MANAGING E-MEDICARE DATABASE USING HADOOP DISTRIBUTED FILESYSTEM

Project Report submitted in partial fulfillment of the requirement for the degree of

Bachelor of Technology.

in

Computer Science & Engineering

under the Supervision of

Dr Rajni Mohana

By

Aanchal Mahajan (111305)

to

Jaypee University of Information and Technology

Waknaghat, Solan – 173234, Himachal Pradesh

CERTIFICATE

This is to certify that project report entitled "E-Medicare Database using Hadoop Distributed File System ", submitted by Aanchal Mahajan in partial fulfillment for the award of degree of Bachelor of Technology in Computer Science & Engineering to Jaypee University of Information Technology, Waknaghat, Solan has been carried out under my supervision.

This work has not been submitted partially or fully to any other University or Institute for the award of this or any other degree or diploma.

Supervisor's Name

Designation

Signature

Date:

<u>ACKNOWLEDGEMENT</u>

The satisfaction that accompanies with the completion of any task would be incomplete without mention of people whose ceaseless corporation made it possible, whose constant guidance and encouragement crown all efforts with success.

I take this opportunity to express my sincere gratitude to Head of Department(CSE), Brig. S.P. Ghrera, for co-operation in extension of all necessary facilities for the conduct of my work, Dr. Rajni Mohana, my Project Supervisor, for her kind and patient efforts in directing my project.

I would also like to thank all those who have directly or indirectly contributed to my project. Their co-operation and support has been a vital input.

Date: Name of the student

TABLE OF CONTENTS

ABBREVIATIONS AND SYMBOLS USED

HDFS:	Hadoop Distributed File System
DFD:	Data Flow Diagram

LIST OF FIGURES

LIST OF TABLES

Abstract

The project aims at developing an E-Medicare database, through which a user can get the probable diseases from the symptoms and the system will recommend a suitable doctor for the same.

The user can register on the website to avail this facility. Once the user has registered, he/she can now login through the website and search his/her query accordingly by entering their symptoms on the search bar. The results are displayed depending upon the combined number of occurrences of the keywords. The user can also write a review on the basis of which the website and the services provided by the website will be improved.

For this project a dataset is used which will be containing the symptoms of the diseases and also a list of suitable doctors for the treatment of the same which is present in the HDFS, further processed by the MapReduce framework to generate results.

CHAPTER- 1

Introduction

The vision of my project is to use technology as an enabler to facilitate and foster the availability and accessibility to quality healthcare services and information security to the healthcare seeker, operational efficiency and visibility to the healthcare expert. As with the increase in use of the Internet, E-Medicare systems will soon be very popular and will provide medical information to the users according to their query or the problems they have. It will provide the information health services suiting to the need of the user.

The demand for E- Medicare systems are rising these days because around the world, every health care system is struggling with rising costs and uneven quality despite the hard work of well-intentioned, well-trained clinicians. Health care leaders and policy makers have tried countless incremental fixes—attacking fraud, reducing errors, enforcing practice guidelines, making patients better "consumers."

At core it is maximizing value for patients: that is, **achieving the best outcomes at the lowest cost**. We must move away from a supply-driven health care system organized around what physicians do and toward a patient-centered system organized around what patients need. We must shift the focus from the volume and profitability of services provided—physician visits, hospitalizations, procedures, and tests—to the patient outcomes achieved. And we must replace today's fragmented system, in which every local provider offers a full range of services, with a system in which services for particular medical conditions are concentrated in health-delivery organizations and in the right locations to deliver high-value care.

In a nutshell the E-Medicare system that is needed and which will change the system will have features which include management of patient database with the capability of working online and offline. It should also have facility for supporting local languages, SMS alerts for patients, forum for sharing complicated problems with attachments, tracking migrated patients and a lot more.

The domain of an E-Medicare project is very vast, so on a small scale this project aims at developing an expert system that will take a query as an input from the user and diagnose the probable disease and also recommend a suitable doctor for the same.

