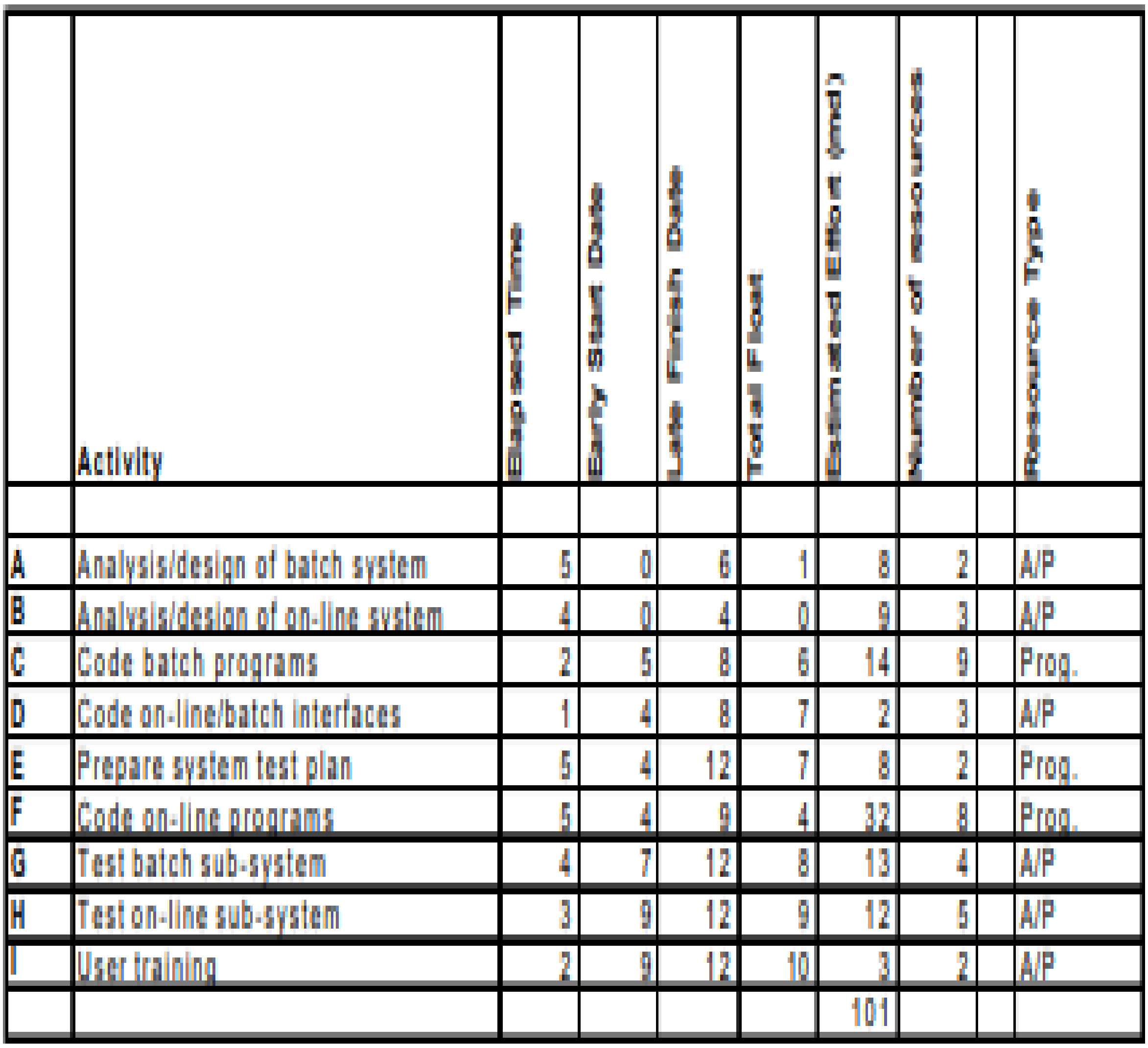
PROJECT PLANNING

Project planning is part of project management, which relates to the use of schedules such as Gantt charts to plan and subsequently report progress within the project environment.Project scheduling is one of the critical management tasks as it dictates the time frames in which the project will be completed, the budgets/costs in terms of resource requirements and the sequence of tasks to be completed. Project scheduling is defined as the process of determining when project activities will take place depending upon defined durations and precedent activities. Schedule constraints specify when an activity should start or end, based on duration, predecessors, external predecessor relationships, resource availability, target dates or other time constraints. Project scheduling is a complex and iterative task which typically involves the following steps:-

The steps we followed while developing this project are-:

1. Analysis of the problem statement.
2. Gathering of the requirement specification.
3. Analyzation of the feasibility of the project.
4. Development of a general layout.
5. Going by the journals regarding the previous related works on this field.
6. Choosing the method for developing the algorithm.
7. Analyzing the various pros and cons.
8. Starting the development of the project.
9. Installation of software like ANACONDA.
10. Developing an algorithm.
11. Analyzation of algorithm by guide.
12. Coding as per the developed algorithm in PYTHON.

Our application will be at affordable cost. Decision Tree Machine Learning Algorithm predicts Diseases as well as all sub diseases. Map Reduce Algorithm is implemented to increase operational efficiency. It reduces Query retrieval time. Accuracy is improved using Machine Learning algorithm. The proposed system begin with the thought that was not executed by the ancestors. It gadget Decision Tree machine learning procedure for calculating diseases as well as calculating all the other thinkable sub diseases. It member Map Reduce algorithm for subdividing the data such that a request would be scrutinized only in the explicit partition, which will increase effective proficiency but cut query rescue time. In tally to that, it provide definite rations for specific clients to pattern his/her condition. Thus making our presentation broadly open by all at cheap cost.



	Activity	Elapsed Time	Early Start Date	Late Finish Date	Total Float	Estimated Effort (pd)	Number of resources		Resource Type
A	Analysis/design of batch system	5	0	6	1	8	2		A/P
B	Analysis/design of on-line system	4	0	4	0	9	3		A/P
C	Code batch programs	2	5	8	6	14	9		Prog.
D	Code on-line/batch interfaces	1	4	8	7	2	3		A/P
E	Prepare system test plan	5	4	12	7	8	2		Prog.
F	Code on-line programs	5	4	9	4	32	8		Prog.
G	Test batch sub-system	4	7	12	8	13	4		A/P
H	Test on-line sub-system	3	9	12	9	12	5		A/P
I	User training	2	9	12	10	3	2		A/P
						101			

PROJECT SCHEDULING

Project scheduling is a mechanism to communicate what tasks need to get done and which organizational resources will be allocated to complete those tasks in what timeframe. A project schedule is a document collecting all the work needed to deliver the project on time.

- Some other optimizing considerations may include:
- Opportunities for optimizing may be limited within a project.
- Opportunities for optimizing may be greater across projects.
- The approach at the point of integration of projects may need to be re-addressed.
- The impact of any change on all projects must be considered. Where the end-date of the integrated program schedule is later than the latest individual project completion dates.
- All critical parameters must be reconciled.
- The overall program and all project schedules must be balanced when optimizing.
- Look for opportunities to reach mutually agreeable compromises between projects .
- Try not to compress or expand one project at the expense of the others
- Focus on the diagram.
- For each proposed project change, evaluate the effect on the program as well as onthe individual projects.

Planning and Scheduling

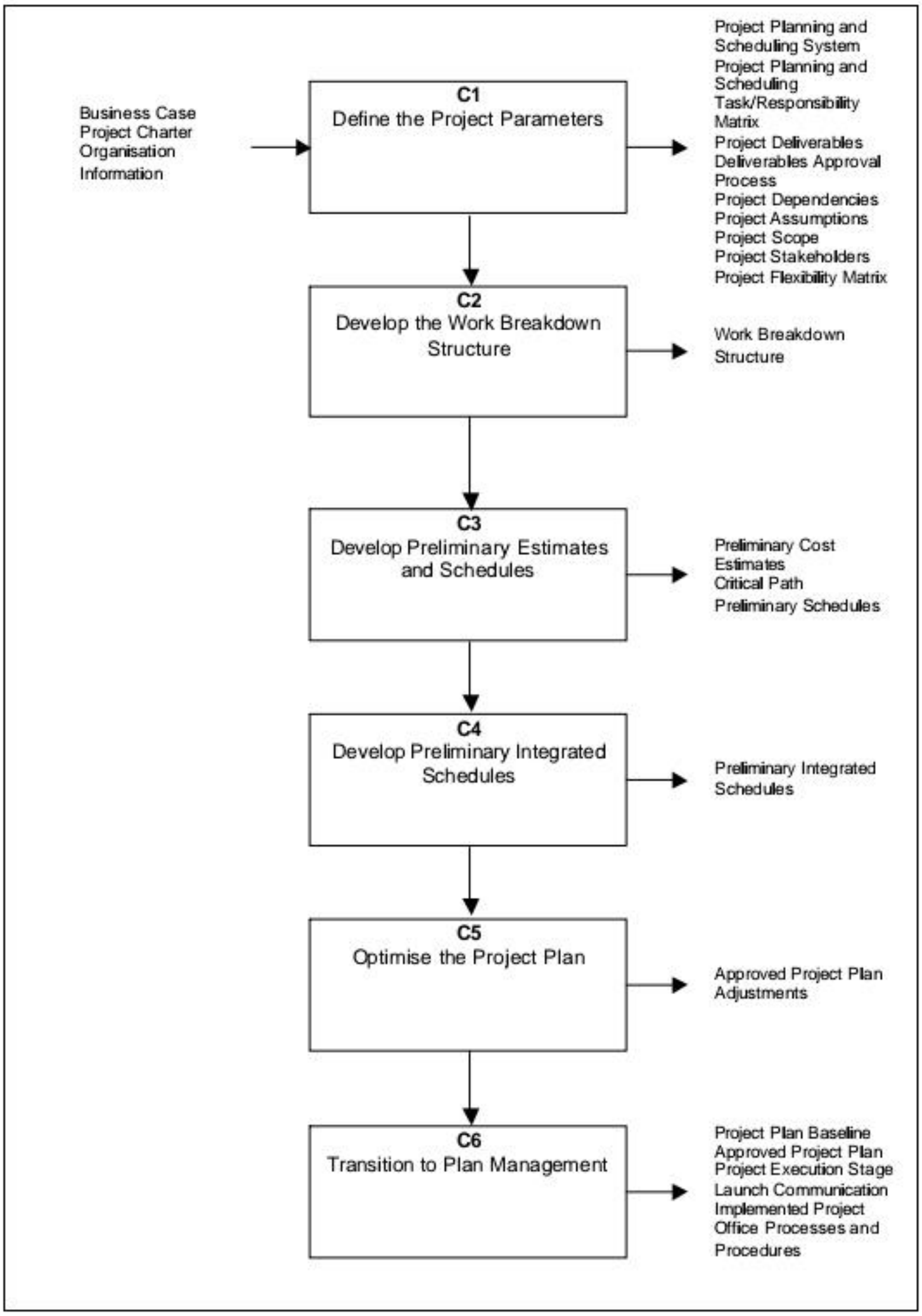

Business Case
Project Charter
Organisation
Information
C1
Define the Project Parameters
Project Planning and Scheduling System
Project Planning and Scheduling
Task/Responsibility Matrix
Project Deliverables
Deliverables Approval Process
Project Dependencies
Project Assumptions
Project Scope
Project Stakeholders
Project Flexibility Matrix
C2
Develop the Work Breakdown Structure
Work Breakdown Structure
C3
Develop Preliminary Estimates and Schedules
Preliminary Cost Estimates
Critical Path
Preliminary Schedules
C4
Develop Preliminary Integrated Schedules
Preliminary Integrated Schedules
C5
Optimise the Project Plan
Approved Project Plan Adjustments
C6
Transition to Plan Management
Project Plan Baseline
Approved Project Plan
Project Execution Stage
Launch Communication
Implemented Project Office Processes and Procedures

Fig: - PERT Chart

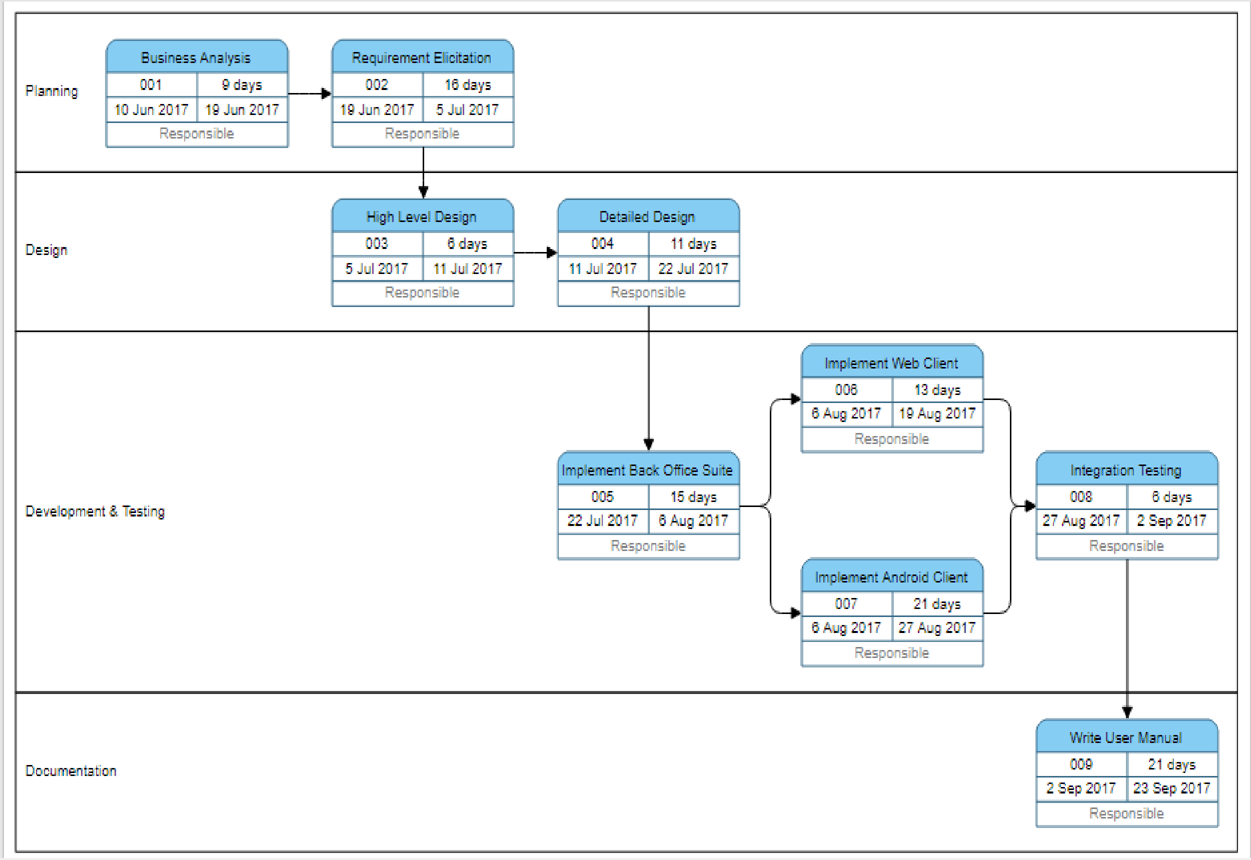
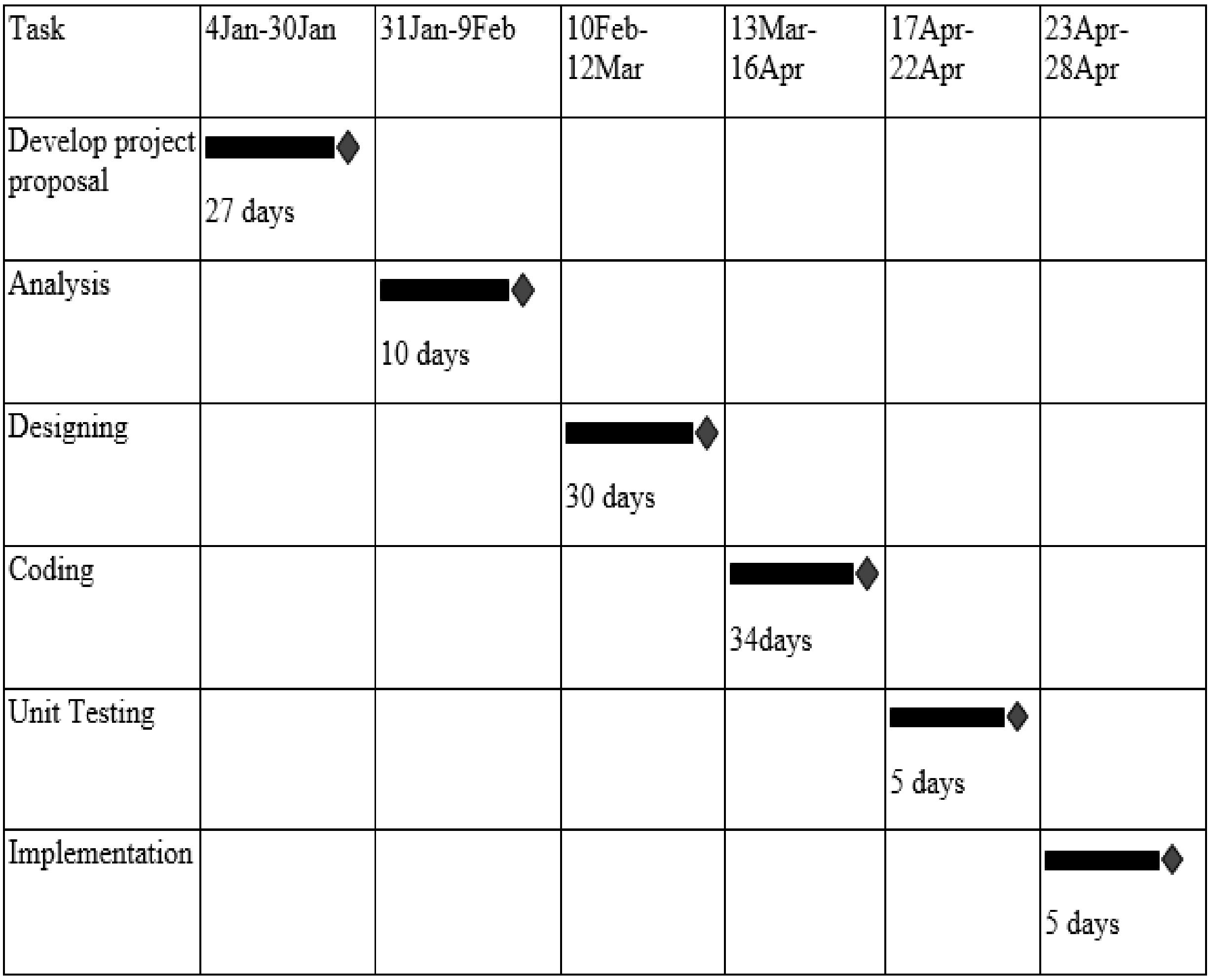


