SOFTWARE REQUIREMENTS SPECIFICATION

for

<Disease Prediction Application>

Version 1.0 approved.

Prepared by

19Z202 -Anita Priyadarshini A

19z211-Divya Darshini R

19z217- Hemavarshini B

19Z227 - Monirhithikka S.P

19Z240 - Samyuktha Sreekanth

19Z243 - Sarayu Miththira V.C

BATCH - 4

1. Introduction

1.1 Purpose

The purpose of this document is to describe the software specification requirements of a Chronic disease prediction web application. The document will describe how the system can predict a user's closest matching disease based on the symptoms entered and gives details regarding the treatment.
Nowadays health industry plays a major role in curing the diseases of the patients , it is useful for the user in case he/she doesn't want to go to the hospital or any other clinics so just by entering the symptoms and all other useful informations the user can get to know the disease he/she is suffering from and the health industry can also get benefit from this system by just asking the symptoms from the user and entering in the system and in due time the user is provided with a closest disease prediction .
This is just a preliminary/precautionary app to help during these hard times when going to the hospital and getting an appointment is not so convenient. The document contains the functional behavior and non-functional requirements of the system project. The document also contains the guidelines for system engineers and programmers to start working and accomplish the project on a given time frame. The product will be beneficial to society.

1.2 Document Conventions

The format of this Software Requirement Specifications for disease prediction web application is simple. In general, this document follows the IEEE formatting requirements. Bold face and indentation are used on general topics and on specific points of interest. The rest of the document will be written using the standard font, Arial italics font size 11, or 12 throughout the document for text. Document text should be single spaced and maintain the 1" margins found in this template. For Section and Subsection titles please follow the template.

1.3 Intended Audience and Reading Suggestions`

The intended readers of this document are the developers and the users (i.e common people). This application is mainly designed to help people and provide medical assistance during the pandemic when going to the hospital and getting an appointment is not so convenient.

1.4 Product Scope

The software product is named Chronic disease prediction web application. It will be able to collect and analyse a user's closest matching disease based on the inputs provided. The main goal is to help people and provide medical assistance during the pandemic when going to the hospital and getting an appointment is not so convenient.
The application's objective is to provide an easy and straight-forward way by which users can obtain disease predictions and treatment suggestions. New users will register accounts with the application. Moreover, the application is free of charge so that more users could benefit.

1.5 References

1.5.1 IEEE 830-1998 standard for writing SRS documents.

1.5.2 IEEE Std 1233, 1998 Edition, IEEE Guide for Developing System Requirements Specifications

2. Overall Description

2.1 Product Perspective

The product can be described as self contained and stand-alone. It isn't dependent on any other system. Although it acquires its database from an external source that is the users and medical journals, it still works independently. The database of symptoms is included during its creation and prior to that it isn't dependent on any external component. It may require updates in the database if required but that wouldn't affect the current working of the application.

2.2 Product Functions

User:

1. Register
2. Login
3. Log Out
4. Select Chronic Disease
5. Enter Parameters
6. Viewing Suggested Treatments

Web Server:

1. Updating Database

DFD attached below.

2.3 User Classes and Characteristics

We have two user classes interacting with each other in our application.

1. User
    - This is the person that registers and uses the application. They create their profile and enter their symptoms. They can then view their predicted disease and suggested treatment.
2. Web server

- This is the application itself. It updates the database it receives and analyzes the symptoms entered to make a prediction with suggested treatment.

2.4 Operating Environment

Requirements would include a keyboard ,monitor , an OS although Windows 10 operating system will be an ideal working environment for the system. Additionally we will also require an API(Application Programming Interface) and GUI Design Software

2.5 User Documentation

We will provide a user manual along with the software which will include the description of the working of the application . We may also additionally provide a video tutorial, depending on the the time constraints applicable.