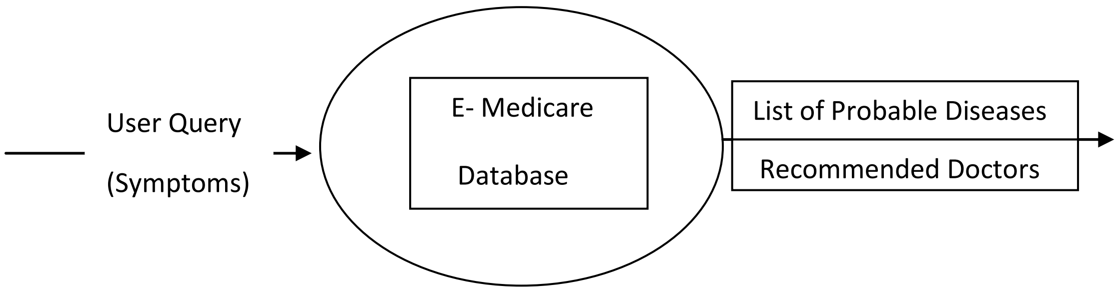


Figure: 1 E- Medicare Database

As medical information accounts for a lot of data so the project is using Apache Hadoop. Apache Hadoop is an open-source software framework for distributed storage and distributed processing of Big Data on clusters of commodity hardware. Its Hadoop Distributed File System (HDFS) splits files into large blocks and distributes the blocks amongst the nodes in the cluster. Hadoop makes it possible to run applications on systems with thousands of nodes involving thousands of terabytes. Hadoop was inspired by Google's MapReduce, a software framework in which an application is broken down into numerous small parts. Any of these parts (also called fragments or blocks) can be run on any node in the cluster.

Hadoop Distributed File System has various advantages like:

- Scalability

  Hadoop is a highly scalable storage platform, because it can store and distribute very large datasets across hundreds of inexpensive servers that operate in parallel. Unlike traditional relational database systems (RDBMS) that can’t scale to process large amounts of data, Hadoop enables businesses to run applications on thousands of nodes involving thousands of terabytes of data.

- Cost Effective

  Hadoop, on the other hand, is designed as a scale-out architecture that can affordably store all of a company's data for later use. The cost savings are staggering: instead of costing thousands to tens of thousands of pounds per terabyte, Hadoop offers computing and storage capabilities for hundreds of pounds per terabyte.

- Flexible

  Hadoop enables businesses to easily access new data sources and tap into different types of data (both structured and unstructured) to generate value from that data. This means businesses can use Hadoop to derive valuable business insights from data sources such as social media, email conversations or clickstream data. In addition, Hadoop can be used for a wide variety of purposes, such as log processing, recommendation systems, data warehousing, market campaign analysis and fraud detection.

- Fast

  Hadoop's unique storage method is based on a distributed file system that basically 'maps' data wherever it is located on a cluster. The tools for data processing are often on the same servers where the data is located, resulting in much faster data processing. If you're dealing with large volumes of unstructured data, Hadoop is able to efficiently process terabytes of data in just minutes, and petabytes in hours.

- Resilient to Failure

  A key advantage of using Hadoop is its fault tolerance. When data is sent to an individual node, that data is also replicated to other nodes in the cluster, which means that in the event of failure, there is another copy available for use. The MapReduce distribution goes beyond that by eliminating the NameNode and replacing it with a distributed No NameNode architecture that provides true high availability. Our architecture provides protection from both single and multiple failures. When it comes to handling large data sets in a safe and cost-effective manner, Hadoop has the advantage over relational database management systems, and its value for any size business will continue to increase as unstructured data continues to grow.

Further the project report will cover various topics partitioned into chapters, where each chapter describes every topic in full depth and detail.

Chapter -2 covers the E-Medicare website which lists all the modules that are present in the E-Medicare website. The modules are Home Page, About us Page, Healthy Living Page, Signup Page, Login Page, Search Page, Review Page.

Chapter -3 introduces the Hadoop Distributed File System explaining its assumptions and goals which include Hardware Failure, Large Datasets, Platform Independence and then later explaining the architecture which comprises of Name Nodes and Data Nodes and Data Organization.

Chapter- 4 talks about the Map Reduce Framework as to what this framework is, its architecture, the flow of control after which it gives detailed explanation of the Map Function, Combine Function and Reduce Function.

Chapter- 5 focuses on System Requirements that is detailed requirement analysis covering the hardware and software requirements for Ubuntu, Xampp and Hadoop. Further the chapter also talks about Functional and Non- Functional Requirements.

Chapter- 6 brings to light the system design which initially talks about the physical and logical design and then later the system design is explained through UML Diagrams like Usecase Diagram, Sequence Diagram, Activity Diagram, Data Flow Diagram and ER Diagram.

Chapter- 7 presents the implementation of the project giving the pseudo codes of the Map and Reduce Functions, screenshots of the various modules of the E-Medicare website, screenshots of the dataset and the index file and finally the screenshots of the final sample output.

CHAPTER-2

E-Medicare Website

E-Medicare website has the following modules:

2.1) Home Page:

This module is the homepage of the website where the user can read about our website and services provided by us.

2.2) About us Page:

This module describes the website and the services that are provided by the website.

2.3) Healthy Living Page:

This page gives tips on how to lead a healthy life and the daily routines that should be followed for a healthy lifestyle.

2.4) Signup Page:

Through this page a guest can register as a user and avail the services provided by us by entering the E-mail id, First Name, Last Name, Password, Street, City, State, Country, Mobile No and agreeing to the Terms and Conditions of our website.