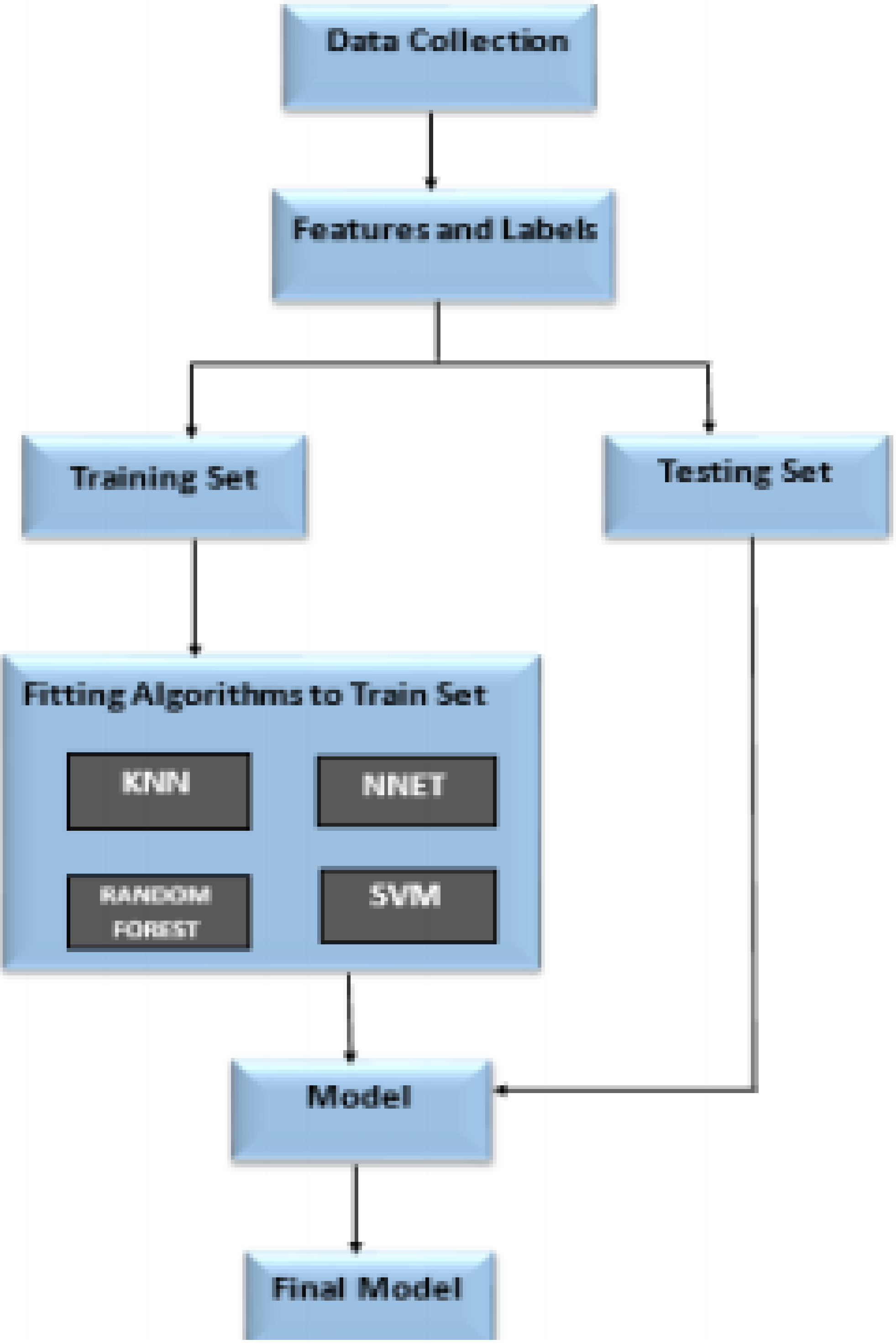
Fig:- Gantt Chart



Task	4Jan-30Jan	31Jan-9Feb	10Feb-12Mar	13Mar-16Apr	17Apr-22Apr	23Apr-28Apr
Develop project proposal	▬◆ 27 days					
Analysis		▬◆ 10 days				
Designing			▬◆ 30 days			
Coding				▬◆ 34days		
Unit Testing					▬◆ 5 days	
Implementation						▬◆ 5 days

Gantt chart

Proposed system of retinopathy:-



SOFTWARE REQUIREMENT SPECIFICATION

- **Hardware Specification :-**

  It is recommended that the minimum configuration for clients is as appended below:-

System	:	Dual Core Processor or above
Hard Disk	:	500 GB.
Monitor	:	15''LED
Input Devices	:	Keyboard, Mouse
Ram	:	4GB.

- **Software Specification:-**

Operating system	:	Windows 7/UBUNTU.
Coding Language	:	Python 3.7
IDE	:	Anaconda(Visual Studio)
Database	:	Sqlite3 Manager
Other	:	Db Browser

- **Platform:-**

Frontend	:	Visual Studio Code
Backend	:	Db Browser (SQL lite 3)

As the project is developed in python, we have used Anaconda for Python 3.6.5 and Visual Studio Code.

- **Anaconda:**

  It is a free and open source distribution of the Python and R programming languages for data science and machine learning related applications (large-scale data processing, predictive analytics, scientific computing), that aims to simplify package management and deployment. Package versions are managed by the package management system conda. The Anaconda distribution is used by over 6 million users, and it includes more than 250 popular data science packages suitable for Windows, Linux, and MacOS.

- **Visual Studio Code:**

Visual Studio Code is a source-code editor developed by Microsoft for Windows, Linux and macOS. It includes embedded Git and support for debugging, syntax highlighting, intelligent code completion, snippets, and code refactoring. It is highly customizable, allowing users to change the theme, keyboard shortcuts, preferences, and install extensions that add additional functionality. The source code is free and open-source, released under the permissive MIT License.The compiled binaries are freeware for any use.

Features include:

- Editor with syntax highlighting and introspection for code completion.
- Support for multiple Python consoles (including IPython)
- The ability to explore and edit variable from a GUI

Available plugins include:

- Static Code Analysis with Pylint
- Code Profiling
- Conda Package Manager with Conda

SOFTWARE TESTING

Software Testing is a critical element of software quality assurance and represents the ultimate review of specification, design and coding, Testing presents an interesting anomaly for the software engineer.

Testing Objectives include:

1. Testing is a process of executing a program with the intent of finding an error
2. A good test case is one that has a probability of finding an as yet undiscovered error

3. A successful test is one that uncovers an undiscovered error

Testing Principles:

- All tests should be traceable to end user requirements
- Tests should be planned long before testing begins
- Testing should begin on a small scale and progress towards testing in large
- Exhaustive testing is not possible
- To be most effective testing should be conducted by a independent third party

TESTING STRATERGIES

A Strategy for software testing integrates software test cases into a series of well-planned steps that result in the successful construction of software. Software testing is a broader topic for what is referred to as Verification and Validation. Verification refers to the set of activities that ensure that the software correctly implements a specific function Validation refers he set of activities that ensure that the software that has been built is traceable to customer's requirements

Unit Testing:

Unit testing focuses verification effort on the smallest unit of software design that is the module. Using procedural design description as a guide, important control paths are tested to uncover errors within the boundaries of the module. The unit test is normally white box testing oriented and the step can be conducted in parallel for multiple modules.

Integration Testing:-

Integration testing is a systematic technique for constructing the program structure while conducting test to uncover errors associated with the interfacing. The objective is to take unit tested methods and build a program structure that has been dictated by design.

Top-down Integration:-

Top down integrations an incremental approach to construction of program structure. Modules are integrated by moving downward through the control hierarchy, beginning with the main control program. Modules subordinate to the main program are incorporated in the structure either in the breath-first or depth-first manner.

- **Bottom-up Integration:-**

This method as the name suggests, begins construction and testing with atomic modules i.e., modules at the lowest leveling the program structure. Because the modules are integrated in the bottom up manner the processing required for the modules subordinate to a given level is always available and the need for stubs is eliminated.

- **Validation Testing:-**

At the end of integration testing software ids completely assembled as a package. Validation testing is the next stage which can be defined as successful when the software functions in the manner reasonably expected by the customer. Reasonable expectations are those defined in the software requirements specifications. Information contained in those sections form a basis for validation testing approach.

- **System Testing:**

System testing is actually a series of different tests whose primary purpose is to fully exercise the computer-based system. Although each test has a different purpose, all work to verify that all system elements have been properly integrated to perform allocated functions.

- **Recovery Testing:**

It is a system test that forces the system to fail in a variety of ways and verities that the recovery is properly performed.

- **Security Testing:**

Attempts to verify the protection mechanisms built into the system.

- **Performance Testing:**

This method is designed to test runtime performance of software within the context of an integrated system..Software testing is a critical element of software quality assurance and represents the ultimate review of specification, design and coding. Testing is the exposure of the system to trial input to see whether it produces correct output.

- **Testing Phases:**

Software testing phases include the following:
Test activities are determined and test data selected.

The test is conducted and test results are compared with the expected results.

There are various types of Testing:

- **Unit Testing:-**

Unit testing is essentially for the verification of the code produced during the coding phase and the goal is test the internal logic of the module/program.

This project is thoroughly tested by exposing it to the various test cases regarding correct event generation, as this project passed all the tests its quality is completely assured.

- **Integration Testing:-**

All the tested modules are combined into sub systems, which are then tested. The goal is to see if the modules are properly integrated, and the emphasis being on the testing interfaces between the modules. On this project integration testing is done mainly while implementing menus in a sample application such as Browser for Mobiles.

- **System Testing:-**

It is mainly used if the software meets its requirements. The reference document for this process is the requirement document.

- **Acceptance Testing:-**

It is performed with realistic data of the client to demonstrate that the software is working satisfactorily.

- **Testing Methods:-**

Testing is a process of executing a program to find out errors. If testing is conducted successfully, it will uncover all the errors in the software. Any testing can be done basing on two ways:

- **White Box Testing:-**

It is a test case design method that uses the control structures of the procedural design to derive test cases. using this testing a software Engineer can derive the following test cases:
Exercise all the logical decisions on either true or false sides. Execute all loops at their boundaries and within their operational boundaries. Exercise the internal data structures to assure their validity.

- **Black Box Testing:-**

It is a test case design method used on the functional requirements of the software. It will help a software engineer to derive sets of input conditions that will exercise all the functional requirements of the program. Black Box testing attempts to find errors in the following categories:

Incorrect or missing functions
Interface errors
Errors in data structures
Performance errors
Initialization and termination errors
By Black Box Testing we derive a set of test cases that satisfy the following criteria:

Test cases that reduce by a count that is greater than one, the number of additional test cases that must be designed to achieve reasonable testing.
Test cases that tell us something about the presence or absence of classes of errors rather than errors associated only with a specific test at hand.

- **Test Approach:-**

Testing can be done in two ways:
Bottom up approach
Top down approach

Bottom up Approach:

Testing can be performed starting from smallest and lowest level modules and proceeding one at a time. For each module in bottom up testing a short program executes the module and provides the needed data so that the module is asked to perform the way it will when embedded within the larger system. When bottom level modules are tested attention turns to those on the next level that use the lower level ones they are tested individually and then linked with the previously examined lower level modules.

Top down approach:-

This type of testing starts from upper level modules. Since the detailed activities usually performed in the lower level routines are not provided stubs are written. A stub is a module shell called by upper level module and that when reached properly will return a message to the calling module indicating that proper interaction occurred. No attempt is made to verify the correctness of the lower level module.

Input and Output approach:-

Input design is the link that ties the information system into the world of its Users. The input design involves determining the inputs, validating the data, minimizing the data entry and provides a multi-user facility. Inaccurate inputs are the most common cause of errors in data processing. Errors entered by the data entry operators can be controlled by input design.

The user-originated inputs are converted to a computer based format in the input design. Input data are collected and organized into groups of similar data. Once identified, the appropriate input media are selected for processing. All the input data are validated and if any data violates any conditions, the user is warned by a message.

If the data satisfies all the conditions, it is transferred to the appropriate tables in the database. In this project the student details are to be entered at the time of registration. A page is designed for this purpose which is user friendly and easy to use. The design is done such that users get appropriate messages when exceptions occur.

Computer output is the most important and direct source of information to the user. Output design is a very important phase since the output needs to be in an efficient manner. Efficient and intelligible output design improves the system relationship with the user and helps in decision making.

Allowing the user to view the sample screen is important because the user is the ultimate judge of the quality of output. The output module of this system is the selected notifications.

SOFTWARE ENGINEERING PARADIGM

The software engineering paradigm which is also referred to as a software process model or Software Development Life Cycle (SDLC) model is the development strategy that encompasses the process, methods and tools. SDLC describes the period of time that starts with the software system being conceptualized and ends with the software system been discarded after usage.

The objectives of the use of software engineering paradigms include:

1. The software development process becomes a structured process.
2. Determine the order of states involved in software development and evolution, and to establish the transitions criteria for the next stage.
3. The software engineering paradigm provides the guidance to the software engineer.

A paradigm specifies the particular approach or philosophy for designing, building and maintaining software. Each paradigm has its own advantages and disadvantages which make some paradigm more suitable to be used in developing a given software system in a given situation than another

There are common software process tasks, phases and activities that are modeled bysoftware models. They are heavily affected by selected software paradigms. These tasks, phases and activities include:

1. Requirements Engineering: Software specification and functional requirements obtained from the user.
2. Requirements Analysis and Modeling
3. Architectural Engineering, implementation and Design: Production of the software system as a product
4. Software Testing and Validation: Activity that assures that customer specifications are met
5. System Delivery.
6. Software Evolution and Upgrading: System modification to meet continuing customer needs
7. System Documenting
8. Maintenance

The four basic models are:

a. Waterfall model, also known as the traditional software development life cycle (SDLC).
b. The spiral model
c. Incremental process model
d. Agile development model

SDLC is a standard project management methodology with proven effectiveness over many years of use in information technology infrastructure and application development projects. The SDLC follows a multistep, iterative process that enablesideas to be transformed into well-documented projects that can be turned into repeatable, operational successes which meet or exceeds a customer's or business's expectations.Patient journey modeling, first described formally in 2006 as quality improvement of health care services, has rapidly become one of the priorities of HCOs around the world.

As in any method based on an SDLC project management framework, stakeholder involvement is critical to its success in delivering quality results within the schedule, budget and scope constraints of a project. Stakeholders such as hospital management executives, physicians, nurses, information technology personnel and patients must be committed to actively participate in the process and provide honest feedback.

The development process of system applications is primarily determined by different project methodologies . The development approach defines a structure used in planning, design and manages the implementation process of computer systems. Aided by software engineering; a systematic approach to the design of software, the software methodologies result to quality software systems balanced with usability requirements. As such, SDLC provides a balanced, systematic practice for the development of high quality and reliable computer systems. The SDLC framework describes sequences of activities that are involved in the process of software production that IT specialists and software engineers must follow. It presents a standard procedure that outlines all the activities undertaken in software development and maintenance. However, several variations of SDLC model such as waterfall, V-model, and spiral models have evolved defining different processes and guiding principles of software development. The SDLC model presents a set of phases where each phase depends on the results of the preceding stage. Various SDLC models are considered for different projects based on their suitability to project conditions such as user's requirements, project risks, cost, and development timeframe. A particular SDLC model may suit a particular project while at the same time other models may appear suitable for the elicited requirements but is it essential to consider trade-offs when choosing an SDLC model. A formal SDLC model theoretically consists of the following phases for developing and implementing computer software.

- Planning
- Analysis
- Design
- Implementation
- Testing
- Deployment and Evolution

Design:- The next phase of SDLC is Design. It is concerned with designing a basic structural framework that identifies the significant component of the product and the communication between these components.

Implementation:- During this stage, the actual process of product development is executed. The product meets the SRS is produced. Deployment: In this phase verification and validation of the product is done.

Verification:- When the software product is built in the right way.

Validation:- When the right software is built. Evolution: It is the last phase of SDLC. Once the product is ready after completion of testing, it is deployed into the runtime environment. Based on the feedback of the product may enhance for further development which is called as maintenance.

This project follows the Phased Life Cycle Model or the Water Fall model to a large extent.

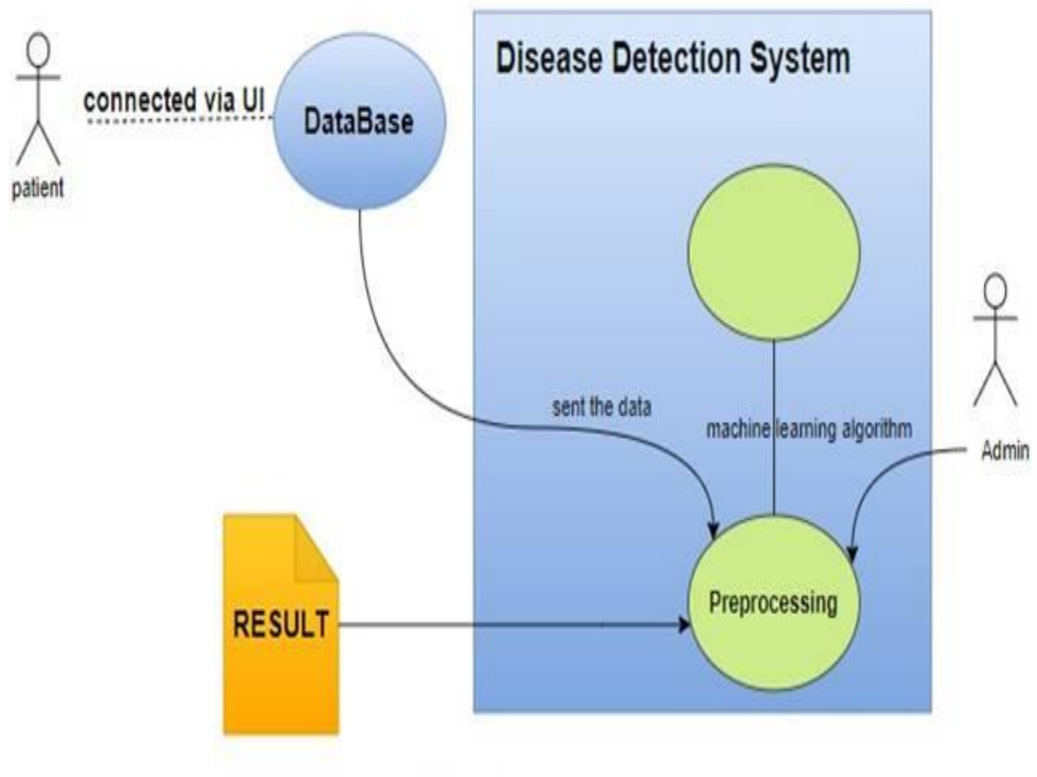
The project is divided into various stages with few overlaps between stages.

- More emphasis is given to planning phase, timeframe, budget, and implementation of the whole project.
- There is strict control over the model cycle of the task through broad reviews, certification and approval user acceptance by the management after implementation phase ends.

The analysis stage consisted of listening to the needs and requirements of all departments obtaining the required format of thesystem as desired by them, taking the required data to be stored for future use etc., In the design stage the structure of the system was designed and all the required screenswere formatted. This was then shown to the medical officer's approval and the system was built. Implementation phase was also done at as they provided a computer with all the required software and with required configuration. The coding and debugging was done even after this stage certain changes were made as made as requested by the guide. The testing was done to check for anyerrors or bugs or unwanted behavior in the system. Individual modules as well as the whole system were tested separately.

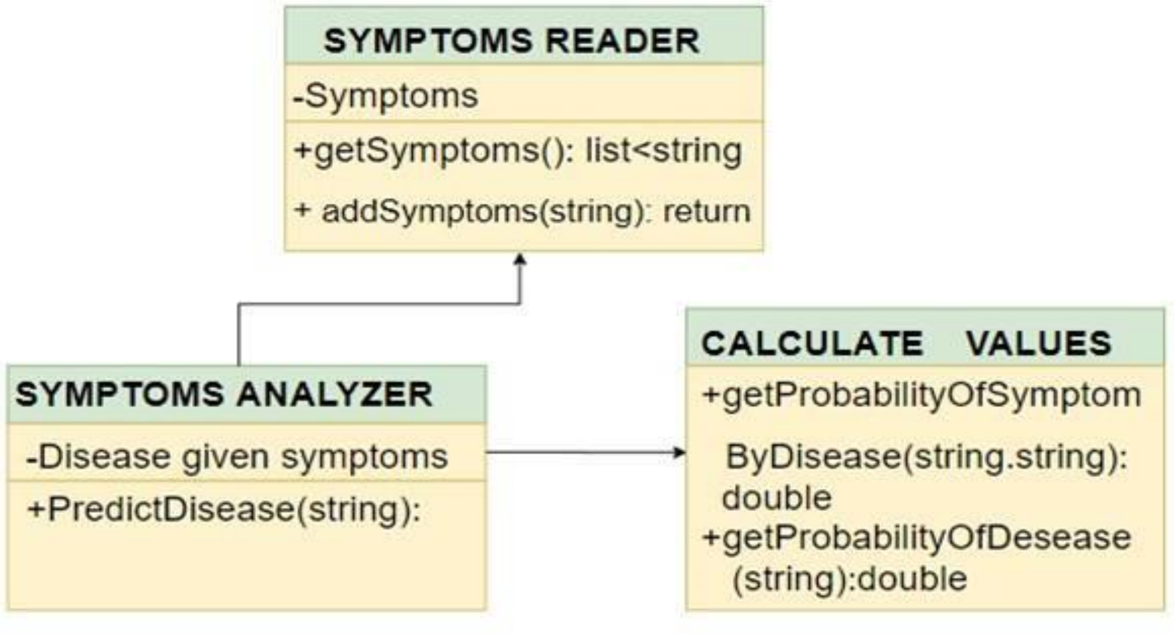
UML MODEL

Use-case diagram:



A use case diagram at its simplest is a representation of a user's interaction with the system that shows the relationship between theuser and the different use cases in which the user is involved. A use case diagram at its simplest is a representation of a user's interaction with the system that shows the relationship between the user and the different use cases in which the user is involved.