2.6 Assumptions and Dependencies

- The Application will only serve as a preliminary measure to visiting a doctor or hospital
- The accuracy of the disease predicted is parallel to the accuracy of the symptoms entered by the user
- The application does not substitute visiting the hospital or doctor
- The application attempts to make the “most probable choice”. This is important because the user must not use this as a final diagnosis and must consult their doctor for figuring out their final course of treatment
- The application provides an immediate/emergency course of treatment, which may only provide temporary relief.
- The application does not prescribe pharmacy drugs. It provides safe home based, or extremely controlled dosage of commonly prescribed medicines, which don't require a prescription.
- The application is created with a database of common or more easily curable diseases, and some more common diseases of mild intensity. It is not updated to make predictions for serious diseases as that would cause unnecessary panic as most of them would require proper diagnostic tests to confirm and should not be confirmed though this or any similar disease prediction application. So the user should keep these things in mind to use the application effectively and as the developers intended it to be used as.

3. EXTERNAL INTERFACE REQUIREMENTS

3.1 User Interface

The user interfaces are divided into two major components as follows :

- One part includes the user accessing the system using a cell phone. ● The other portion involves accessing the system through a remote site or at a particular location specifically designed to access the system.

The aspects of optimizing the interface with the person who must use the system are briefly described below:

- Allow new users to become members of the system
- Allow current users to login into the system using a unique user id and password.
- Allow the users to build new itineraries, change and/or view existing itineraries, and pay for planned itineraries using different methods (credit cards, CRM credit account).
- Provide an on-line help for all users of the system.
- Provides five different services- Prognosis, Search Hospital (nearest), Appointment, Blood Bank, and Emergency (contacts).
- Allows analysis with the help of efficient multiple machine learning algorithms helps to predict the disease more correctly and help treat patients.

3.2 Hardware Interface

For the hardware requirements the SRS specifies the logical characteristics of each interface between the software product and the hardware components. It specifies the hardware requirements like memory restrictions, cache size, the processor, RAM size etc. that are required for the software to run.

3.2.1 Minimum Hardware Requirements

1. Operating System: Windows
2. Hard disk: 40 GB

3. RAM: 256 MB
4. Processor: Pentium(R)Dual-core CPU

3.3 Software Interface

Any window based operating system with DOS support are primary requirement for software development. The systems must be connected via LAN and connection to the internet is mandatory.

The software requirements are:

1. Windows 2000 OS
2. Visual Studio
3. Java Language

3.4 Communication Interface

This project can be compatible with all platforms. Connections to the system will be over TCP/IP connection, project supports all types of web browsers. I have used database so my system can work offline.

4. System Features

4.1 Description and Priority

The disease prediction web application is limited by specific technologies, tools, and databases to be used and money or funding for system development. The speed and availability of a user's internet connection will affect how quickly the application responds. The following are the essential features sorted according to the priority requirement of the proposed system:

· The users of disease prediction web applications must be equipped with a device. (High Priority)

· A stable internet connection is recommended for a smooth user experience. (Medium

Priority) Benefits(high - 9):

1. The application is free of cost and is made available to all users .
2. Less time consuming and provides added options for consulting .

Risk(medium-6):

The Application predicts the closest disease based on the user inputs although it cannot guarantee an exact diagnosis of the disease.

4.2 Stimulus/Response Sequences

A new user is prompted to register with the application, On a successful login the user will get access to the services provided by the application. The user can update his profile for better performance and accuracy. User can enter his/her symptoms which is analysed by the application and the closest disease is predicted and displayed on the screen. The application also provides suggestions on further treatments available for the same. After successful completion of the tasks, the user can log out of the application.

4.3 Functional Requirements

User:

4.3.1 Register

Users that do not have an account will be directed to make an account after entering prompted details.

4.3.2 Login

The application will prompt registered users to log in to get access to their accounts.

4.3.3 Log Out

After successfully completing their task, users can log out of their account.

4.3.4 Chronic Disease prediction

Users can select the Chronic disease and enter the required parameters. The Ml Algorithm predicts the likelihood of the user being diagnosed with the particular chronic disease

4.3.5 Viewing Risk likelihood

The user is displayed the result message

Web Server :

4.3.6 Updating Database

The database will be periodically updated by the administrator for better user experience.