2.5) Login Page:

Through this page a user can login post registering for our website and then can further search accordingly.

2.6) Search Page:

Just after the login the user is directed to the search page where the user can enter his/her query.

2.7) Review Page:

Through this page a user or a guest can post any reviews about our website or our services by entering the Subject of the Review, Review and additional information like E-mail id, First Name, Last Name, State, Country.

CHAPTER-3

Hadoop Distributed File System (HDFS)

3.1) Introduction

[1] The Hadoop Distributed File System (HDFS) is a distributed file system designed to run on commodity hardware. It has many similarities with existing distributed file systems. However, the differences from other distributed file systems are significant. HDFS is highly fault-tolerant and is designed to be deployed on low-cost hardware. HDFS provides high throughput access to application data and is suitable for applications that have large data sets. HDFS relaxes a few POSIX requirements to enable streaming access to file system data.

3.2) Assumptions and Goals

3.2.1) Hardware Failure

An HDFS instance may consist of hundreds or thousands of server machines, each storing part of the file system’s data. The fact that there are a huge number of components and that each component has a non-trivial probability of failure means that some component of HDFS is always non-functional. Therefore, detection of faults and quick automatic recovery from them is a core architectural goal of HDFS.

3.2.2) Streaming Data Access

Applications that run on HDFS need streaming access to their data sets. HDFS is designed more for batch processing rather than interactive use by users. The emphasis is on high throughput of data access rather than low latency of data access.

3.2.3) Large Datasets

Applications that run on HDFS have large data sets. A typical file in HDFS is gigabytes to terabytes in size. Thus HDFS is tuned to support large files. It should provide high aggregate data bandwidth and scale to hundreds of nodes in a single cluster. It should support tens of millions of files in a single instance.

3.2.4) Platform Independent

HDFS has been designed to be easily portable from one platform to another. This facilitates widespread adoption of HDFS as a platform of choice for a large set of applications.

3.3) NameNodes and DataNodes

HDFS has a master/slave architecture. An HDFS cluster consists of a single NameNode, a master server that manages the file system namespace and regulates access to files by clients. In addition, there are a number of DataNodes, usually one per node in the cluster, which manage storage attached to the nodes that they run on. HDFS exposes a file system namespace and allows user data to be stored in files. Internally, a file is split into one or more blocks and these blocks are stored in a set of DataNodes. The NameNode executes file system namespace operations like opening, closing, and renaming files and directories. It also determines the mapping of blocks to DataNodes. The DataNodes are responsible for serving read and write requests from the file system's clients. The DataNodes also perform block creation, deletion, and replication upon instruction from the NameNode.

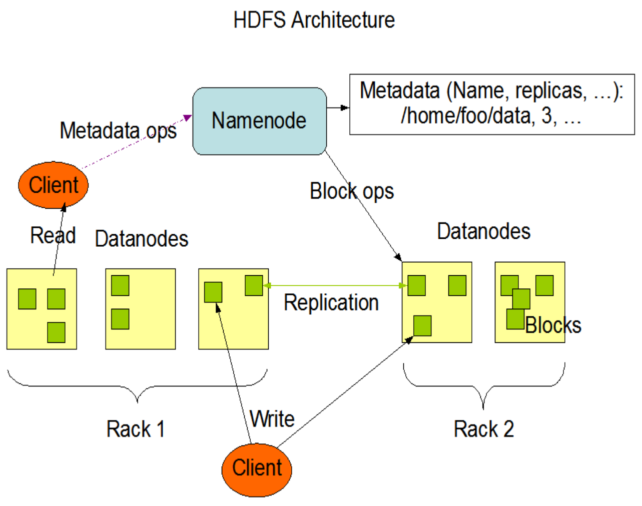


Figure: 2 HDFS Architecture

3.4) Data Organization

3.4.1) Data Blocks

HDFS is designed to support very large files. Applications that are compatible with HDFS are those that deal with large data sets. These applications write their data only once but they read it one or more times and require these reads to be satisfied at streaming speeds. HDFS supports write-once-read-many semantics on files. A typical block size used by HDFS is 64 MB. Thus, an HDFS file is chopped up into 64 MB chunks, and if possible, each chunk will reside on a different DataNode.

3.4.2) Staging

A client request to create a file does not reach the NameNode immediately. Initially the HDFS client caches the file data into a temporary local file. Application writes are transparently redirected to this temporary local file. When the local file accumulates data worth over one HDFS block size, the client contacts the NameNode. The NameNode inserts the file name into the file system hierarchy and allocates a data block for it. The NameNode responds to the client request with the identity of the DataNode and the destination data block. Then the client flushes the block of data from the local temporary file to the specified DataNode. When a file is closed, the remaining un-flushed data in the temporary local file is transferred to the DataNode. The client then tells the NameNode that the file is closed. At this point, the NameNode commits the file creation operation into a persistent store. If the NameNode dies before the file is closed, the file is lost.