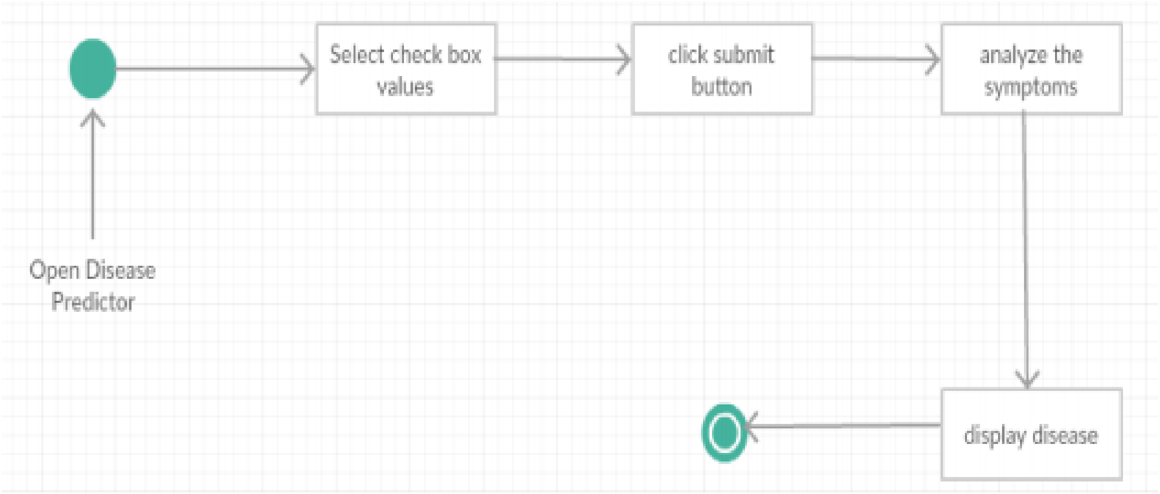
Class diagram:-



CLASS DIAGRAM

It explain the classes used in the Disease Predictor. There are three classes used in total, Symptoms Reader: Reads the user input and creates the list of symptoms Symptoms Analyzer: According to symptoms parameter displays the subjective result. Calculate Values: Calculates the probabilistic model of the diseases.

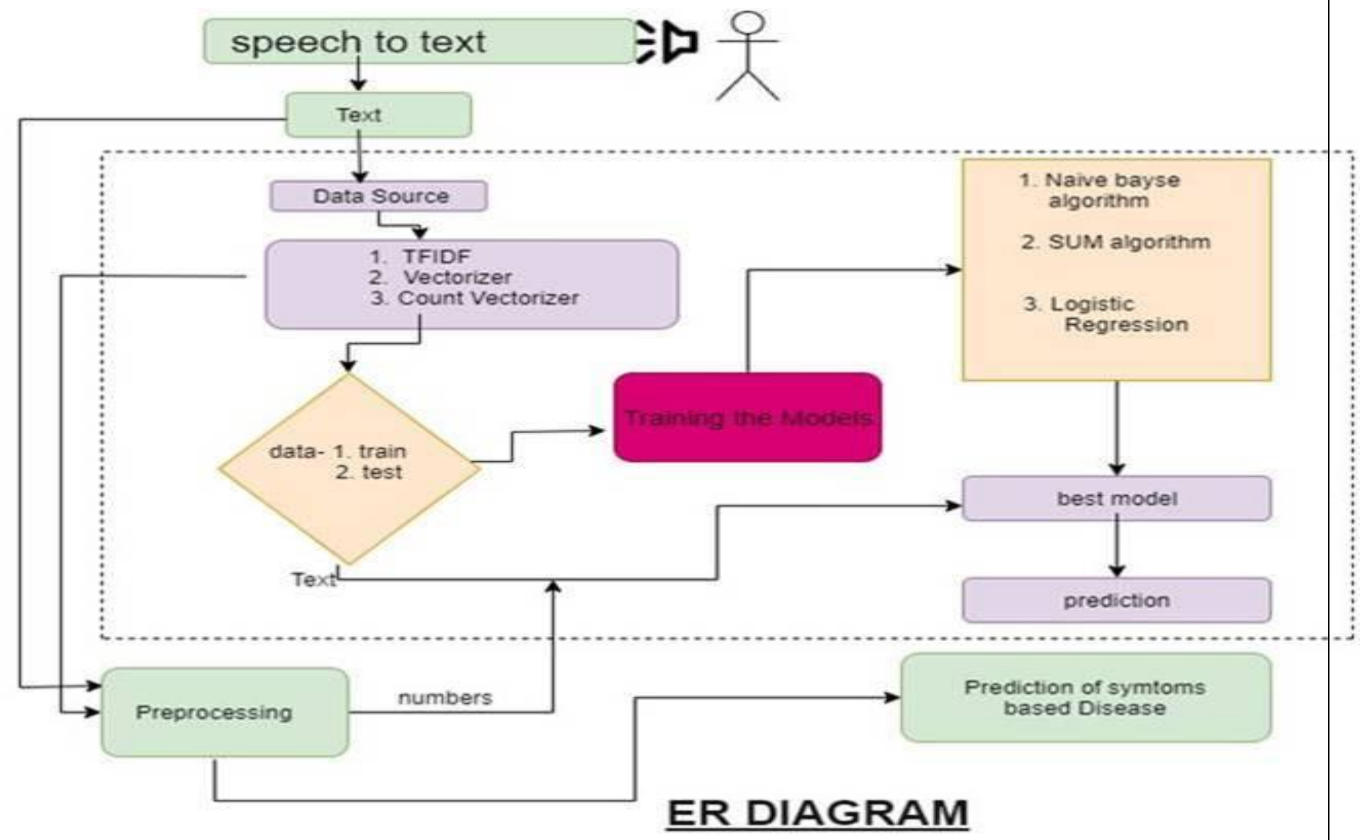
State diagrams:-



It explains different state of the system. First the user opens Disease Predictor. The user selects the symptoms. When finished selecting symptoms the user submits the symptoms.

Disease Predictor analyzes the symptoms and displays the result.

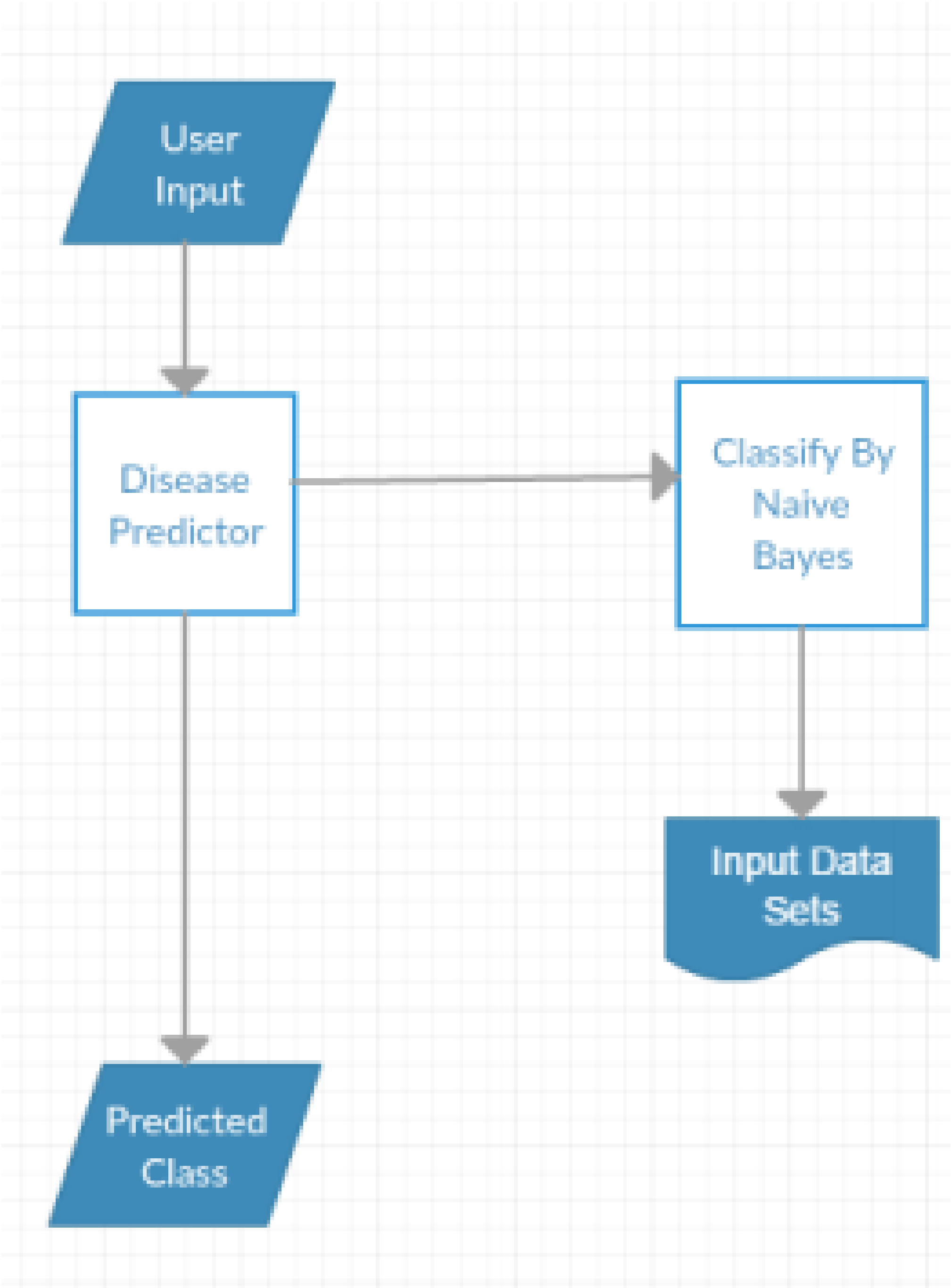
Activity Diagrams:-



It’s explains different process of the system. First the user speaks to the boat and boat converts the speechrecongnisation into the text format. After that boat fetch the data from data source and select the best method for prediction. Data has divided into two process first one is for train where we use the existing data to match with given data and second one is the test data which we’ll going to predict . In this we use Naïve Bayes algorithm, k-nearest neighbor and Random forest. The system will choose the best model and after that predict the symptoms from given data source throughout the algorithm and give as output to user.

Workflow:-

Disease Predictor is the ability to predict the disease that has been provided to the system. For disease prediction, we need to implement the naïve Byes Classifier.



ALGORITHMS

Algorithms- Since there are so many algorithms for machine learning, it is not possible to use all of them for analysis. For this research paper, we will be using four of them neural networks (NNET), random forest (RF), K-Nearest Neighbor (KNN) and support vector machi

1. **Random Forest:-** Random forest algorithm can use both for classification and the regression kind of problems. It is supervised classification algorithm which creates the forest with a number of tress [9]. In general, the more trees in the forest the more robust the forest looks like. It could be also said that the higher the number of trees in the forest gives the high accuracy results. There are many advantages of random forest algorithms. The classifier can handle the missing values. It can also model the random forest classifier for categorical values [10]. The over fitting problem will never come when we use the random forest algorithm in any classification problem. Most importantly it can be used for feature engineering which means identifying the most important feature out of the available feature from the training dataset.

2. **K-Nearest Neighbors-** K-nearest Neighbors is a simple algorithm that stores all available cases and classifies new cases based on a similarity measure [11]. KNN has been used in statistical estimation and pattern recognition. KNN makes prediction for a new instance (x) by searching through the entire training set for the k most similar instances and summarizing the output variable for those k instances. For regression this might be the mean output variable, in classification this might be the mode class determine which of the k instances in the training dataset are most similar to new input many distance measure is used like Euclidean distance, Manhattan distance, Minkowski distance.

3. **K-Nearest Neighbors- K-nearest Neighbors is a simple algorithm that stores all available cases and classifies new cases based on a similarity measure [11]. KNN has been used in statistical estimation and pattern recognition. KNN makes prediction for a new instance (x) by searching through the entire training set for the k most similar instances and summarizing the output variable for those k instances. For regression this might be the mean output variable, in classification this might be the mode class determine which of the k instances in the training dataset are most similar to new input many distance measure is used like Euclidean distance, Manhattan distance, Minkowski distanc**

SYSTEM DESIGN

Modularization details:-

Patient Details:-Doctor can view patient's personal details.

- Patient registration
- Patient login

Notification:Doctor will get notification how many people had accessed the system
and what all are the diseases predicted by the system.

Admin Login: Admin can login to the system using his ID and Password.

- Admin registration
- Admin login

Add Doctor: Admin can add new doctor details into the database.

Add Disease: Admin can add disease details along with symptoms and type.

View Doctor: Admin can view various Doctors along with their personal details.

View Disease:Admin can view various diseases details stored in database.

View Patient: Admin can view various patient details that had accessed the system.

Location Tracker: There will be a location tracker which will track patient's location where he/she accessed the system.

DATA INTEGRITY AND CONSTRAINTS

Database Structure:-

1. Admin Registration
2. Patient Registration

Admin Registration:-

Name	Null/Not Null	Type	Key
Name	Not Null	Varchar(50)	
Email	Null	Varchar(80)	
Phone	Not Null	Int	
Password	Not Null	Varchar(20)	
Confirm Password	Not Null	Varchar(20)	

***Patient Registration*:-**

Name	Null/Not Null	Type	Key
Name	Not Null	Varchar(50)	
Email	Null	Varchar(80)	
Phone	Not Null	Int	
Password	Not Null	Varchar(20)	
Confirm Password	Not Null	Varchar(20)	

TESTING

TESTING is a level of software testing where a complete and integrated software is tested. The purpose of this test is to evaluate the system's compliance with the specified requirements.

The test case designed for the project is discussed below:

Test Case- I: Submit the symptoms from the list.

Precondition: The application is open.

Test steps: 1. Select the checkbox from the list.

2. Select submit.

Expected Result: The symptoms selected should be submitted and further analyzed to calculate the probability of the disease.

SOFTWARE TESTING

Software Testing is a critical element of software quality assurance and represents the ultimate review of specification, design and coding, Testing presents an interesting anomaly for the software engineer.

Testing Objectives include:-

1. Testing is a process of executing a program with the intent of finding an error
2. A good test case is one that has a probability of finding an as yet undiscovered error
3.A successful test is one that uncovers an undiscovered error

Testing Principles:-

- All tests should be traceable to end user requirements
- Tests should be planned long before testing begins
- Testing should begin on a small scale and progress towards testing in large
- Exhaustive testing is not possible

To be most effective testing should be conducted by a independent third party.

Test Planning:-

- Using the test plan as the basis, the testing team design test case specification which then becomes the basis for preparing for individual test cases.
- A test case is nothing but a series of step executed on a product, using a predefined set of input data, expected to produce a pre-defined set of outputs, in a given environment.
- It describes “how” to implement those test cases
- Test case specifications are useful as it enlists the specification details of the items.

Testing Strategies-

A Strategy for software testing integrates software test cases into a series of well-planned steps that result in the successful construction of software. Software testing is a broader topic for what is referred to as Verification and Validation. Verification refers to the set of activities that ensure that the software correctly implements a specific function Validation refers he set of activities that ensure that the software that has been built is traceable to customer’s requirements

- **Unit Testing**:-

Unit testing focuses verification effort on the smallest unit of software design that is the module. Using procedural design description as a guide, important control paths are tested to uncover errors within the boundaries of the module. The unit test is normally white box testing oriented and the step can be conducted in parallel for multiple modules.

- **Integration Testing:-**

Integration testing is a systematic technique for constructing theprogram structure while conducting test to uncover errors associated with the interfacing. The objective is to take unit tested methods and build a program structure that has been dictated by design.

- **Top-down Integration:-**

Top down integrations an incremental approach to construction of program structure. Modules are integrated by moving downwardthrough the control hierarchy, beginning with the main control program. Modules subordinate to the main program are incorporated in the structure either in the breath-first or depth-firstmanner.

- **Bottom-up Integration:-**

This method as the name suggests, begins construction and testing with atomic modules i.e., modules at the lowest leveling the program structure. Because the modules are integrated in the bottom up manner the processing required for the modules subordinate to a given level is always available and the need for stubs is eliminated.

- **Validation Testing:-**

At the end of integration testing software ids completely assembled as a package. Validation testing is the next stage which can be defined as successful when the software functions in the manner reasonably expected by the customer. Reasonable expectations are those defined in the software requirements specifications. Information contained in those sections form a basis for validation testing approach**.**

- **System Testing:-**

System testing is actually a series of different tests whose primary purpose is to fully exercise the computer-based system. Although each test has a different purpose, all work to verify that all system elements have been properly integrated to perform allocated functions.

- **Recovery Testing:-**

It is a system test that forces the system to fail in a variety of ways and verities that the recovery is properly performed.

- **Security Testing:-**

Attempts to verify the protection mechanisms built into the system.

- **Performance Testing:-**

This method is designed to test runtime performance of software within the context of an integrated system..Software testing is a critical element of software quality assurance and represents the ultimate review of specification, design and coding. Testing is the exposure of the system to trial input to see whether it produces correct output.

- **Testing Phases:-**

Software testing phases include the following:
Test activities are determined and test data selected.
The test is conducted and test results are compared with the expected results.

There are various types of Testing:

- **Unit Testing:-**

Unit testing is essentially for the verification of the code produced during the coding phase and the goal is test the internal logic of the module/program.

This project is thoroughly tested by exposing it to the various test cases regarding correct event generation, as this project passed all the tests its quality is completely assured.

❖ Integration Testing:-

All the tested modules are combined into sub systems, which are then tested. The goal is to see if the modules are properly integrated, and the emphasis being on the testing interfaces between the modules. On this project integration testing is done mainly while implementing menus in a sample application such as Browser for Mobiles.

❖ System Testing: -

It is mainly used if the software meets its requirements. The reference document for this process is the requirement document.

❖ Acceptance Testing:-

It is performed with realistic data of the client to demonstrate that thesoftware is working satisfactorily.

❖ Testing Methods:-

Testing is a process of executing a program to find out errors. If testing is conducted successfully, it will uncover all the errors in the software. Any testing can be done basing on two ways:

❖ White Box Testing:-

It is a test case design method that uses the control structures of the procedural design to derive test cases. using this testing a software Engineer can derive the following test cases:
Exercise all the logical decisions on either true or false sides. Execute all loops at their boundaries and within their operational boundaries. Exercise the internal data structures to assure their validity.