5 NON FUNCTIONAL REQUIREMENTS

5.1 PERFORMANCE REQUIREMENTS

As an overview, the system must be interactive and the delays involved must be less .So in every action-response of the system, there are no immediate delays. In the case of loading the pages, there is delay much below 4 seconds and in the case of matching the entered details during the login with the available database of the registered users, the delay is much below 2 seconds. In case of opening databases, searching for the disease predicted and evaluation of the disease that is to be displayed, there are no delays and the operation is performed with negligible delay for opening, searching, optimizing the result, and displaying the desired results for the frequently accessed tuples of the database using on par DB manipulation techniques, ie for > 80% of the tuples. Also when connecting to the server the delay is based on the distance of the 2 systems and the configuration between them, so there is high probability that there might be an incompatible connection, so we expect less than 20 seconds for sake of good communication.

5.2 SAFETY REQUIREMENTS

Since our system deals with sensitive data of our users, utmost effort to be taken to protect the database of our users and their results, symptoms entered and other demographics. Encryption to be used to transmit the data relayed between the users and the admin between different systems/ servers for data safety as it is a web based application.

5.3 SECURITY REQUIREMENTS

The ‘Users’ details’ database is to be maintained until the expiration of the system. The transactions on the database needs to be documented and kept in secure environments for our business’ and state’s use and the logs shall be verbose enough to support forensics. The user registration shall be verified each time the user logs in. The user registration is made incorporating best security practices. The final evaluation of the result is expected to be maintained with all the peripheral data in a comprehensive manner long after the expiration of the system.

5.4 SOFTWARE QUALITY ATTRIBUTES

It is developed keeping in mind the possible functionalities that can be appended to the

software, hence providing *extensibility*.

5.5 BUSINESS RULES

This software isn't aimed at monetization or for any kind of business collaborations with third party institutions.