Chapter- 4

Map-Reduce Framework

4.1) Overview

[2] Hadoop MapReduce is a software framework for easily writing applications which process vast amounts of data (multi-terabyte data-sets) in-parallel on large clusters of commodity hardware in a reliable, fault-tolerant manner.

A MapReduce job usually splits the input data-set into independent chunks which are processed by the map tasks in a completely parallel manner. The framework sorts the outputs of the maps, which are then input to the reduce tasks. Typically both the input and the output of the job are stored in a file-system. The framework takes care of scheduling tasks, monitoring them and re-executes the failed tasks.

Typically the compute nodes and the storage nodes are the same, that is, the MapReduce framework and the Hadoop Distributed File System are running on the same set of nodes. This configuration allows the framework to effectively schedule tasks on the nodes where data is already present, resulting in very high aggregate bandwidth across the cluster.

The MapReduce framework consists of a single master JobTracker and one slave TaskTracker per cluster-node. The master is responsible for scheduling the jobs component tasks on the slaves, monitoring them and re-executing the failed tasks. The slaves execute the tasks as directed by the master.

Minimally, applications specify the input/output locations and supply map and reduce functions via implementations of appropriate interfaces and/or abstract-classes. These, and other job parameters, comprise the job configuration. The Hadoop job client then submits the job (jar/executable etc.) and configuration to the JobTracker which then assumes the responsibility of distributing the software/configuration to the slaves, scheduling tasks and monitoring them, providing status and diagnostic information to the job-client.

4.2) Map Function

The Map operation is parallelized the input file set is first split to several pieces called File Splits. When an individual map task starts it will open a new output writer per configured reduce task. It will then proceed to read its FileSplit using the RecordReader it gets from the specified InputFormat. InputFormat parses the input and generates key-value pairs. It is not necessary for the InputFormat to generate both meaningful keys and values.

As key-value pairs are read from the RecordReader they are passed to the configured Mapper. The user supplied Mapper does whatever it wants with the input pair and calls OutputCollector.collect with key-value pairs of its own choosing. The output it generates must use one key class and one value class. This is because the Map output will be written into a SequenceFile which has per-file type information and all the records must have the same type.

When Mapper output is collected it is partitioned, which means that it will be written to the output specified by the Partitioner.

N input files will generate M map tasks to be run and each map task will generate as many output files as there are reduce tasks configured in the system. Each output file will be targeted at a specific reduce task and the map output pairs from all the map tasks will be routed so that all pairs for a given key end up in files targeted at a specific reduce task.

4.3) Combine Function

When the map operation outputs its pairs they are already available in memory. For efficiency reasons, sometimes it makes sense to take advantage of this fact by supplying a combiner class to perform a reduce-type function.

If a combiner is used then the map key-value pairs are not immediately written to the output. Instead they will be collected in lists, one list per each key value. When a certain number of key-value pairs have been written, this buffer is flushed by passing all the values of each key to the combiner's reduce method and outputting the key-value pairs of the combine operation as if they were created by the original map operation.

4.4) Reduce Function

When a reduce task starts, its input is scattered in many files across all the nodes where map tasks ran. If run in distributed mode these need to be first copied to the local file system.

Once all the data is available locally it is appended to one file in an append phase. The file is then merge sorted so that the key-value pairs for a given key are contiguous.

The file is read sequentially and the values are passed to the reduce method with an iterative reading the input file until the next key value is encountered.

Finally the output will consist of one output file per executed reduce task.