❖ Black Box Testing:-

It is a test case design method used on the functional requirements of the software. It will help a software engineer to derive sets of input conditions that will exercise all the functional requirements of the program. Black Box testing attempts to find errors in the following categories:

1. Incorrect or missing functions
2. Interface errors
3. Errors in data structures
4. Performance errors
5. Initialization and termination errors

By Black Box Testing we derive a set of test cases that satisfy the following criteria:

TEST CASE

TEST CASE SPECIFICATION	DESCRIPTION
Test Case ID(TC_ID)	Unique ID to identify/report the bug if present in the functionality of software
Test Case Objective	The purpose of the test. The lists can be generated to perform intended task, for which software is developed. Results should always follow the test case objective
Pre-requisite	This can include environment setup, supporting software environment setup. for the project, or any fields in which user will give the input. So that test cases can be planned accordingly.
Steps	This includes steps to be performed to give the input to the system, so that system can perform its specified task and display the result accordingly. If automated testing is used, then, these steps are translated to the scripting language of the tool.
Input Data	The choice of input data will be depended on the test case itself and the technique followed in the test case. For e.g. equivalence partitioning, boundary value analysis etc.
Expected	It can be the user required output to be shown

Result	
Actual Result	This step should do a comparison of the expected and actual results to highlight any differences.
Status	Whether expected results and actual result match, if it matches then PASS or else FAIL

Test cases that reduce by a count that is greater than one, the number of additional test cases that must be designed to achieve reasonable testing.
Test cases that tell us something about the presence or absence of classes of errors rather than errors associated only with a specific test at hand.

Test Approach:

Testing can be done in two ways:

1. Bottom up approach
2. Top down approach

1. Bottom up Approach:

Testing can be performed starting from smallest and lowest level modules and proceeding one at a time. For each module in bottom up testing a short program executes the module and provides the needed data so that the module is asked to perform the way it will when embedded within the larger system. When bottom level modules are tested attention turns to those on the next level that use the lower level ones they are tested individually and then linked with the previously examined lower level modules.

2. Top down approach:

This type of testing starts from upper level modules. Since the detailed activities usually performed in the lower level routines are not provided stubs are written. A stub is a module shell called by upper level module and that when reached properly will return a message to the calling module indicating that proper interaction occurred. No attempt is made to verify the correctness of the lower level module.

DEBUGGING AND CODE IMPROVEMENT

ADMIN PAGE:-

```
from tkinter import*
from PIL import ImageTk,Image
from subprocess import call

# SCREEN
scr=Tk(className=" AdminPage")

# FOR BACKGROUND IMAGE
i=ImageTk.PhotoImage(Image.open('C:\\Users\\Hackers
world\\Desktop\\projects\\healthprediction\\images\\bgimage.jpg'))
l=Label(scr,image=i)
l.pack()

#for screen size
scr.geometry("{0}x{1}+0+0".format(scr.winfo_screenwidth(),
scr.winfo_screenheight()))

#functions
def exit():
scr.destroy()
   call(["python","home.py"])

def viewpatient():
scr.destroy()
   call(['python','viewpatient.py'])

def viewdisease():
scr.destroy()
   call(['python','viewdisease.py'])
```

```
def adddisease():
scr.destroy()
   call(['python','adddisease.py'])

# MENU BUTTONS
logoutbutton=Button(master=scr,text='LOGOUT', fg='white',font=("Times", "18", "bold"),bg='darkcyan',activebackground='green',activeforeground='black',command=exit)
logoutbutton.place(width=1400,height=40,x=0,y=0)


# frame
fr=Frame(master=scr,width=1000,height=250,bd=1,relief=RIDGE)
fr.place(x=200,y=300)
fr1=Frame(master=scr,width=500,height=30,bg='#33ccff',bd=0,relief=RIDGE)
fr1.place(x=450,y=560)
fr2=Frame(master=scr,width=250,height=20,bg='#3399ff',bd=0,relief=RIDGE)
fr2.place(x=575,y=600)
fr3=Frame(master=scr,width=125,height=15,bg='#3366ff',bd=0,relief=RIDGE)
fr3.place(x=637,y=630)
fr4=Frame(master=scr,width=62,height=7,bg='#3333ff',bd=0,relief=RIDGE)
fr4.place(x=668,y=655)
fr5=Frame(master=scr,width=31,height=3,bg='#3333ff',bd=0,relief=RIDGE)
fr5.place(x=683,y=670)



# FOR BACKGROUND IMAGE
i1=ImageTk.PhotoImage(Image.open('C:\\Users\\Hackers world\\Desktop\\projects\\healthprediction\\images\\frame.jpg'))
l1=Label(fr,image=i1)
l1.pack(
# WIDGETS(FUNCTIONS) BUTTONS
addd=Button(master=scr, text='ADD
DISEASE',bg='red',fg='white',bd=2,relief=SUNKEN,font=("Times", "18",
"bold"),command=adddisease)
viewdisease=Button(master=scr, text='VIEW
DISEASE',bg='red',fg='white',bd=2,relief=SUNKEN,font=("Times", "18",
"bold"),command=viewdisease)
viewpatient=Button(master=scr, text='VIEW
PATIENT',bg='red',fg='white',bd=2,relief=SUNKEN,font=("Times", "18",
"bold"),command=viewpatient)
addd.place(width=200,height=200,x=235,y=325)
viewdisease.place(width=200,height=200,x=590,y=325)
viewpatient.place(width=200,height=200,x=970,y=325)



scr.mainloop()
```

ADMIN LOGIN PAGE:-

```
from tkinter import*
from subprocess import call
import sqlite3 as s
from tkinter import messagebox
from PIL import ImageTk,Image


adminloginscr=Tk(className=' Admin Login')
adminloginscr.geometry('300x180')
adminloginscr.maxsize(width=300,height=180)


# FOR BACKGROUND IMAGE
```

```
i=ImageTk.PhotoImage(Image.open('C:\\Users\\Hackers
world\\Desktop\\projects\\healthprediction\\images\\3.jpg'))
l=Label(adminloginscr,image=i)
l.pack()

# DATABASE FOR LOGIN
try:
    client=s.connect("C://Users//Hackers
world//Desktop//projects//healthprediction//register.db")
    cu=client.cursor()
cu.execute("create table aregister(name varchar(50),email varchar(80),phone
int,password varchar(20),confirmpassword varchar(20))")
except:
    pass

def adminpage():
cu.execute("select count(*) from aregister where Name=%r and
Password=%r"%(id_t.get(),pswd_t.get()))
        a=cu.fetchall()
        if a[0][0]==1:
messagebox.showinfo('Successful','LoginSucessfull')
adminloginscr.destroy()
            call(["python","adminpage.py"])
        else:
messagebox.showinfo('not found','Username or Password not found')




def destroy():
adminloginscr.destroy()
    call(['python','home.py'])
```

```
# LABELS
heading=Label(master=adminloginscr,text='ADMIN    LOGIN',bg='darkcyan',fg='white')
id=Label(master=adminloginscr,text='User_Id    ',bg='darkcyan',fg='white')
pswd=Label(master=adminloginscr,text='Password',bg='darkcyan',fg='white')
heading.place(x=0,y=0,width=300)
id.place(x=25,y=45)
pswd.place(x=25,y=75)

# ENTRY BOX
id_t=Entry(master=adminloginscr,width=30)
pswd_t=Entry(master=adminloginscr,show="*",width=30)
id_t.place(x=90,y=45)
pswd_t.place(x=90,y=75)

# LOGIN BUTTON
login=Button(master=adminloginscr,text='LOGIN',bg='darkcyan',fg='white',command=adminpage)
login.place(x=90,y=100)

exitb=Button(master=adminloginscr,text='BACK',bg='darkcyan',fg='white',command=destroy)
exitb.place(x=150,y=100)




adminloginscr.mainloop()
```

ADMIN REGISTERATION PAGE:-

```
from tkinter import *
from subprocess import call
from subprocess import call
from PIL import ImageTk,Image
from tkinter import messagebox
import sqlite3 as s

scr=Tk(className='Admin Registeration')

# FOR BACKGROUND IMAGE
i=ImageTk.PhotoImage(Image.open('C:\\Users\\Hackers
world\\Desktop\\projects\\healthprediction\\images\\bgimage.jpg'))
l=Label(scr,image=i)
l.pack()

#for screen size
scr.geometry("{0}x{1}+0+0".format(scr.winfo_screenwidth(),
scr.winfo_screenheight()))

try:
   client=s.connect("C://Users//Hackers
world//Desktop//projects//healthprediction//register.db")
   cu=client.cursor()
cu.execute("create table aregister(name varchar(50),email varchar(80),phone
int,password varchar(20),confirmpassword varchar(20))")
except:
   pass

# login button function to redirect to the dashboard
def register():
   if en3.get()==en4.get():
cu.execute("insert into aregister
values(%r,%r,%d,%r,%r)"%(en.get(),en1.get(),int(en2.get()),en3.get(),en4.get()))
client.commit()
scr.destroy()
      call(['python','home.py'])
```

```
  else:
messagebox.showinfo('Error','password does not match')


def destroy():
scr.destroy()
  call(['python','home.py'])


a=Label(scr,text='Name        ',fg='black',bd=1,relief=RIDGE,font=('Time',24,'bold'))
a1=Label(scr,text='Email        ',bd=1,relief=RIDGE,fg='black',font=('Time',24,'bold'))
a2=Label(scr,text='Phone       ',bd=1,relief=RIDGE,fg='black',font=('Time',24,'bold'))
a3=Label(scr,text='Password ',bd=1,relief=RIDGE,fg='black',font=('Time',24,'bold'))
a4=Label(scr,text='Confirm    ',bd=1,relief=RIDGE,fg='black',font=('Time',24,'bold'))
en=Entry(scr,font=('default',24),fg='blue',bd=5)
en1=Entry(scr,font=('default',24),fg='blue',bd=5)
en2=Entry(scr,font=('default',24),fg='blue',bd=5)
en3=Entry(scr,font=('default',24),fg='blue',bd=5,show='*')
en4=Entry(scr,font=('default',24),fg='blue',bd=5,show='*')
bu=Button(scr,text=' REGISTER
',font=('default',32),bg='red',fg='White',command=register)
a5=Label(scr,text='ADMIN    REGISTRATION
',width=46,fg='yellow',bg='black',font=('Arial Black',32,'bold'))
exitb=Button(scr,text='BACK',font=('default',32),bg='green',fg='White',command=destr
oy)
exitb.place(x=890,y=600)
a5.place(x=0,y=0)
a.place(x=300,y=130)
a1.place(x=300,y=230)
a2.place(x=300,y=330)
a3.place(x=300,y=430)
a4.place(x=300,y=530)
en.place(x=600,y=130)
```

```
en1.place(x=600,y=230)
en2.place(x=600,y=330)
en3.place(x=600,y=430)
en4.place(x=600,y=530)
bu.place(x=600,y=600)



scr.mainloop()
```

ADMIN HOMEPAGE:-

```
from tkinter import *
from PIL import Image,ImageTk
from subprocess import call




class screen():
    # SUBPROCESS CALL FUNCTION FOR CALLING OTHER MODULE IN OPEN MODE
    def fun(self):
self.ms.destroy()
call(["python", "adminlogin.py"])

    def fun1(self):
self.ms.destroy()
call(["python", "patientlogin.py"])

    def adminregister(self):
self.ms.destroy()
        call(["python","adminregister.py"])

    def patientregister(self):
self.ms.destroy()
        call(["python","patientregistration.py"])
    def __init__(self):
        self.ms=Tk(className=" Health Prediction")
self.image=Image.open("C:\\Users\\Hackers
world\\Desktop\\projects\\healthprediction\\images\\dentist.png")
self.photo=ImageTk.PhotoImage(self.image)
self.lbl1=Label(image=self.photo)
self.lbl1.pack()
self.frame1 = Frame(master=self.ms, highlightbackground="black",
highlightcolor="black", highlightthickness=1, width=400, height=150, bd= 0)
self.frame1.place(x=300,y=550)

self.lbl4=Label(master=self.ms,text="Admin",bd=1,relief=SOLID,font=("Times", "30",
"bold"))
self.lbl4.place(x=425,y=520)
self.btn1=Button(master=self.ms,text="Register",font=("Times", "20",
"bold"),bg="red",fg="white",command=self.adminregister)
self.btn1.place(x=320,y=600)
self.btn2=Button(master=self.ms,text="Login",font=("Times", "20",
"bold"),bg="Red",fg="white",command=self.fun)
self.btn2.place(x=570,y=600)
self.frame2 = Frame(master=self.ms, highlightbackground="black",
highlightcolor="black", highlightthickness=1, width=400, height=150, bd= 0)
self.frame2.place(x=800,y=550)

self.lbl4=Label(master=self.ms,text="Patient",bd=1,relief=SOLID,font=("Times", "30",
"bold"))
self.lbl4.place(x=930,y=520)
self.btn3=Button(master=self.ms,text="Register",font=("Times", "20",
"bold"),bg="red",fg="white",command=self.patientregister)
self.btn3.place(x=820,y=600)
self.btn4=Button(master=self.ms,text="Login",font=("Times", "20",
"bold"),bg="red",fg="white",command=self.fun1)
self.btn4.place(x=1070,y=600)
self.btn5=Button(master=self.ms,text="Exit",font=("Times", "20",
"bold"),bg="white",fg="black",command=lambda:self.ms.destroy())
self.btn5.place(x=1380,y=650)
self.ms.mainloop()
scr=screen()
```

PATIENT HOMEPAGE:-

```
from tkinter import*
from subprocess import call
from PIL import ImageTk,Image



# FUNCTIONS
def cleancode():
scr.destroy()
    call(['python','clean_code.py'])

def otherdisease():
scr.destroy()
    call(['python','addedprediction.py'])




scr=Tk(className='choose disease type')

# FOR BACKGROUND IMAGE
i=ImageTk.PhotoImage(Image.open('C:\\Users\\Hackers
world\\Desktop\\projects\\healthprediction\\images\\bgimage.jpg'))
l=Label(scr,image=i)
l.pack()
```

```
#for screen size
scr.geometry("{0}x{1}+0+0".format(scr.winfo_screenwidth(),
scr.winfo_screenheight()))

label=Label(scr,text='Choose Your Disease Prediction
Category',width=80,bg='darkcyan',fg='white',font=('Helvetica',24,'bold'))
b1=Button(master=scr,width=30,bg='red',fg='white',text='GENERAL
DISEASE',font=('Time',24,'bold'),command=cleancode)
#b2=Button(master=scr,width=30,bg='red',fg='white',text='RETINOPATHY',font=('Tim
e',24,'bold'))
b3=Button(master=scr,width=30,bg='red',fg='white',text='OTHER
DISEASES',font=('Time',24,'bold'),command=otherdisease)
label.place(x=0,y=0)
b1.place(x=480,y=300)
#b2.place(x=480,y=400)
b3.place(x=480,y=500)



scr.mainloop()
```