6 OTHER REQUIREMENTS

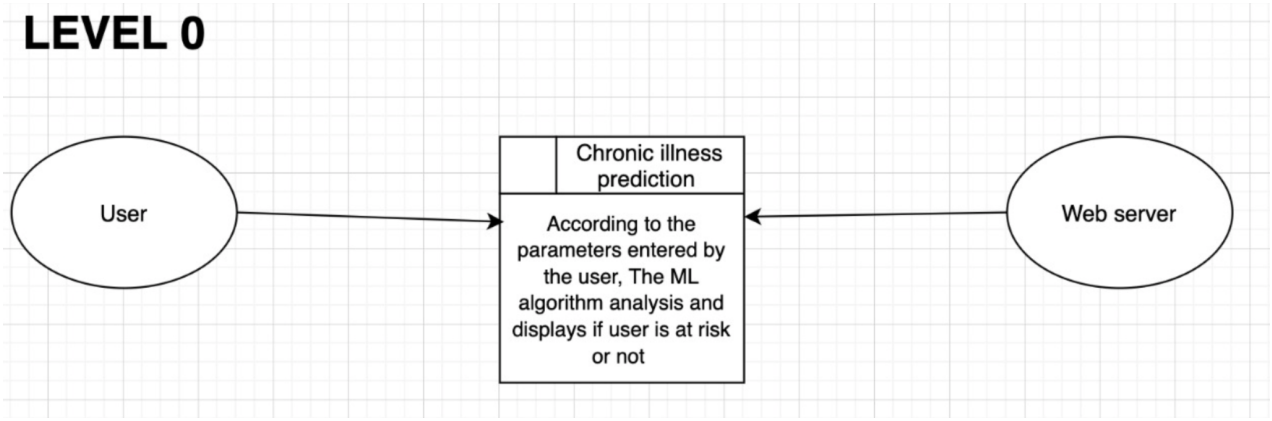
6.1 Appendix A: Glossary

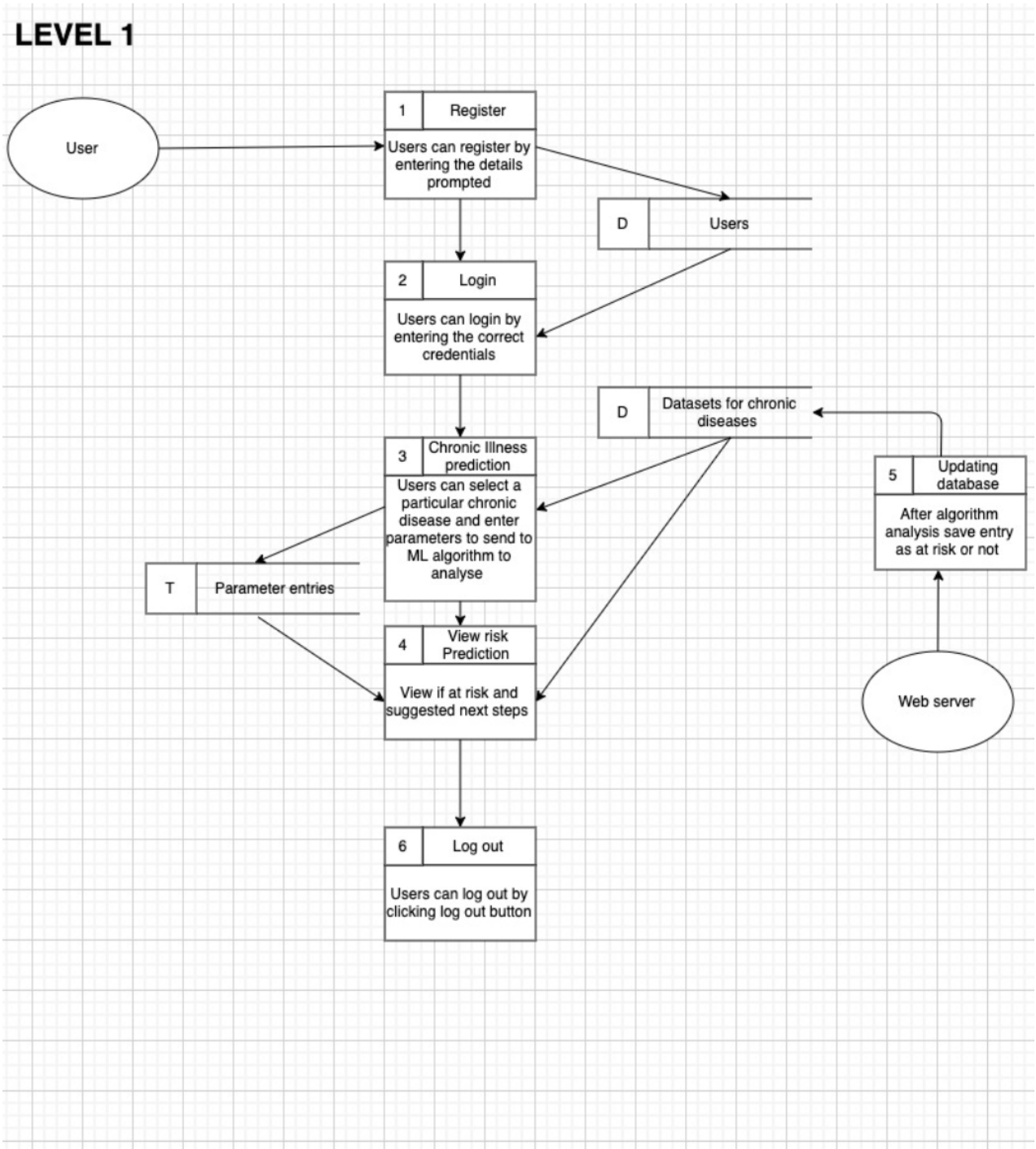
This part involves acronyms and abbreviations used in the SRS.

- SRS: Software Requirements Specification
- DFD: Data Flow diagram
- IEEE: Institute of Electrical and Electronics Engineers.
- Symptoms: indicating a condition of disease, particularly such a feature that is apparent to the patient.