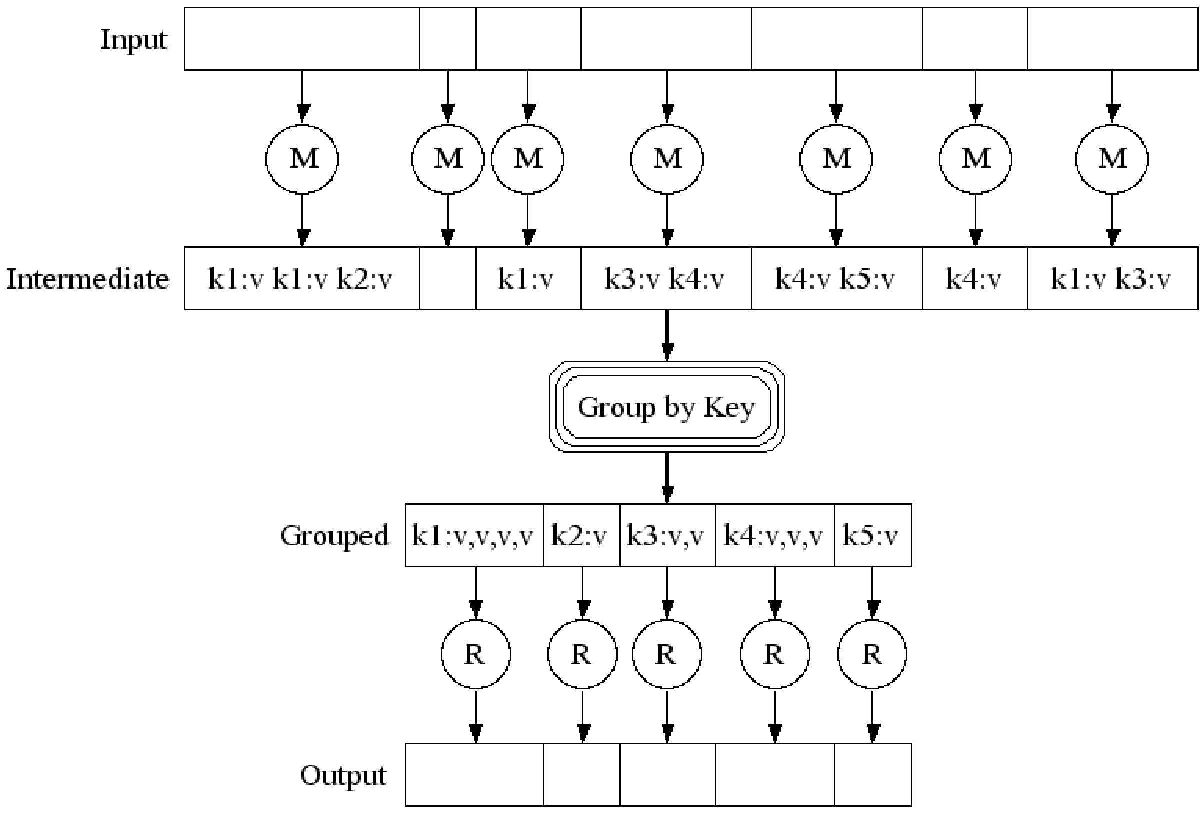


Figure: 3MapReduce Architecture

Chapter-5

System Requirements

5.1) Requirement Analysis

Information gathering is usually the first step of a project. The purpose of this phase is to identify and document the exact requirements for the system. The user's request identifies the need for a new information system. The objective is to determine whether the request is valid and feasible before a recommendation is made to build a new or an existing manual system.

The major steps are-

- Defining the user requirements
- Studying the present system to verify the problem

5.1.1) Hardware Requirements

a) For Ubuntu

Memory	1 GB
Processor	1
Hard Disk	40 GB
CD/DVD	Using ISO File Of Ubuntu 12.04LTS
RAM	2 GB
Network Adapter	NAT
USB Controller	Present
Sound Card	Auto Detect
Printer	Present
Display	Auto Detect

Table: 1 Hardware Requirements for Ubuntu 12.04LTS

b) For Hadoop

There is no single hardware requirement set for installing Hadoop.

c) For Xampp

[3] There are no limits on the type of hardware used to host an installation. However, as your needs grow, so will the need for additional resources. A simple installation requires about 50M of free disk space and 32M of free system memory. However, a minimum of 128M free memory available to PHP is recommended. Disk space should be allocated according to usage.

5.1.2) Software Requirements

a) [4] For Hadoop

- Java^{TM} 1.6.x, preferably from Sun, must be installed.
- ssh must be installed and sshd must be running to use the Hadoop scripts that manage remote Hadoop daemons.

b) For Xampp

[3] The installation requires an environment running the following:

- PHP version 5.1+ (PHP 5.2+ recommended)
- Mysql Server 4+ (Mysql 5 is strongly recommended)
- Web server (Apache 2+ recommended)
- Write access from the web server to some folders inside the efront installation

5.1.3) Functional Requirements

- The website must allow input of personal information from user.
- The website must request username and password for access to data, only after authentication will allow access to the system.
- On authentication the website will enable a user to search for a probable disease through the diagnosed symptoms.
- The website will display the diseases in order of probability.
- The website will enable user to search for multiple times with a number of keywords.

- The website will also enable user to post their reviews and can read the reviews posted by other users.
- The website also gives tips on how to lead a healthy life on a daily basis.

5.1.4) Non- Functional Requirements

Non-Function requirements mean the characteristics that are not related to system’s physical functions.