GENERAL DISEASE PREDICTION:-

```
from tkinter import *
import numpy as np
import pandas as pd
from subprocess import call
from PIL import ImageTk,Image
import pyttsx3
import datetime
import speech_recognition as sr
import sqlite3 as s
```

```
# from gui_stuff import *
df=pd.read_csv("Training.csv")
tr=pd.read_csv("Testing.csv")


l1=['itching','skinrash','nodal skin
eruptions','continuoussneezing','shivering','chills','jointpain','stomach pain',
'acidity','ulcers on
tongue','musclewasting','vomiting','burningmicturition','spottingurination','fatigue',
'weight gain','anxiety','cold hands and feets','mood
swings','weightloss','restlessness','lethargy','patches in throat',
'irregular sugar
level','cough','highfever','sunkeneyes','breathlessness','sweating','dehydration','indigesti
on',
'headache','yellowishskin','darkurine','nausea','loss of appetite','pain behind the
eyes','backpain','constipation',
'abdominal pain','diarrhoea','mild fever','yellowurine','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','sinuspressure','runnynose','congestion','chestpain','weakness in
limbs','fast heart rate',
'pain during bowel movements','pain in anal region','bloody stool',
'irritation in
anus','neckpain','dizziness','cramps','bruising','obesity','swollenlegs','swollen blood
vessels',
'puffy face and eyes','enlargedthyroid','brittlenails','swollenextremeties','excessive
hunger',
'extra marital contacts','drying and tingling lips','slurredspeech','kneepain','hip joint
pain','muscle weakness',
'stiff neck','swellingjoints','movementstiffness','spinningmovements','loss of
balance','unsteadiness',
'weakness of one body side','loss of smell','bladderdiscomfort','foul smell of
urine','continuous feel of urine',
'passage of gases','internalitching','toxic look (typhos)','depression','irritability','muscle
pain',
'altered sensorium','red spots over body','bellypain','abnormal
menstruation','dischromic patches',
'watering from eyes','increasedappetite','polyuria','familyhistory','mucoidsputum','rusty
sputum',
'lack of concentration','visualdisturbances','receiving blood transfusion','receiving
unsterile injections',
'coma','stomachbleeding','distention of abdomen','history of alcohol
consumption','fluidoverload','blood in sputum',
'prominent veins on calf','palpitations','painfulwalking','pus filled
pimples','blackheads','scurring','skin peeling',
'silver like dusting','small dents in nails','inflammatorynails','blister','red sore around
nose','yellow crust ooze']
```

```
disease=['Fungal infection','Allergy','GERD','Chroniccholestasis','Drug Reaction',
'Peptic ulcer diseae','AIDS','Diabetes','Gastroenteritis','BronchialAsthma','Hypertension',
' Migraine','Cervical spondylosis',
'Paralysis (brain
hemorrhage)','Jaundice','Malaria','Chickenpox','Dengue','Typhoid','hepatitis A',
'Hepatitis B','HepatitisC','HepatitisD','HepatitisE','Alcoholichepatitis','Tuberculosis',
'Common Cold','Pneumonia','Dimorphichemmorhoids(piles)',
'Heartattack','Varicoseveins','Hypothyroidism','Hyperthyroidism','Hypoglycemia','Osteo
arthristis',
'Arthritis','(vertigo) Paroymsal  PositionalVertigo','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.columns = df.columns.str.replace('dischromic _patches', 'dischromic patches')
df.columns = df.columns.str.replace('spotting_ urination','spotting urination' )
df.columns = df.columns.str.replace('_', ' ')



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.columns = tr.columns.str.replace('dischromic _patches', 'dischromic patches')
tr.columns = tr.columns.str.replace('spotting_ urination','spotting urination' )
tr.columns = tr.columns.str.replace('_', ' ')

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)
# -----------------------------------------------------------------------------------------------------
lst=[]
diseaselist=[]
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)
    pre=(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")
lst.append(int(pre))
diseaselist.append(disease[a])




def randomforest():
    from sklearn.ensemble import RandomForestClassifier
    clf4 = RandomForestClassifier()
    clf4 = clf4.fit(X,np.ravel(y))

    # calculating accuracy------------------------------------------------------------------
    from sklearn.metrics import accuracy_score
y_pred=clf4.predict(X_test)
print(accuracy_score(y_test, y_pred))
    pre1=(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 = clf4.predict(inputtest)
    predicted=predict[0]

    h='no'
    for b in range(0,len(disease)):
```

```
if(predicted == b):
            h='yes'
            break

    if (h=='yes'):
        t2.delete("1.0", END)
        t2.insert(END, disease[b])
    else:
        t2.delete("1.0", END)
        t2.insert(END, "Not Found")
lst.append(int(pre1))
diseaselist.append(disease[b])

def NaiveBayes():
    try:
        client=s.connect("C://Users//Hackers
world//Desktop//projects//healthprediction//register.db")
        cu=client.cursor()
cu.execute("create table patient(name varchar(50),disease varchar(80))")
    except:
        pass

    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))
    pre2=(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 c in range(0,len(disease)):
if(predicted == c):
      h='yes'
      break

  if (h=='yes'):
    t3.delete("1.0", END)
    t3.insert(END, disease[c])
  else:
    t3.delete("1.0", END)
    t3.insert(END, "Not Found")
lst.append(int(pre2))
diseaselist.append(disease[c])
  print(diseaselist)
  t4.insert(END,max(diseaselist))
cu.execute("insert into patient values(%r,%r)"%(NameEn.get(),diseaselist[2]))
client.commit()

# gui_stuff--------------------------------------------------------------------------------

root = Tk(className=' Disease Predictor')
```

```
root.geometry('450x740')
root.maxsize(740,450)
root.minsize(740,450)



# FOR BACKGROUND IMAGE
i=ImageTk.PhotoImage(Image.open('C:\\Users\\Hackers
world\\Desktop\\projects\\healthprediction\\images\\png.png'))
l=Label(root,image=i)
l.grid(row=0,column=0)
# 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()

def sym1():
Symptom1.set(None)
def sym2():
Symptom2.set(None)
def sym3():
Symptom3.set(None)
def sym4():
Symptom4.set(None)
def sym5():
```

```
Symptom5.set(None)

#logout function
def logoutfun():
root.destroy()
    call(["python","home.py"])



# Heading
w2 = Label(root,width=62,height=1,justify='center', text=" Disease Predictor
",font=('times',15,'bold'), fg="white",bg='darkcyan')

w2.place(x=0,y=0)

# labels
NameLb = Label(root, text="Patient Name",
font=('times',15,'bold'),fg="white",bg='darkcyan')
NameLb.place(x=10,y=72)



S1Lb = Label(root, text="Symptom 1    ",
font=('times',15,'bold'),fg="white",bg='darkcyan')
S1Lb.place(x=10,y=118)

S2Lb = Label(root, text="Symptom 2    ",
font=('times',15,'bold'),fg="white",bg='darkcyan')
S2Lb.place(x=10,y=158)

S3Lb = Label(root, text="Symptom 3    ",
font=('times',15,'bold'),fg="white",bg='darkcyan')
S3Lb.place(x=10,y=198)
```

```
S4Lb = Label(root, text="Symptom 4    ",font=('times',15,'bold'),
fg="white",bg='darkcyan')
S4Lb.place(x=10,y=238)

S5Lb = Label(root, text="Symptom 5    ",font=('times',15,'bold'),
fg="white",bg='darkcyan')
S5Lb.place(x=10,y=278)



lrLb = Label(root, text="DecisionTree", font=('times',15,'bold'),fg="white",bg='green')
lrLb.place(x=10,y=322)

destreeLb = Label(root, text="Rand_Forest ",
font=('times',15,'bold'),fg="white",bg='green')
destreeLb.place(x=10,y=362)

ranfLb = Label(root, text="NaiveBayes  ",font=('times',15,'bold'), fg="white",bg='green')
ranfLb.place(x=10,y=402)
# entries
OPTIONS = sorted(l1)

NameEn = Entry(root, textvariable=Name,width=25)
NameEn.place(x=140,y=73)

S1En = OptionMenu(root, Symptom1,*OPTIONS)
S1En.place(x=140,y=113)

S2En = OptionMenu(root, Symptom2,*OPTIONS)
S2En.place(x=140,y=153)



S3En = OptionMenu(root, Symptom3,*OPTIONS)
S3En.place(x=140,y=193)
```

```
S4En = OptionMenu(root, Symptom4,*OPTIONS)
S4En.place(x=140,y=233)


S5En = OptionMenu(root, Symptom5,*OPTIONS)
S5En.place(x=140,y=273)

# clear button for option menu
btn1=Button(root,text='CLEAR',command=sym1)
btn1.place(x=420,y=116)
btn2=Button(root,text='CLEAR',command=sym2)
btn2.place(x=420,y=156)
btn3=Button(root,text='CLEAR',command=sym3)
btn3.place(x=420,y=196)
btn4=Button(root,text='CLEAR',command=sym4)
btn4.place(x=420,y=236)
btn5=Button(root,text='CLEAR',command=sym5)
btn5.place(x=420,y=276)



dst = Button(root, text="DecisionTree ",
font=('times',15,'bold'),command=DecisionTree,bg="green",fg="white")
dst.place(x=600,y=113)

rnf = Button(root, text="Randomforest",
font=('times',15,'bold'),command=randomforest,bg="green",fg="white")
rnf.place(x=600,y=173)

lr = Button(root, text="NaiveBayes    ",font=('times',15,'bold'),
command=NaiveBayes,bg="green",fg="white")
```

```
lr.place(x=600,y=233)

#textfileds
t1 = Text(root, height=1, width=40,bg="orange",fg="black")
t1.place(x=150,y=323)

t2 = Text(root, height=1, width=40,bg="orange",fg="black")
t2.place(x=150,y=363)

t3 = Text(root, height=1, width=40,bg="orange",fg="black")
t3.place(x=150,y=403)
fl=Label(root,text='   FINAL PREDICTION
',font=('times',10,'bold'),bg='darkcyan',fg='black')
fl.place(x=600,y=283)
t4=Text(root,width=18,height=40)
t4.place(x=600,y=303)



# Speech recognintion

def SpeechRecog():

    #voices
    engine=pyttsx3.init('sapi5')
    voices=engine.getProperty('voices')
engine.setProperty('voice',voices[0].id)

    #for speak
    def speak (audio):
engine.say(audio)
engine.runAndWait()

    #wishing
```

```
  def wishme():
speak("hello, this is disease predictor")


  # it takes microphone input from user and return string output
  def takeCommand():
    r=sr.Recognizer()
    with sr.Microphone() as source:
speak("please tell your name.")
print('listening...')
r.pause_threshold=1
      audio=r.listen(source)
    try:
print('recogninzing...')
      query=r.recognize_google(audio,language='en-in')
      print(query)
Name.set(query)
    except Exception as e:
print('say that again please...')
      return takeCommand()

nolist=['first','second','third','fourth','fifth']
symoptions=list(set(OPTIONS))

    def symptomsrec():
      r=sr.Recognizer()
      with sr.Microphone() as source:
        for i in range(len(nolist)):
speak('your'+ nolist[i]+ 'symptom please')
print('listening..')
r.pause_threshold=1
          audio=r.listen(source)
```

```
                    try:
print('recongnizing...')
                        query=r.recognize_google(audio,language='en-in')
                        print(query)
                        if query in symoptions:
                            if i==0:
                                Symptom1.set(query)
                            if i==1:
                                Symptom2.set(query)
                            if i==2:
                                Symptom3.set(query)
                            if i==3:
                                Symptom4.set(query)
                            if i==4:
                                Symptom5.set(query)
                    except:
print('say that again please...')

speak('do you want to fill your symptoms by using me.')
        r=sr.Recognizer()
        with sr.Microphone() as source:
print('listening..')
r.pause_threshold=1
            audio=r.listen(source)
print('recognizing..')

            query1=r.recognize_google(audio,language='en-in')

mainquery=[]
mainquery.append(query1)
            if 'yes' in mainquery:
                print(mainquery)
symptomsrec()
```

```
        else:
            pass



    #main function
    if __name__=="__main__":
wishme()
takeCommand()
DecisionTree()
randomforest()
NaiveBayes()



#mic
mic=ImageTk.PhotoImage(Image.open('C:\\Users\\Hackers
world\\Desktop\\projects\\healthprediction\\images\\mic.jpg'))
microphone=Button(master=root,width=13,height=13,image=mic,relief='flat',command
=SpeechRecog)
microphone.place(x=300,y=74)
#logout
logout=ImageTk.PhotoImage(Image.open('C:\\Users\\Hackers
world\\Desktop\\projects\\healthprediction\\images\\logoutbtn.jpg'))
log_out=Button(master=root,width=41,height=41,image=logout,relief='flat',command=l
ogoutfun)
log_out.place(x=600,y=30)

btn=Button(master=root,width=9,height=2,bg='green',fg='white',relief='flat',text='LOG
OUT',font=('times',11,'bold'),command=logoutfun)
btn.place(x=645,y=30)



root.mainloop()
```

OTHER DISEASEPREDICTION:-

```
from tkinter import*
from subprocess import call
from PIL import ImageTk,Image
import sqlite3 as s


# DATABASE FOR LOGIN
try:
   client=s.connect("C://Users//Hackers
world//Desktop//projects//healthprediction//register.db")
   cu=client.cursor()
cu.execute("create table adddisease(disease_name varchar(50),sym1 varchar(80),sym2
varchar(80),sym3 varchar(80),sym4 varchar(80),sym5 varchar(80),sym6 varchar(80)")
except:
   pass


cu.execute("select * from adddisease")
a=list(cu.fetchall())


mydict={}
for i in range(1,(len(a)+1)):
   b=list(a[i-1])
   b=[j.replace('\\n','') for j in b]
   mydict[str(b[0])]=[str(b[1]),str(b[2]),str(b[3]),str(b[4]),str(b[5]),str(b[6])]
print(mydict)



# FUNCTIONS
```

```
def button():
    for k, v in mydict.items():
if  e1.get() and e2.get() and e3.get() and e4.get() in v:
            print(k)
elif  e1.get() and e2.get() and e3.get() in v:
            print(k)
elif  e1.get() and e2.get() in v:
            print(k)
elif  e1.get() in v:
            print(k)
text.insert(END,k)
def back():
scr.destroy()
    call(['python','diseasetypes.py'])

# gui_stuff--------------------------------------------------------------------------------
scr = Tk(className=' Added Disease Predictor')
scr.geometry('450x540')
scr.maxsize(540,450)
scr.minsize(540,450)

# FOR BACKGROUND IMAGE
i=ImageTk.PhotoImage(Image.open('C:\\Users\\Hackers
world\\Desktop\\projects\\healthprediction\\images\\magni.png'))
l=Label(scr,image=i)
l.pack()

# LABEL
l1=Label(scr,text='symptom 1')
l2=Label(scr,text='symptom 2')
l3=Label(scr,text='symptom 3')
l4=Label(scr,text='symptom 4')
l1.place(x=100,y=0)
```

```
l2.place(x=100,y=25)
l3.place(x=100,y=50)
l4.place(x=100,y=75)

# ENTRY
e1=Entry(scr,width=50,bg='cyan')
e1.place(x=200,y=0)
e2=Entry(scr,width=50,bg='cyan')
e2.place(x=200,y=25)
e3=Entry(scr,width=50,bg='cyan')
e3.place(x=200,y=50)
e4=Entry(scr,width=50,bg='cyan')
e4.place(x=200,y=75)



# TEXT ENTRY
text=Text(scr,width=37,height=10)
text.place(x=200,y=120)

# BUTTON
b=Button(scr,text='PREDICT',command=button,font=('Time',15,'bold'),bg='darkcyan',fg
='white')
b.place(x=200,y=300)
b1=Button(scr,text='BACK',command=back,font=('Time',15,'bold'),bg='darkcyan',fg='w
hite')
b1.place(x=350,y=300)
scr.mainloop()
```

<u>ADD DISEASE:-</u>

```
from tkinter import*
from subprocess import call
from PIL import ImageTk,Image
from tkinter import messagebox
import sqlite3 as s

scr=Tk(className='Add Disease')



# DATABASE FOR REGISTER
try:
   client=s.connect("C://Users//Hackers
world//Desktop//projects//healthprediction//register.db")
   cu=client.cursor()
cu.execute("create table adddisease(disease_name varchar(50),sym1 varchar(80),sym2
varchar(80),sym3 varchar(80),sym4 varchar(80),sym5 varchar(80),sym6 varchar(80)")
except:
   pass



# FOR BACKGROUND IMAGE
i=ImageTk.PhotoImage(Image.open('C:\\Users\\Hackers
world\\Desktop\\projects\\healthprediction\\images\\addis.png'))
l=Label(scr,image=i)
l.pack()

#for screen size
scr.geometry("{0}x{1}+0+0".format(scr.winfo_screenwidth(),
scr.winfo_screenheight()))



def register():
cu.execute("insert into adddisease
values(%r,%r,%r,%r,%r,%r,%r)"%(e1.get("1.0","end"),e2.get("1.0","end"),e3.get("1.0","
end"),e4.get("1.0","end"),e5.get("1.0","end"),e6.get("1.0","end"),e7.get("1.0","end")))
client.commit()
```

```
messagebox.showinfo('SUCCESS','Diseasesuccessfuly added')
scr.destroy()
        call(['python','adminpage.py'])



# entry box
e1=Text(scr,relief=RIDGE,height=2,width=45,bg='lightcyan')

e2=Text(scr,relief=RIDGE,height=2,width=45,bg='yellow')
e3=Text(scr,relief=RIDGE,height=2,width=45,bg='yellow')
e4=Text(scr,relief=RIDGE,height=2,width=45,bg='yellow')
e5=Text(scr,relief=RIDGE,height=2,width=45,bg='yellow')
e6=Text(scr,relief=RIDGE,height=2,width=45,bg='yellow')
e7=Text(scr,relief=RIDGE,height=2,width=45,bg='yellow')

e1.place(x=10,y=200)
e2.place(x=500,y=200)
e3.place(x=930,y=200)
e4.place(x=500,y=400)
e5.place(x=930,y=400)
e6.place(x=500,y=600)
e7.place(x=930,y=600)

b=Button(scr,text='ADD',width=5,bg='red',fg='white',font=('Time',15,'bold'),command=
register)
b.place(x=930,y=690)

scr.mainloop()
```

PATIENT LOGINPAGE:-

```
from tkinter import*
from subprocess import call
```

```
import sqlite3 as s
from tkinter import messagebox
from PIL import ImageTk,Image



patientloginscr=Tk(className=' Patient Login')
patientloginscr.geometry('300x180')
patientloginscr.maxsize(width=300,height=180)

# FOR BACKGROUND IMAGE
i=ImageTk.PhotoImage(Image.open('C:\\Users\\Hackers
world\\Desktop\\projects\\healthprediction\\images\\3.jpg'))
l=Label(patientloginscr,image=i)
l.pack()



# DATABASE FOR LOGIN
try:
  client=s.connect("C://Users//Hackers
world//Desktop//projects//healthprediction//register.db")
  cu=client.cursor()
cu.execute("create table pregister(name varchar(50),email varchar(80),phone
int,password varchar(20),confirmpassword varchar(20))")
except:
  pass

def diseasetype():
cu.execute("select count(*) from pregister where Name=%r and
Password=%r"%(id_t.get(),pswd_t.get()))
    a=cu.fetchall()
    if a[0][0]==1:
messagebox.showinfo('Successful','LoginSucessfull')
patientloginscr.destroy()
```

```
          call(["python","diseasetypes.py"])
        else:
messagebox.showinfo('not found','Username or Password not found')



def destroy():
patientloginscr.destroy()
   call(['python','home.py'])



# LABELS
heading=Label(master=patientloginscr,text='PATIENT
LOGIN',bg='darkcyan',fg='white')
id=Label(master=patientloginscr,text='User_Id    ',relief=FLAT,bd=2,
bg='darkcyan',fg='white')
pswd=Label(master=patientloginscr,text='Password',relief=FLAT,bd=2,bg='darkcyan',fg
='white')
heading.place(x=0,y=0,width=300)
id.place(x=25,y=45)
pswd.place(x=25,y=75)

# ENTRY BOX
id_t=Entry(master=patientloginscr,width=30)
pswd_t=Entry(master=patientloginscr,show="*",width=30)
id_t.place(x=90,y=45)
pswd_t.place(x=90,y=75)

# LOGIN BUTTON
login=Button(master=patientloginscr,text='LOGIN',bg='darkcyan',fg='white',command=
diseasetype)
login.place(x=90,y=100)
exitb=Button(master=patientloginscr,text='BACK',bg='darkcyan',fg='white',command=d
estroy)
```

```
exitb.place(x=150,y=100)




patientloginscr.mainloop()
```

PATIENT REGISTRATION:-

```
from tkinter import *
from subprocess import call
from subprocess import call
from PIL import ImageTk,Image
import sqlite3 as s
from tkinter import messagebox

scr=Tk(className='Patient Registeration')

# FOR BACKGROUND IMAGE
i=ImageTk.PhotoImage(Image.open('C:\\Users\\Hackers
world\\Desktop\\projects\\healthprediction\\images\\bgimage.jpg'))
l=Label(scr,image=i)
l.pack()

#for screen size
scr.geometry("{0}x{1}+0+0".format(scr.winfo_screenwidth(),
scr.winfo_screenheight()))

# DATABASE FOR REGISTER
try:
    client=s.connect("C://Users//Hackers
world//Desktop//projects//healthprediction//register.db")
```

```
    cu=client.cursor()
cu.execute("create table pregister(name varchar(50),email varchar(80),phone
int,password varchar(20),confirmpassword varchar(20))")
except:
    pass

# login button function to redirect to the dashboard
def register():
        if en3.get()==en4.get():
cu.execute("insert into pregister
values(%r,%r,%d,%r,%r)"%(en.get(),en1.get(),int(en2.get()),en3.get(),en4.get()))
client.commit()
scr.destroy()
            call(['python','patientlogin.py'])
        else:
messagebox.showinfo('Error','password does not match')

def destroy():
scr.destroy()
    call(['python','home.py'])




a=Label(scr,text='Name       ',fg='black',font=('Time',24,'bold'))
a1=Label(scr,text='Email       ',fg='black',font=('Time',24,'bold'))
a2=Label(scr,text='Phone      ',fg='black',font=('Time',24,'bold'))
a3=Label(scr,text='Password',fg='black',font=('Time',24,'bold'))
a4=Label(scr,text='Confirm   ',fg='black',font=('Time',24,'bold'))
en=Entry(scr,font=('default',24),fg='blue',bd=5)
en1=Entry(scr,font=('default',24),fg='blue',bd=5)
en2=Entry(scr,font=('default',24),fg='blue',bd=5)
en3=Entry(scr,font=('default',24),fg='blue',bd=5,show='*')
```

```
en4=Entry(scr,font=('default',24),fg='blue',bd=5,show='*')
bu=Button(scr,text=' REGISTER
',font=('default',32),bg='red',fg='White',command=register)
exitb=Button(scr,text='BACK',font=('default',32),bg='green',fg='White',command=destr
oy)
exitb.place(x=890,y=600)
a5=Label(scr,text='PATIENT REGISTER',width=45,bg='black',fg='yellow',font=('Arial
Black',32,'bold'))
a5.place(x=0,y=0)
a.place(x=300,y=130)
a1.place(x=300,y=230)
a2.place(x=300,y=330)
a3.place(x=300,y=430)
a4.place(x=300,y=530)
en.place(x=600,y=130)
en1.place(x=600,y=230)
en2.place(x=600,y=330)
en3.place(x=600,y=430)
en4.place(x=600,y=530)
bu.place(x=600,y=600)


scr.mainloop()
```

VIEW DISEASE:-

```
from tkinter import *
import numpy as np
import pandas as pd
from subprocess import call
from PIL import ImageTk,Image
import sqlite3 as s

# functions
```

```
def adminpage():
scr.destroy()
   call(['python','adminpage.py'])



tr=pd.read_csv("Testing.csv")
disease=(list(set(tr['prognosis'])))

# GUI TKINTER
scr=Tk(className='Diseases')
#for screen size
scr.geometry('450x740')
scr.maxsize(740,450)
scr.minsize(740,450)

# FRAME
frame=Frame(scr,width=740,height=450,bg='darkcyan')
frame.place(x=0,y=0)

# LABEL
text=Label(master=scr,font=('times',17,'bold'),bg='darkcyan',fg='white',bd=0,relief=RID
GE,text="\t\t           GENERAL DISEASES")
text.place(x=0,y=0)

# Vertical (y) Scroll Bar
scroll = Scrollbar(scr)
scroll.pack(side=RIGHT, fill=Y)

# Text Widget
frame1=Text(scr,bg='#D3D3D3',width=90,height=26,bd=0)
frame1.place(x=0,y=40)

# Configure the scrollbars
```

```
scroll.config(command=frame1.yview)

for i in range(len(disease)):
   frame1.insert(END,'  '+disease[i]+'\n')



btn=Button(master=scr,text='BACK',command=adminpage)
btn.place(x=670,y=420)

scr.mainloop()
```

VIEW PATIENT:-

```
from tkinter import *
import numpy as np
import pandas as pd
from subprocess import call
from PIL import ImageTk,Image
import sqlite3 as s

# functions
def adminpage():
scr.destroy()
   call(['python','adminpage.py'])



tr=pd.read_csv("Testing.csv")
disease=(list(set(tr['prognosis'])))

# GUI TKINTER
scr=Tk(className='Diseases')
#for screen size
scr.geometry('450x740')
scr.maxsize(740,450)
scr.minsize(740,450)

# FRAME
frame=Frame(scr,width=740,height=450,bg='darkcyan')
frame.place(x=0,y=0)

# LABEL
text=Label(master=scr,font=('times',17,'bold'),bg='darkcyan',fg='white',bd=0,relief=RID
GE,text="\t\t          GENERAL DISEASES")
text.place(x=0,y=0)

# Vertical (y) Scroll Bar
scroll = Scrollbar(scr)
scroll.pack(side=RIGHT, fill=Y)

# Text Widget
frame1=Text(scr,bg='#D3D3D3',width=90,height=26,bd=0)
frame1.place(x=0,y=40)

# Configure the scrollbars
scroll.config(command=frame1.yview)

for i in range(len(disease)):
    frame1.insert(END,'  '+disease[i]+'\n')



btn=Button(master=scr,text='BACK',command=adminpage)
btn.place(x=670,y=420)


scr.mainloop()
```

COST ESTIMATION OF THE PROJECT ALONG WITH COST ESTIMATION MODEL

As healthcare costs continue to rise and the Affordable Care Act makes its presence felt in the marketplace, consumers are becoming more savvy as to how and where to spend their healthcare dollars. As a result, the industry is experiencing increased competition, and owners of healthcare facilities are offering an array of choices to discriminating consumers through newer and more technologicallyadvanced facilities, whether through a new building or an expansion or retrofit of an existing building.