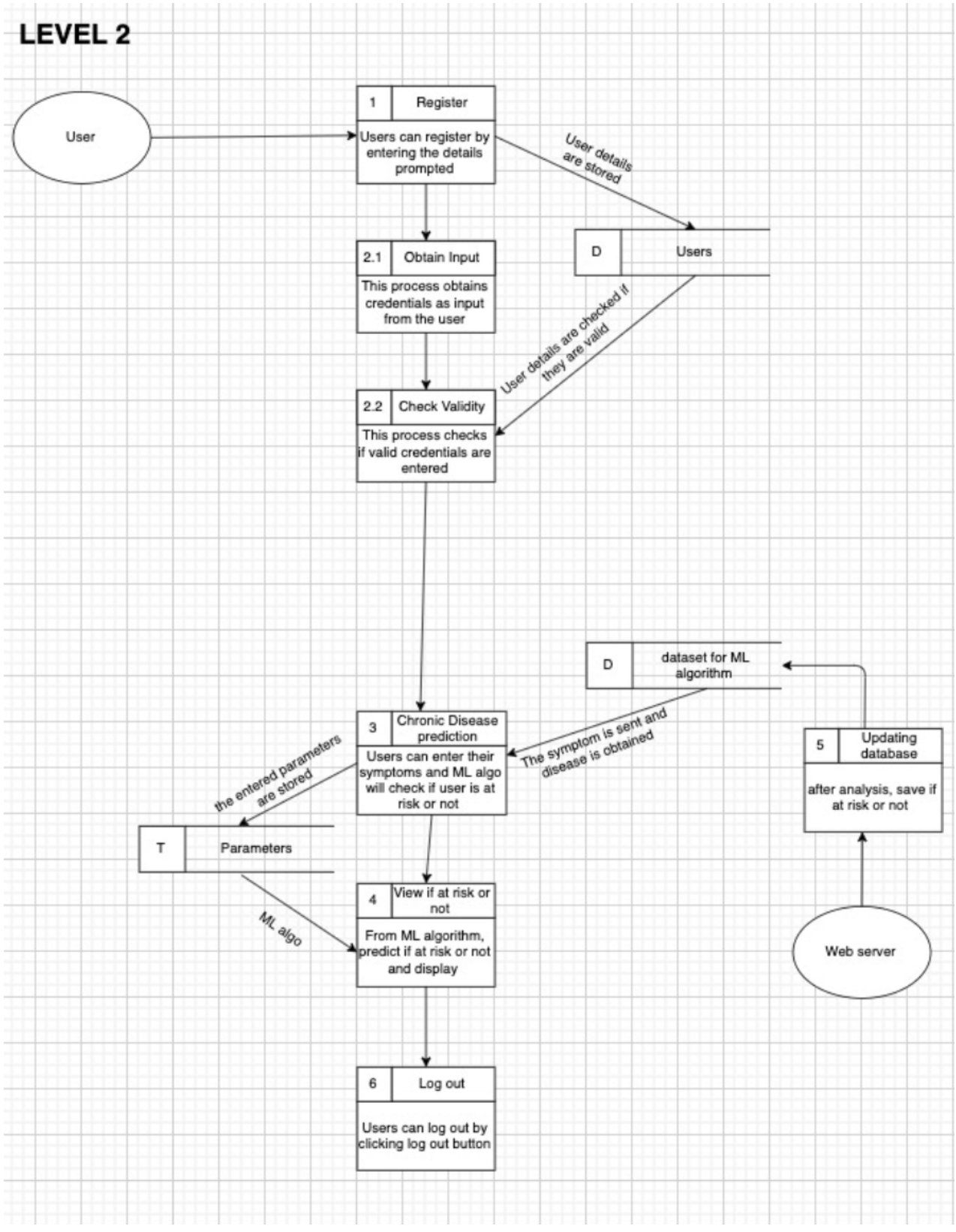
6.2 Appendix B: Analysis Models

The following is a DFD diagram of the Disease Prediction Application:




LEVEL 1
User
1
Register
Users can register by entering the details prompted
D
Users
2
Login
Users can login by entering the correct credentials
D
Datasets for chronic diseases
3
Chronic Illness prediction
Users can select a particular chronic disease and enter parameters to send to ML algorithm to analyse
T
Parameter entries
4
View risk Prediction
View if at risk and suggested next steps
5
Updating database
After algorithm analysis save entry as at risk or not
Web server
6
Log out
Users can log out by clicking log out button

LEVEL 2


User
1
Register
Users can register by entering the details prompted
User details are stored
D
Users
2.1
Obtain Input
This process obtains credentials as input from the user
User details are checked if they are valid
2.2
Check Validity
This process checks if valid credentials are entered
D
dataset for ML algorithm
3
Chronic Disease prediction
Users can enter their symptoms and ML algo will check if user is at risk or not
The symptom is sent and disease is obtained
the entered parameters are stored
T
Parameters
ML algo
4
View if at risk or not
From ML algorithm, predict if at risk or not and display
5
Updating database
after analysis, save if at risk or not
Web server
6
Log out
Users can log out by clicking log out button