- **User Friendly**- The system should be easy to use, i.e. user friendly, for both administrator and external users. This can be done for providing function descriptions.
- **Secure**-The system should be secure. If not, hackers can access the database and undergo destruction.
- **Reliable-** The system must be resistant to failure.
- **Maintainability-** The website must be easy to maintain by the administrator.
- **Availability-** The services should be available 24X7.
- The user interface screen should be loaded immediately.
- The login information shall be verified within five seconds.
- Queries shall return results as soon as possible.

Chapter-6

System Design

6.1) Introduction

The objective of the system design is to deliver the requirements as specified. System design involves first logical design and then physical construction of the system. The logical design describes structure and characteristics of features, such as outputs, inputs, files, databases and procedures. The physical construction produces actual program software, files and a working system.

System design goes through two phases of development-

6.1.1) Logical Design

We know that a data flow diagram shows the logical flow of the system and defines the boundary of the system. Logical design specifies the user need at a level of details that virtually determine the information flow into and out of the system and the required data resources. Logical design describes the input, output, database and procedures, all in a format that meets user requirements.

6.1.2) Physical Design

It provides the working system by designing the design specification that tells programmers what exactly the system must do. In short it can state that physical design is the implementation of the logical design.

Physical system design must consist of the following-

a) Design the physical system

- Specify input, output media
- Design the database and specify the backup procedures
- Design physical information through the system

b) Plan system implementation

6.2) Use Case Diagram

Use Case Diagram is the basic analysis diagram. It is the first analysis diagram. It gives interaction of the system with entities outside the system. These entities are called actors. Actors are the users of the system or other systems who give input to the system and take output from the system. The various scenarios in the system are called use case. They are denoted by oval. The complete system has been explained using a basic use case diagram which shows the interaction between the main actors and a set of use case scenarios.

The detailed use case diagrams following the basic use case diagram gives the detailed interaction of each actor with the system.

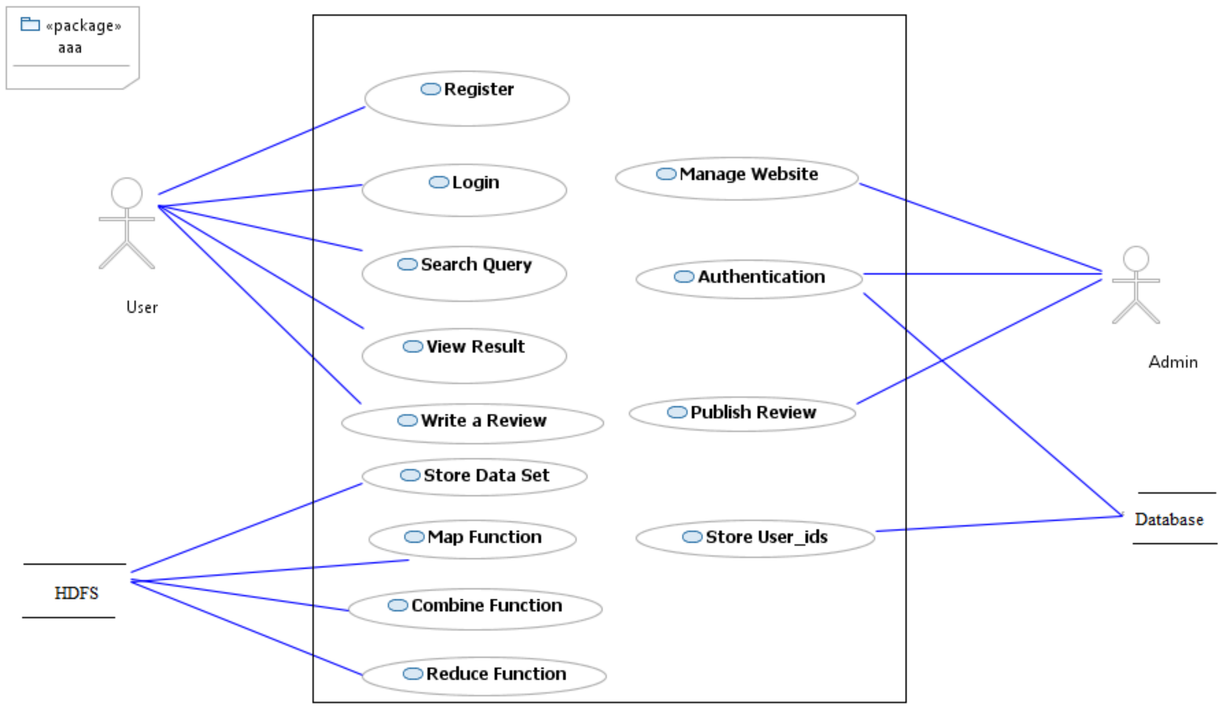


Figure: 4 Use Case Diagram

6.2.1) Actors Involved

a) User

- Register through the website
- Login after registration
- Search for a query

- View Result

b) Administrator

- Manages the website
- Authenticates the user login
- Publishes reviews

c) Database

- Stores user ids
- Authentication of user login

d) HDFS

- Stores dataset
- Performs map function
- Performs combine function
- Performs reduce function

6.3) Sequence Diagram

It is the analysis diagram which gives the sequence of events taking place in the system. The diagram gives the objects in the system and their interaction among each other. The diagram also shows the messages passed between the objects.