Healthcare facility owners are becoming more budget conscious and are taking steps to ensure that funds earmarked for design and construction are spent in the most optimal manner possible. More facility owners are choosing to contract with an independent third party to help proactively establish realistic, accuratebudgets and better manage costs throughout the duration of the project. These professional cost managers work in tandem with the owner, design team, and contractor to establish cost baselines that fully take into account any risks and issues that may emerge during the project.

The ideal time to bring in an independent cost manager is before design commences since decisions made early in the process affect nearly every other aspect of the project. This is especially applicable with healthcare projects since they are very high-end, high-tech spaces that must accommodate multiple functions (imaging, surgery, patient care at various levels from standard to critical, etc.) under one roof but in a variety of spaces ranging from emergency rooms to imaging and diagnostic areas, procedure rooms, operating rooms, and recovery suites.

the predesign phase is the base from which all other phases can potentially succeed or falter in a healthcare project. In many cases, a disconnect can exist between owner and user expectations and the budget, causing frustration and disappointment with the end product once it becomes apparent that those expectations exceed what the budget will accommodate. Once the process reaches the design and construction phases, it becomes more difficult and costly to make changes to the project scope.

The independent cost review process during predesign is comprised of several steps:

- First, program and pre-conceptual estimates are ascertained, and a risk matrix is developed.
- Next, a cost model base estimate is established.
- A realistic overall budget is then generated, and a project Proforma is prepared.
- Finally, the estimate and owner’s budget are reconciled.

Cost estimation methods can be divided into two main categories: micro-costing methods, also known as activity-based costing (ABC), which are based on a detailed analysis of resource use and unit costs of each resource; and gross-costing methods, which are based on aggregate data.9 Most AHRQ Estimating Costs grantees used an ABC method. Each summarizes for eachmethod the purposes it serves, data required, possible analysis methods, and key considerations.

For our project we use ABC cost estimation model to calculate , here in below we describe about the module used:-

The ABC method can be used to either prospectively or retrospectively assess the clinic-level costs of primary care transformation efforts. ABC estimates are based on a detailed analysis of resources used and the unit costs of each resource.

The ABC method is the most common approach to assess the clinic-level costs of PCMH transformation efforts. It is most appropriate for estimating the costs of a single practice or small group of practices. While it is theoretically possible for the ABC method to be used over a large number of practices or an entire health system, this method is very labor intensive, and therefore is generally not practical for this purpose.

The basic ABC method follows these four steps:

- **Step 1:** Identify key cost elements and the degree to which they were or will be utilized. This can be achieved by interviewing a wide variety of key informants (i.e., practice leaders, clinicians, information technology staff, transformation leaders, and administrative staff) to collect information about the activities, staff, investments, and purchases associated with the transformation or by asking practice leaders to complete questionnaires. Questionnaires can be structured according to predefined categories of transformation expenses drawn from the published literature,[11-14] or based on investigators' prior experience with primary care transformation.
- **Step 2:** Assess unit costs for each cost element. In most cases, clinic leaders can provide this information by completing a spreadsheet or questionnaire and reporting on actual costs, such as staff salaries and benefits, equipment and space purchases, or leases. This can also be determined by using external data sources to calculate standardized staff costs based on labor categories.
- **Step 3:** Multiply unit costs by the quantities of each resource utilized (e.g., number of full-time equivalent (FTE) staff per job category) to produce total costs for each item.
- **Step 4:** Add total costs for each item to produce a total cost.

Data Collection:-

For researchers using a micro-costing or ABC method, data collection may present the greatest challenge for generating accurate cost estimates of transforming care. Barriers to data collection using the ABC method include the following:

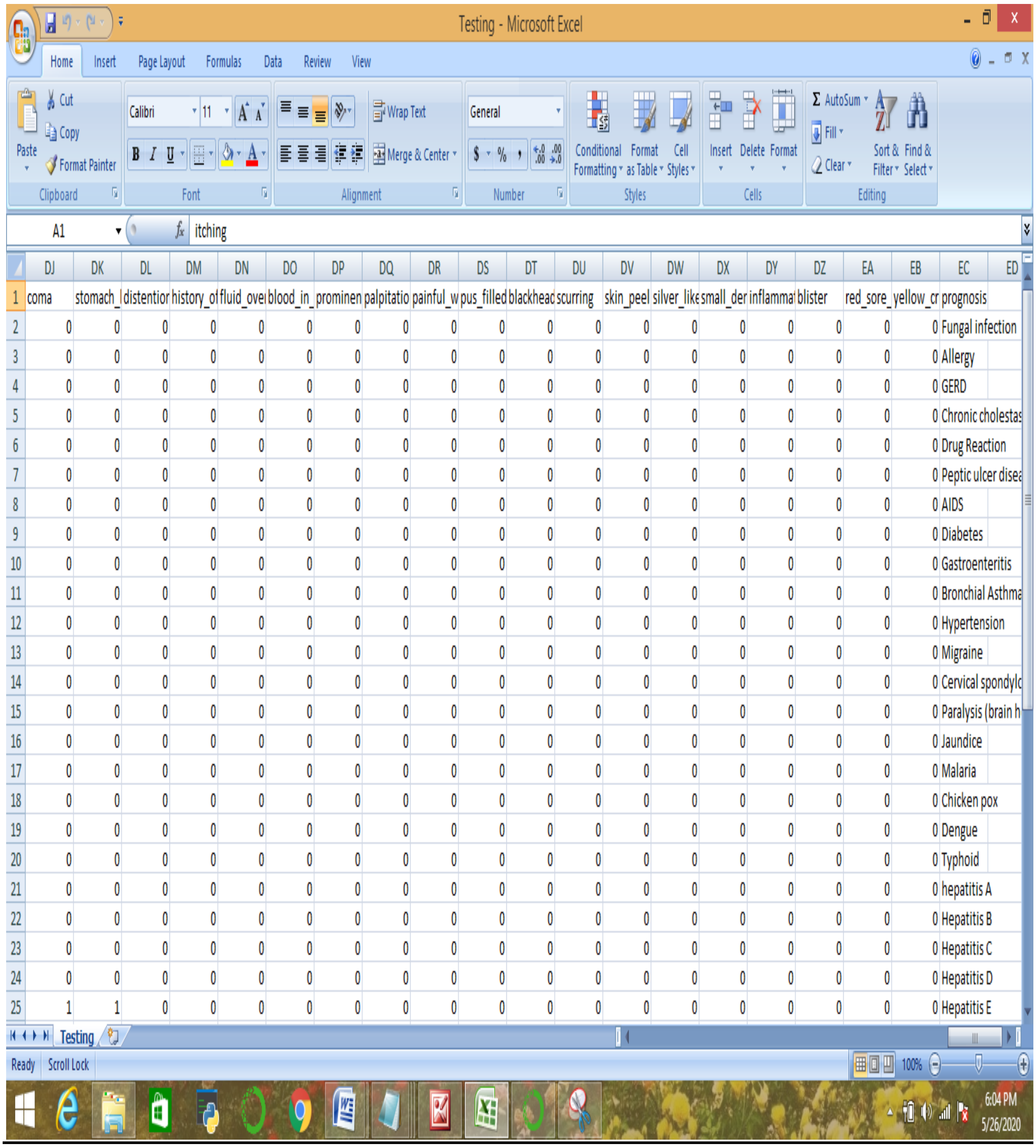
- Data collection can often be time consuming and burdensome for both researchers and respondents.
- Respondents may have limited expertise in cost data collection and a lack of familiarity with practice costs.
- Retrospective data collection is subject to recall and nonresponse biases due to staff turnover. Staff may not recollectall activities implemented as part of the transformation effort, and staff who have left the practice cannot report on the time they spent on transformation-related activities.
- Estimating the costs of maintaining practice changes may be difficult because of challenges of attribution. Once the transformation has been implemented and the changes have become part of the regular workflow, clinicians and staff may not be able to easily distinguish how much time is being spent on general practice activities versus transformation-related activities.
- These challenges can be mitigated by offering flexible options for data collection.

Key considerations for the ABC method include:

- ➢ This method can be used to fairly precisely estimate the costs of practice changes at the clinic level.
- ➢ This method is computationally simple.
- ➢ Data collection using this method can be very time consuming.
- ➢ Estimating Costs grantees found that it was most effective to include on the research team one or more junior staff members who focused on data collection; a clinician who practices in the clinics studied and has a good rapport with the staff who will be interviewed; and a multidisciplinary research team, including finance and systems operations specialists, to facilitate cost attribution and analysis.
- ➢ This method can be used to estimate costs either prospectively or retrospectively.
- ➢ In retrospective analyses, recall bias and staff turnover are barriers to data collection.
- ➢ Researchers must find or develop a tool for accurately collecting cost data. Examples of cost data collection tools developed by Estimating Costs grantees are included in Appendix B. These tools provide a detailed breakdown of cost elements and may be helpful to estimate the costs of primary care transformation in other settings.

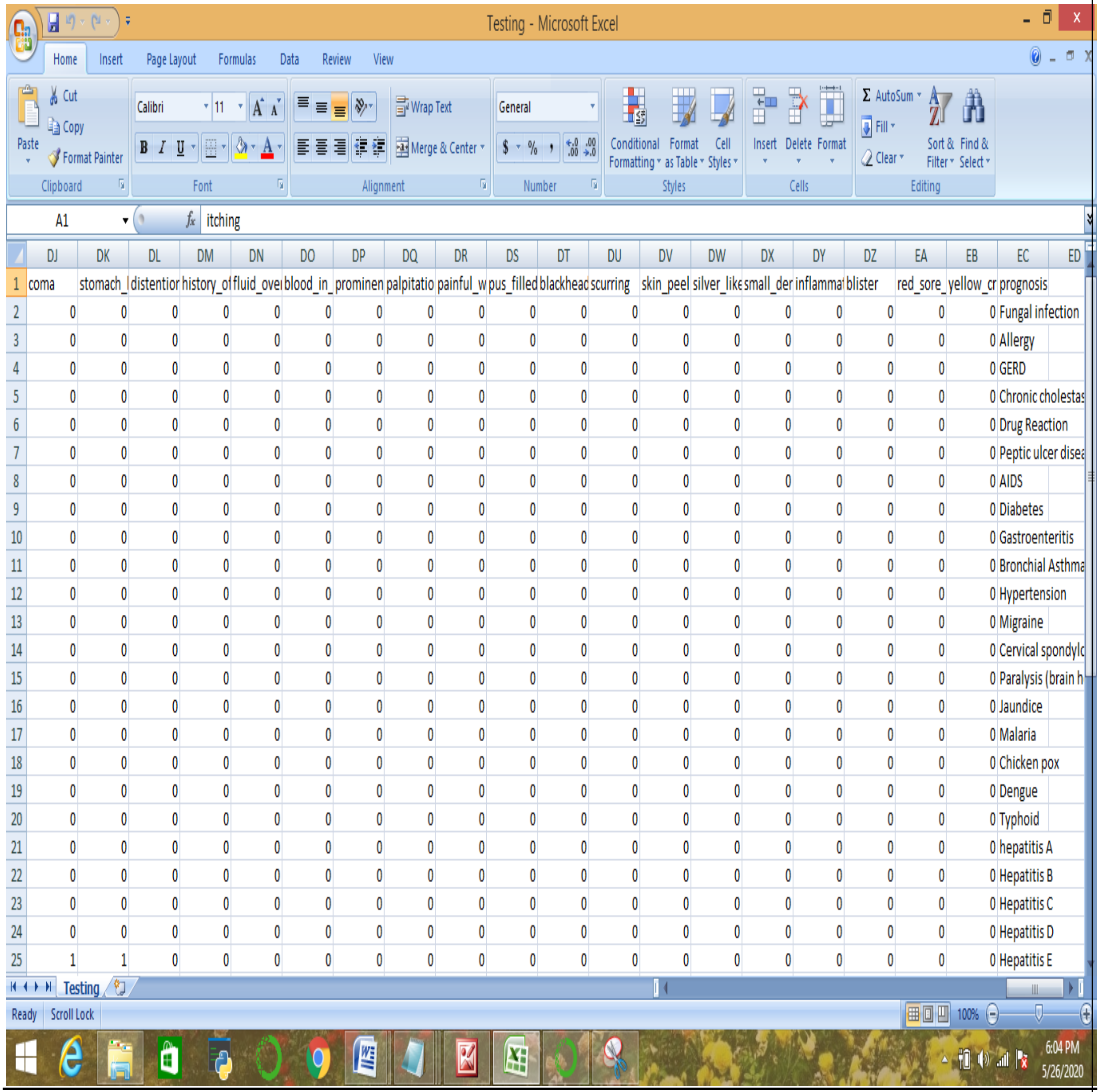
SCREENSHOT

Data set:-



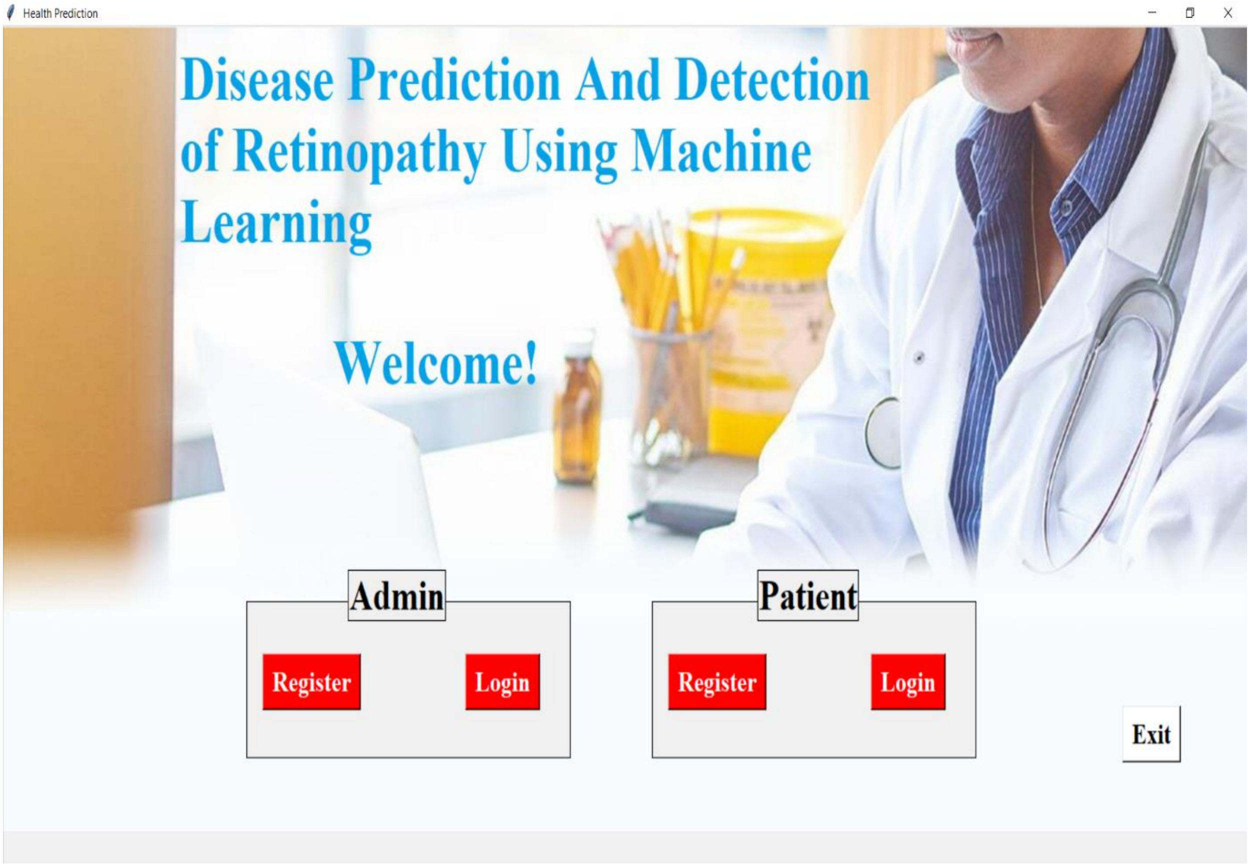
A1 itching

	DJ	DK	DL	DM	DN	DO	DP	DQ	DR	DS	DT	DU	DV	DW	DX	DY	DZ	EA	EB	EC
1	coma	stomach_l	distentior	history_of	fluid_ove	blood_in_	prominen	palpitatio	painful_w	pus_filled	blackhead	scurring	skin_peel	silver_like	small_der	inflammat	blister	red_sore_	yellow_cr	prognosis
2	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	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	GERD
5	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	Chronic cholestas
6	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	0	0	0	0	0	0	0	0	0	0	0	Peptic ulcer disea
8	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	0	0	0	0	0	0	0	0	0	0	0	0	Diabetes
10	0	0	0	0	0	0	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	Bronchial Asthma
12	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	Migraine
14	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	Cervical spondylo
15	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	Paralysis (brain h
16	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	Jaundice
17	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	Malaria
18	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	Chicken pox
19	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	Dengue
20	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	Typhoid
21	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	hepatitis A
22	0	0	0	0	0	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	Hepatitis C
24	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	Hepatitis D
25	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	Hepatitis E



	DJ	DK	DL	DM	DN	DO	DP	DQ	DR	DS	DT	DU	DV	DW	DX	DY	DZ	EA	EB	EC	ED
1	coma	stomach_	distentior	history_of	fluid_ove	blood_in_	prominen	palpitatio	painful_w	pus_filled	blackhead	scurring	skin_peel	silver_like	small_der	inflamma	blister	red_sore_	yellow_cr	prognosis	
2	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	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	GERD	
5	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	Chronic cholestas	
6	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	0	0	0	0	0	0	0	0	0	0	0	Peptic ulcer disea	
8	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	0	0	0	0	0	0	0	0	0	0	0	0	Diabetes	
10	0	0	0	0	0	0	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	Bronchial Asthma	
12	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	Migraine	
14	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	Cervical spondylo	
15	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	Paralysis (brain h	
16	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	Jaundice	
17	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	Malaria	
18	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	Chicken pox	
19	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	Dengue	
20	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	Typhoid	
21	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	hepatitis A	
22	0	0	0	0	0	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	Hepatitis C	
24	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	Hepatitis D	
25	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	Hepatitis E	

Homepage:-

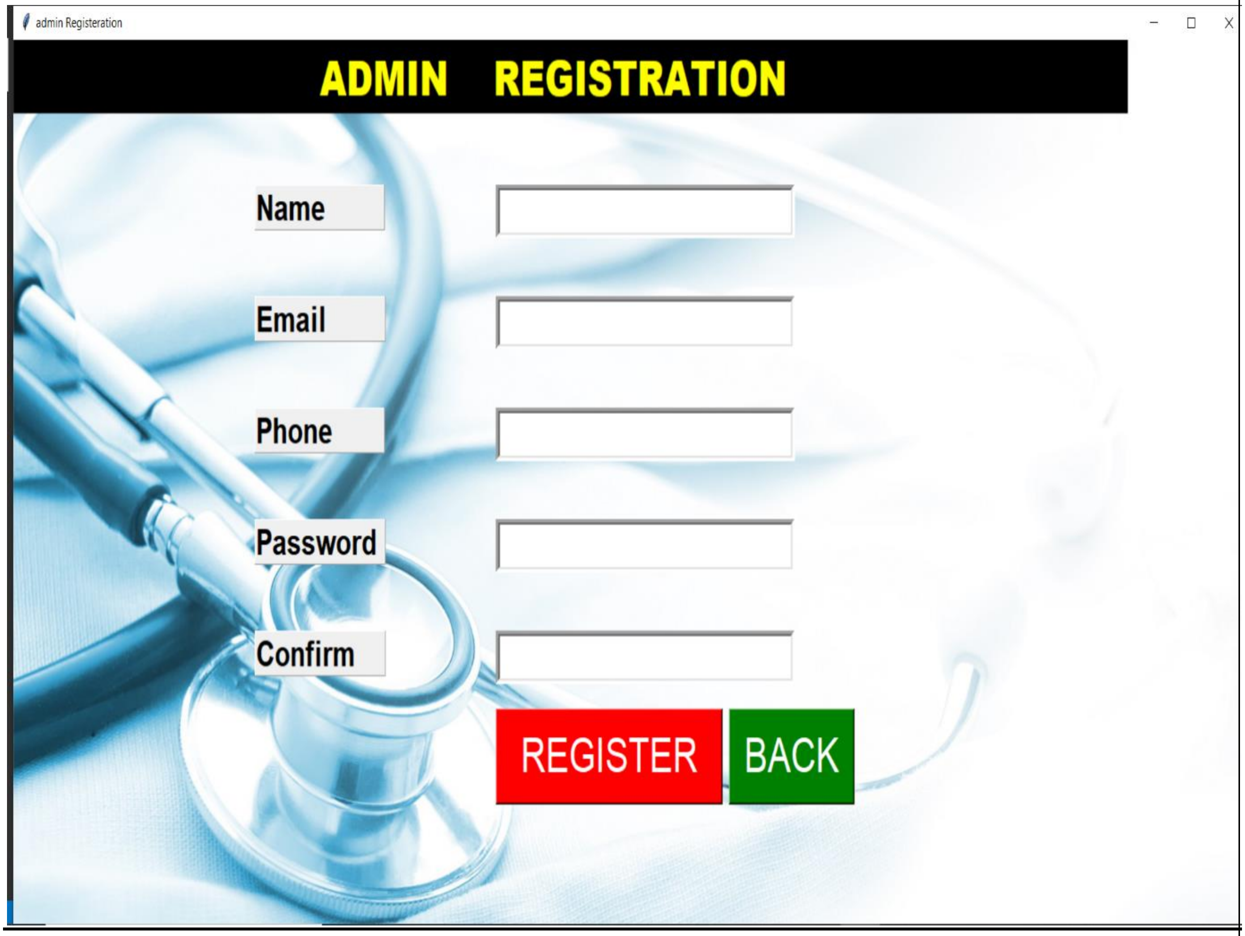


Health Prediction

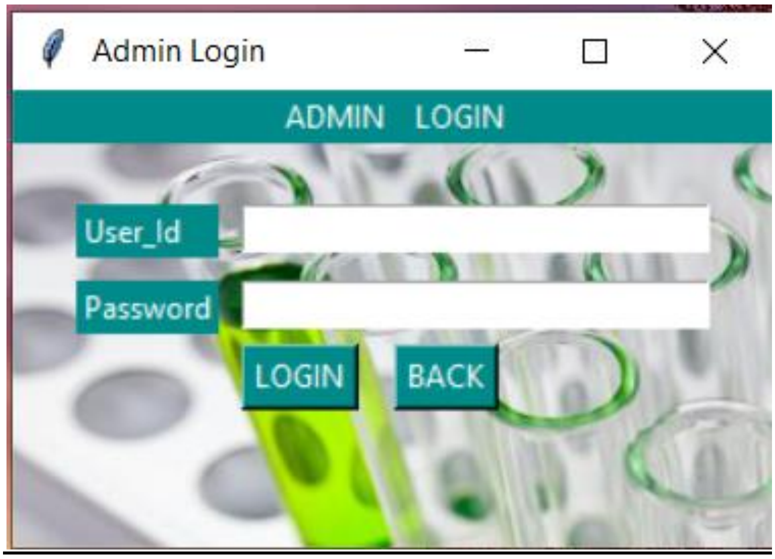
Disease Prediction And Detection of Retinopathy Using Machine Learning

Welcome!

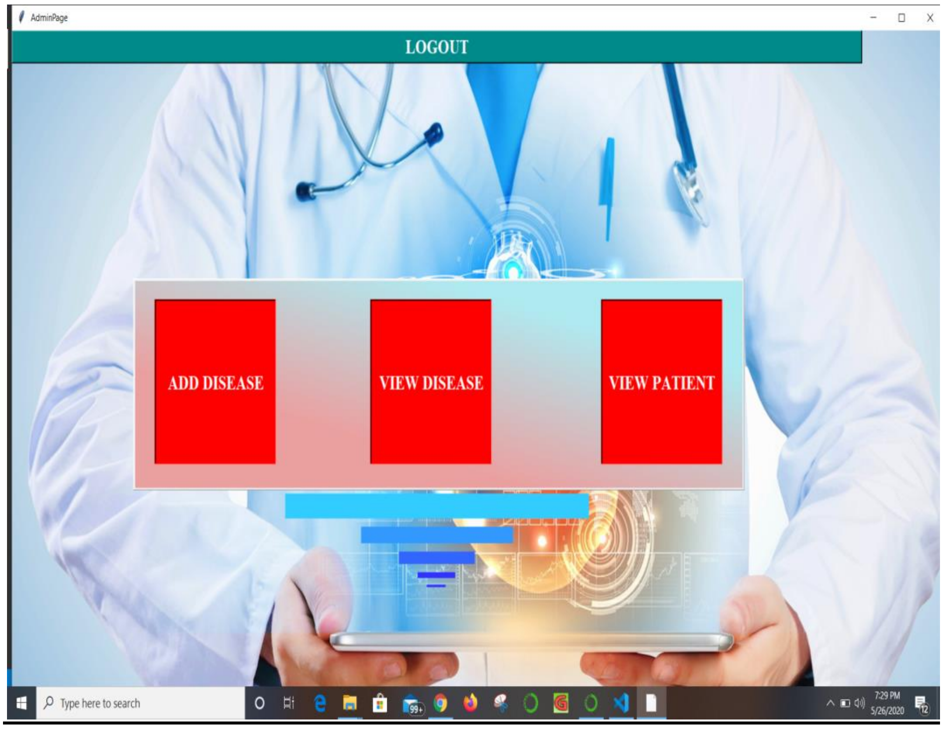
Admin Registration Page:-



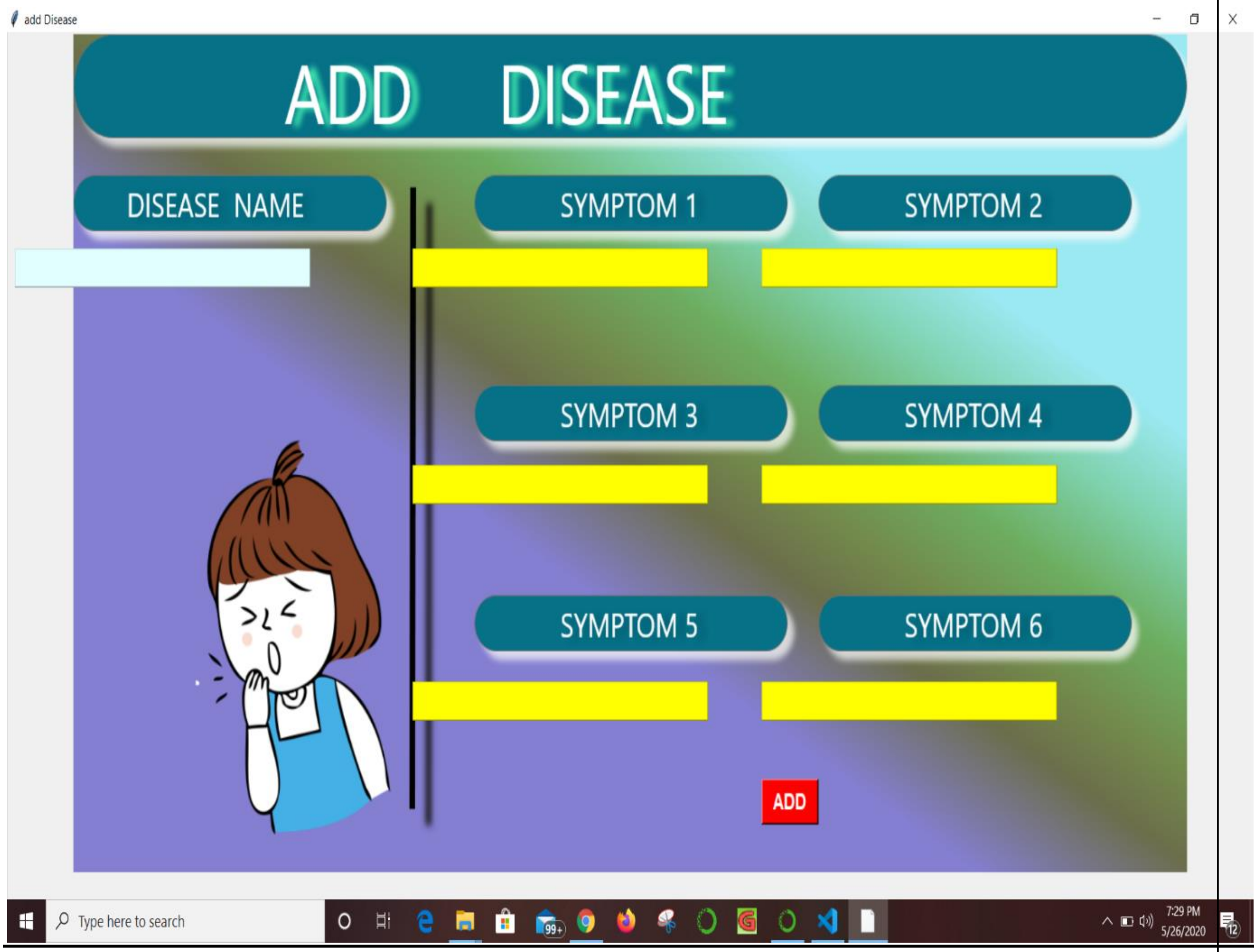
Admin Login page:-



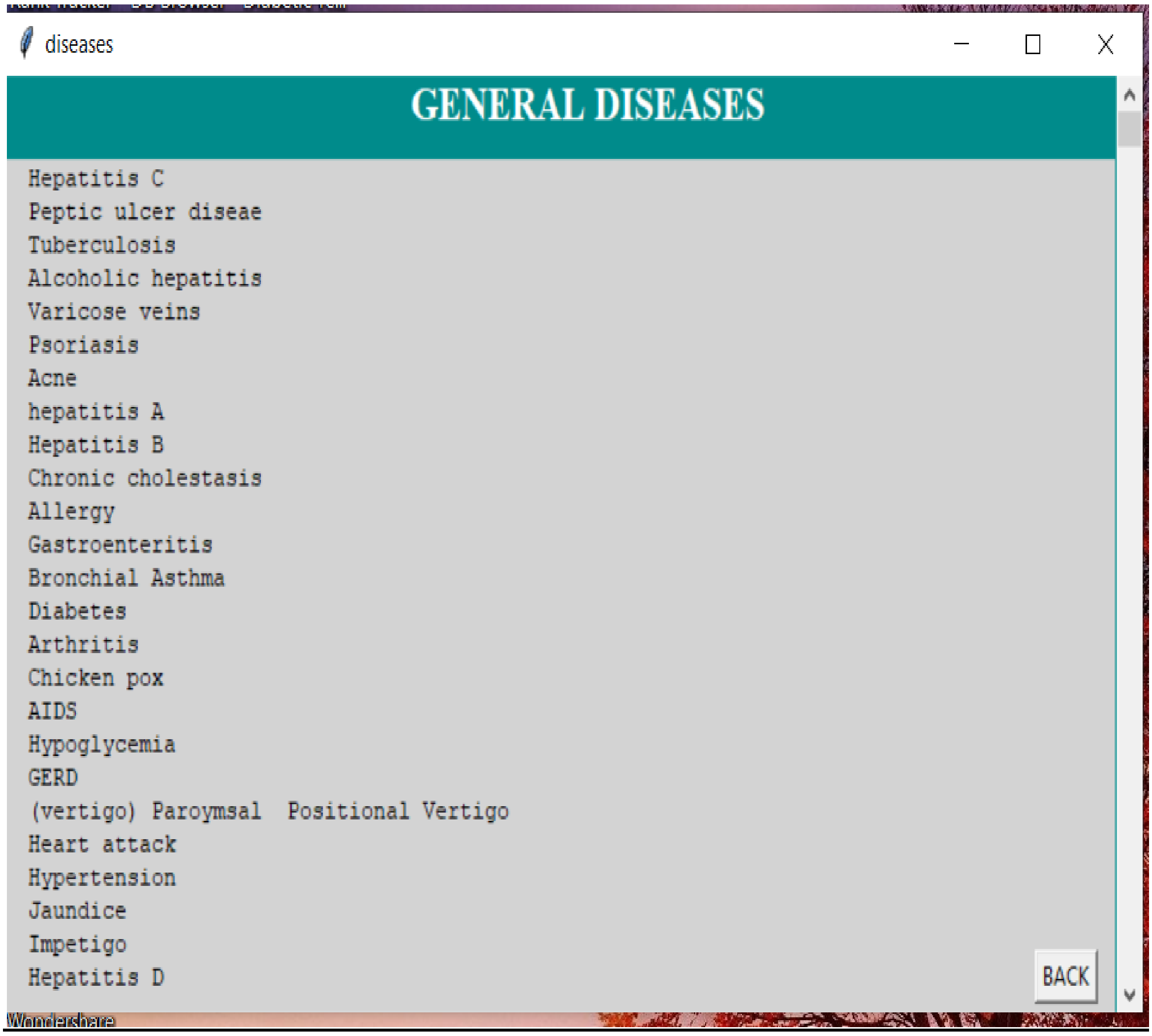
Admin Homepage:-



Add Disease:-


add Disease
ADD DISEASE
DISEASE NAME
SYMPTOM 1
SYMPTOM 2
SYMPTOM 3
SYMPTOM 4
SYMPTOM 5
SYMPTOM 6
ADD
Type here to search
7:29 PM
5/26/2020