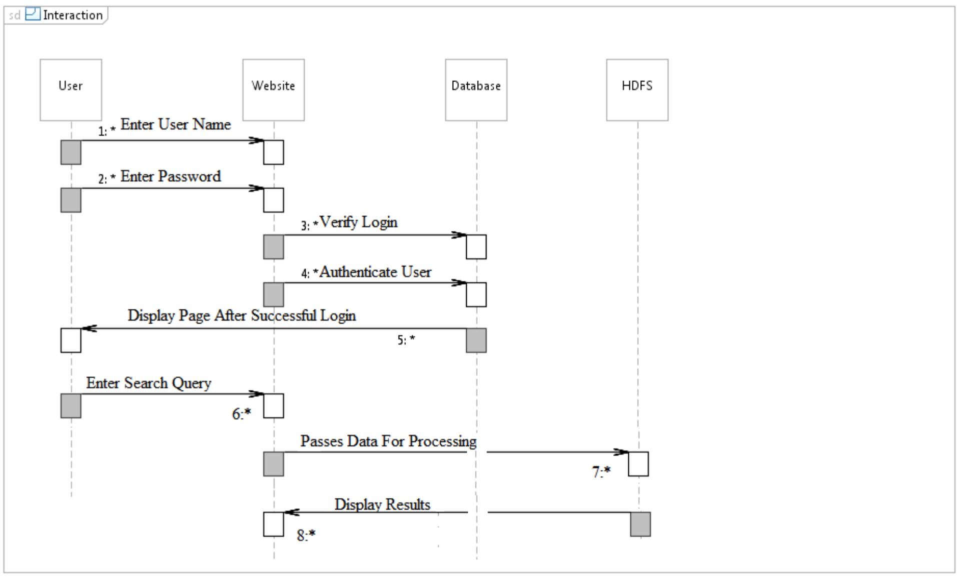


Figure: 5 Sequence Diagram

6.4) Activity Diagram

This is another UML diagram used in object oriented analysis. This diagram gives the list of activities being performed in the system. It starts with a filled circle and ends with bull's eye. A condition can also be shown in activity diagram using a diamond having more than one output. Activity is a particular operation of the system. Activity diagrams are used for visualizing dynamic nature of a system. The specific usage is to model the control flow from one activity to another. This control flow does not include messages.