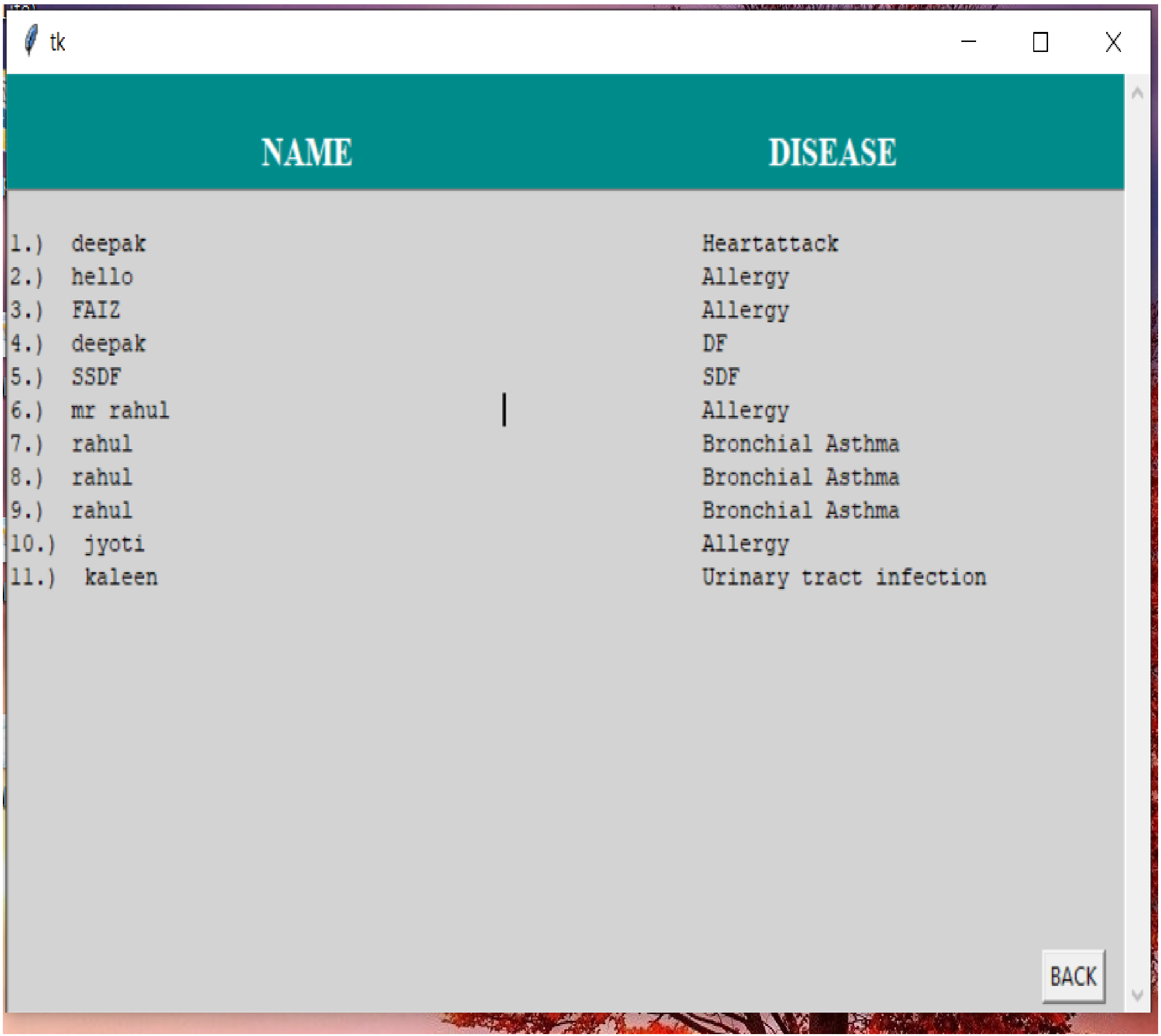
View Disease:-



GENERAL DISEASES

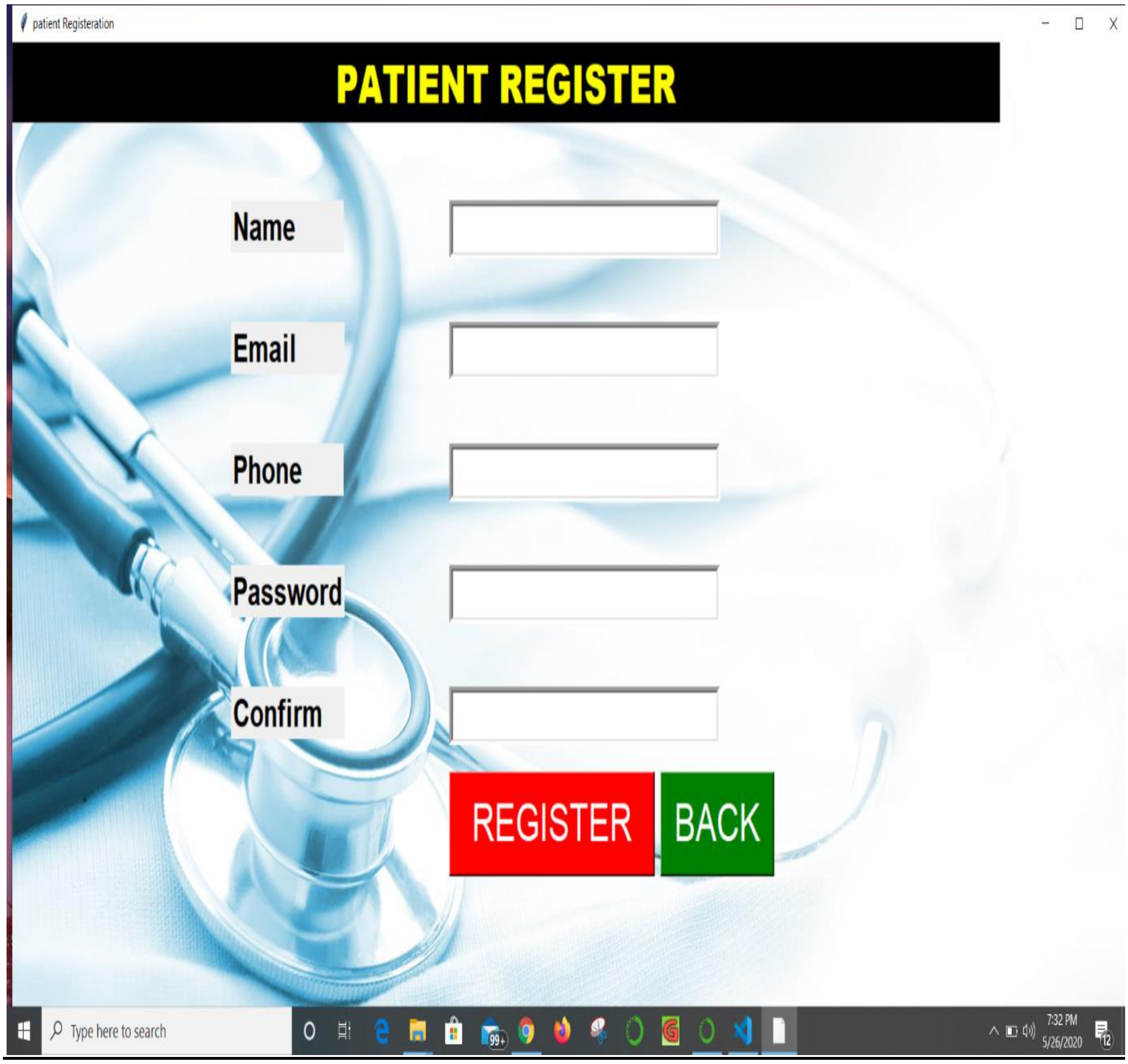
```
Hepatitis C
Peptic ulcer diseae
Tuberculosis
Alcoholic hepatitis
Varicose veins
Psoriasis
Acne
hepatitis A
Hepatitis B
Chronic cholestasis
Allergy
Gastroenteritis
Bronchial Asthma
Diabetes
Arthritis
Chicken pox
AIDS
Hypoglycemia
GERD
(vertigo) Paroymsal  Positional Vertigo
Heart attack
Hypertension
Jaundice
Impetigo
Hepatitis D
```

View Patient:-

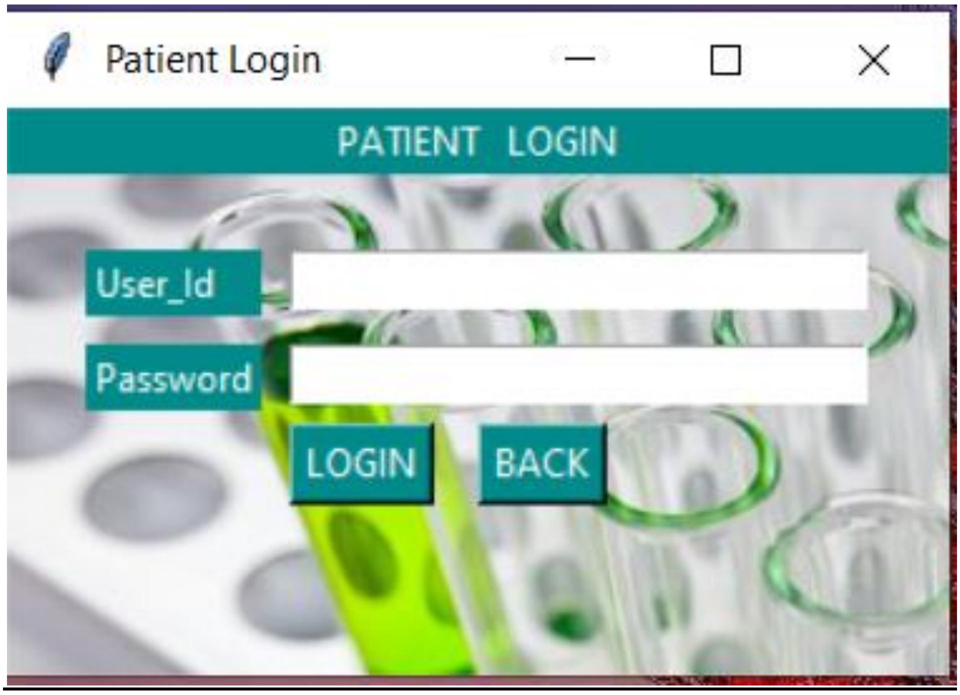


NAME	DISEASE
1.) deepak	Heartattack
2.) hello	Allergy
3.) FAIZ	Allergy
4.) deepak	DF
5.) SSDF	SDF
6.) mr rahul	Allergy
7.) rahul	Bronchial Asthma
8.) rahul	Bronchial Asthma
9.) rahul	Bronchial Asthma
10.) jyoti	Allergy
11.) kaleen	Urinary tract infection

Patient Registration Page:-



Patient Login Page:-



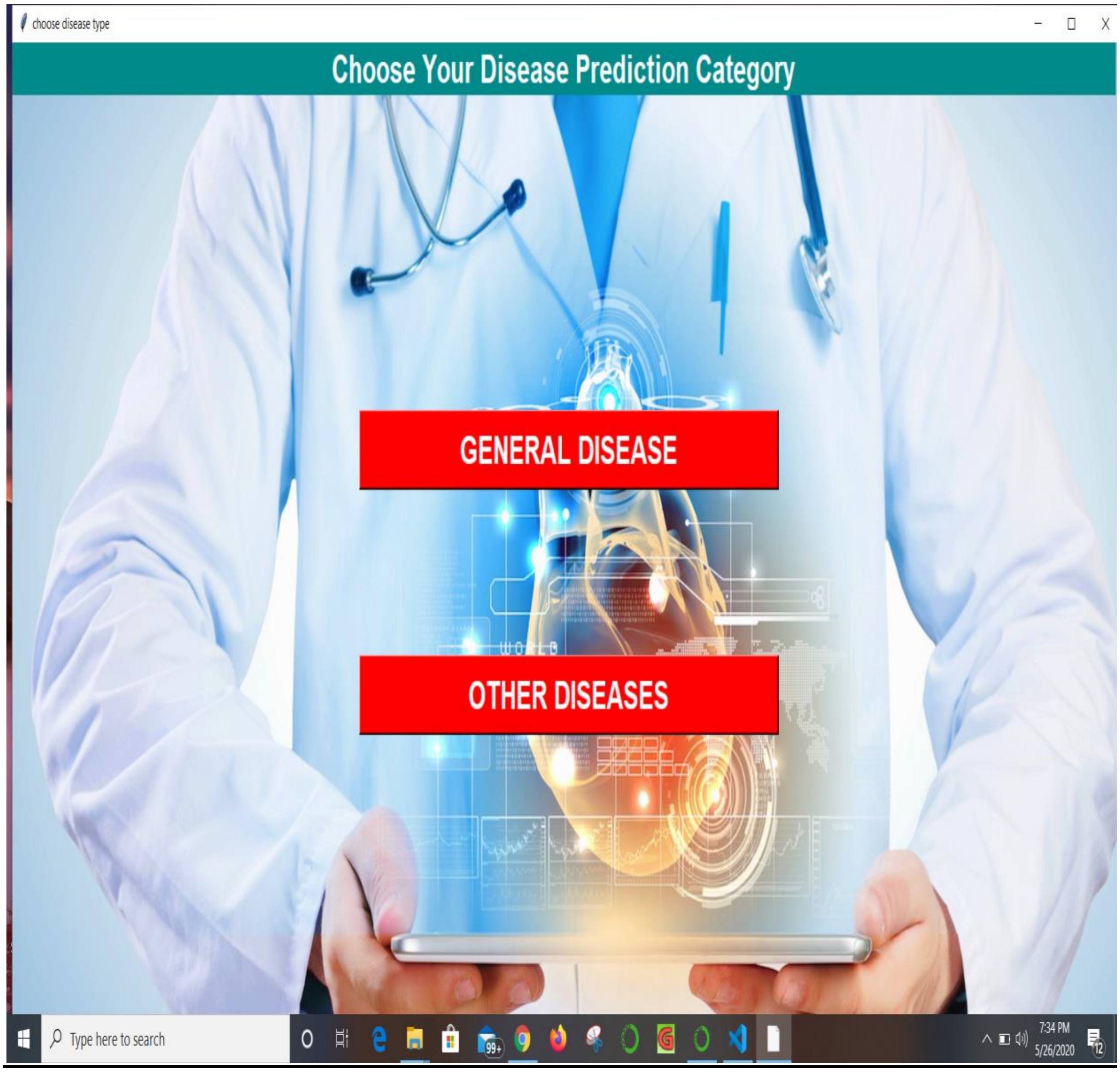
Patient Login

PATIENT LOGIN

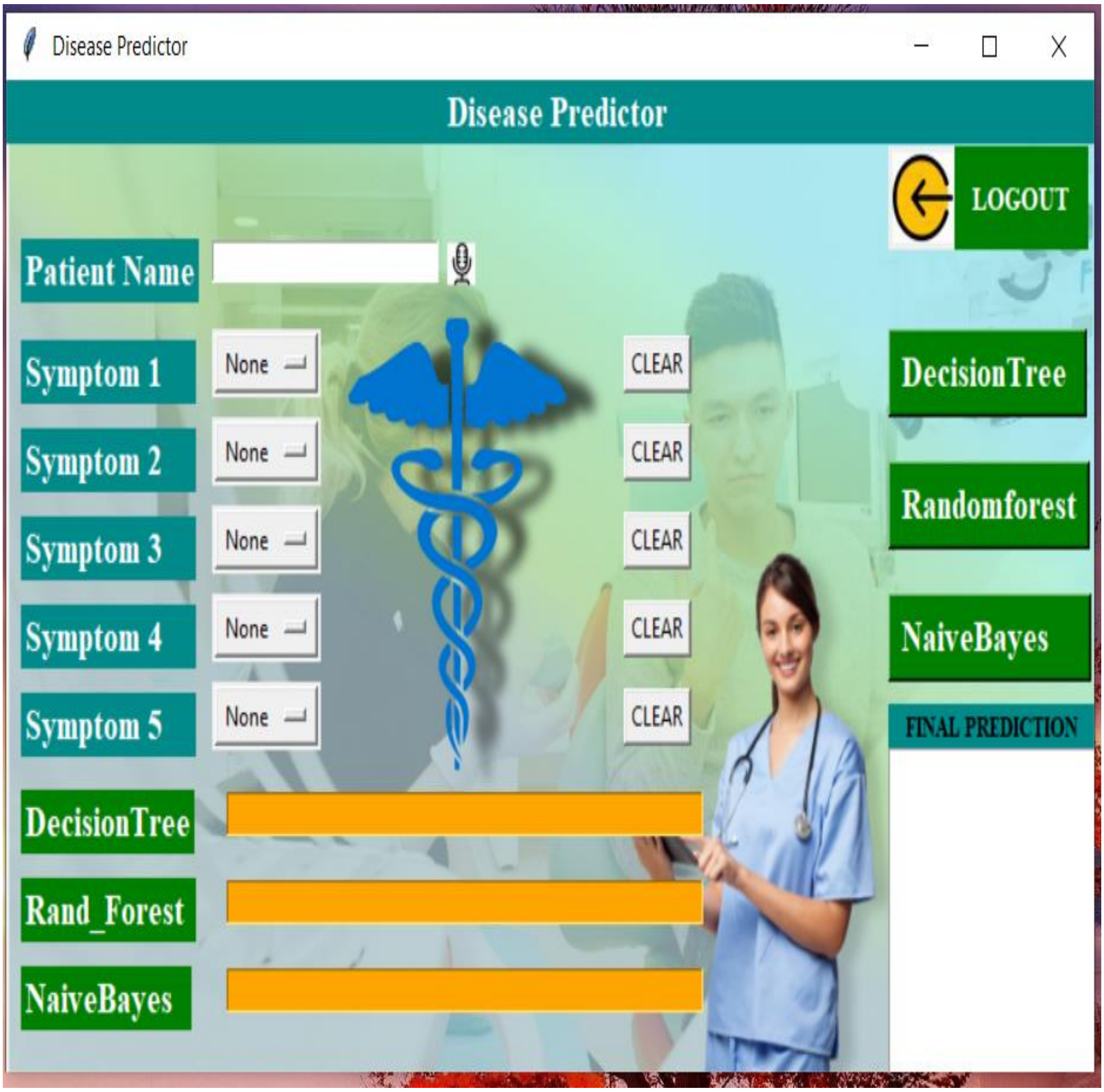
User_Id

Password

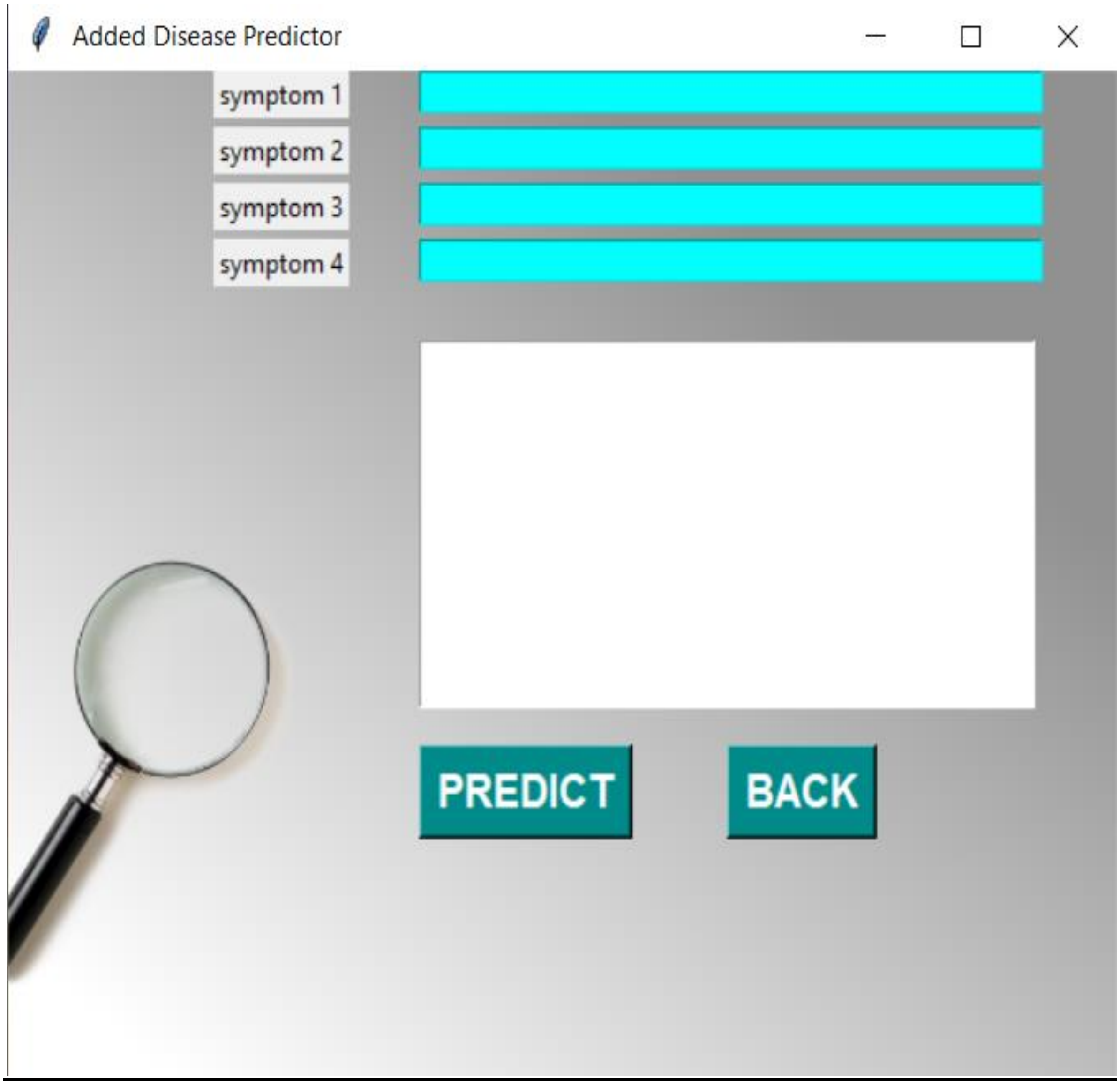
Patient Homepage



General Disease Prediction:-



Other Disease Prediction:-



FUTURE SCOPE

The future of healthcare technology is entering a new era as researchers, innovators, and lawmakers strive to improve the accessibility, effectiveness, and cost of care.Technology undoubtedly will play a key role in healthcare's future In medicine and healthcare, **digital technology** couldhelp transform unsustainable healthcare systems into sustainable ones, equalize the relationship between medical professionals and patients, provide cheaper, faster and more effective solutions for diseases**.** The future of healthcare includes technology that seamlessly combines data on a patient's medical history, real-time health, insurance coverage, and financial information all to support provider decision-making, improve patient health, and reduce costs.

One technology evolution that's already helping is the rapid exchange of patient data, which allows doctors to better understand the context of a patient's overall health," the CISO said. "Providers must continue to move away from manual data sharing, such as faxing records to one another, and toward automation. Improving integration of electronic health record (EHR) data means labs, care plans, and medical histories from various sources are available in minutes or seconds — not days or weeks — so theprovider can make a clear diagnosis and develop the most effective care plan in less time."

From digital networks to wearable, the **health care** industry is undergoing massive technological changes. Here are 10 **types** of innovations changing its **future**. The **health care** industry will see a 21% increase in IT jobs by 2020, according to research by the University of Chicago.

Disease Prediction is done by implementing the Naïve Bayes Classifier. Naïve Bayes Classifier calculates the probability of the disease. Therefore, average prediction accuracy probability 60% is obtained.

BIBLIOGRAPHY

The health care industries collect huge amounts of data that contain some hidden information, which is useful for making effective decisions. For providing appropriate results and making effective decisions on data. It enables significant knowledge, e.g., relationships between medical factors related to heart disease and patterns, to be established. We have employed the multiple train and test data with backpropagation as the training algorithm. The obtained results have illustrated that the designed diagnostic system can effectively predict the risk level of diseases.

In this study, anHEALTH DETECION AND PREDICTION IN RETINOPATHY has been presented using MACHINE LEARNING. From KNN, andNAÏVE BAYES together with RANDOM FOREST algorithm is used to develop the system. The model proves the better results and assists the domain experts and even the person related to the medical field to plan for a better and early diagnosis for the patient. This system performs realistically well even without retraining. Furthermore, the experimental results show that the system predicts disease most possible accuracy by using machine learning algorithm.