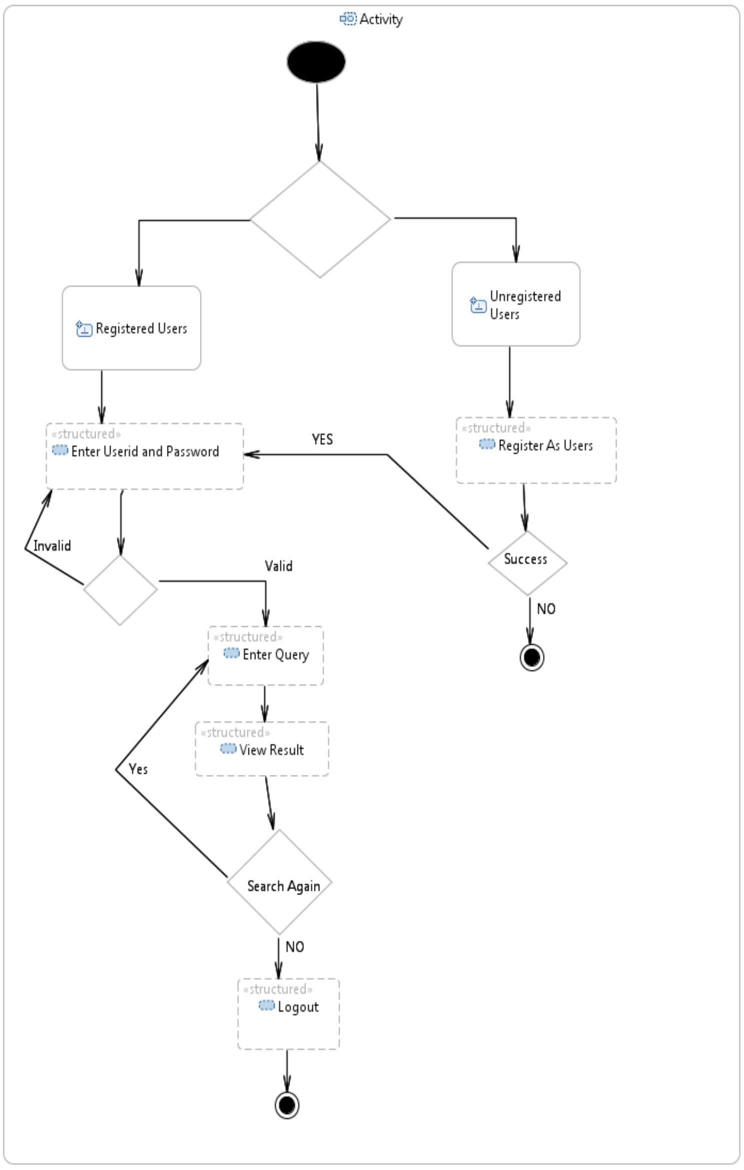


Figure: 6 System Activity Diagram

6.4.1) Flow Of Control In HDFS

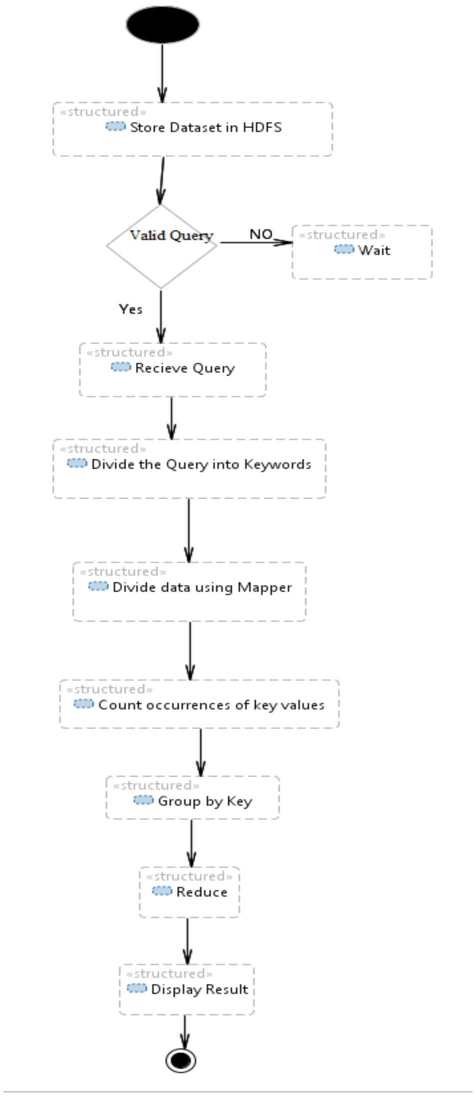

«structured» Store Dataset in HDFS
Valid Query
NO
«structured» Wait
Yes
«structured» Recieve Query
«structured» Divide the Query into Keywords
«structured» Divide data using Mapper
«structured» Count occurrences of key values
«structured» Group by Key
«structured» Reduce
«structured» Display Result

Figure: 7 Control Flow

6.5) Data Flow Diagram

Information is transformed as it flows through a computer based system. The system accepts input in a variety of forms and produces output in varied forms. For depicting this information flow in functional modelling, Data Flow Diagram (DFDs) are used.

A Data Flow Diagram is a graphical representation that depicts information flow and the transforms that are applied as data move from input to output. The data flow diagram may be used to represent the system or software at any level of abstraction. DFDs may be partitioned into levels that represent increasing information flow and functional details. The DFD provides mechanism for functional modelling as well as information flow modelling.

6.5.1) Level – 0 DFD

A level 0 DFD, also called a fundamental system model or a context model, represents the entire software element as a single bubble with input and output data indicated by incoming and outgoing arrows, respectively

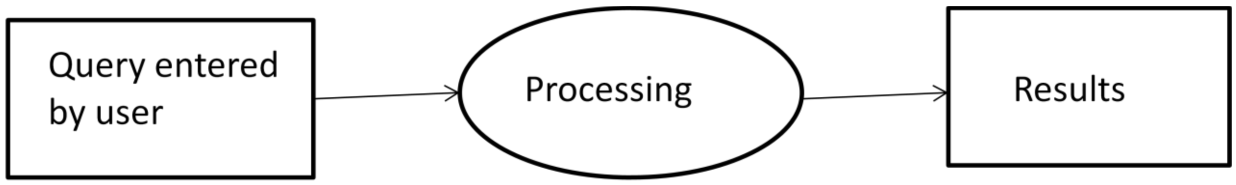


Figure: 8 Level-0 DFD

6.5.2) Level – 1 DFD

Additional processes (bubbles) and information flow paths are represented as level 0 DFD is partitioned to reveal more detail. The input and output remains the same but the process is broken down.

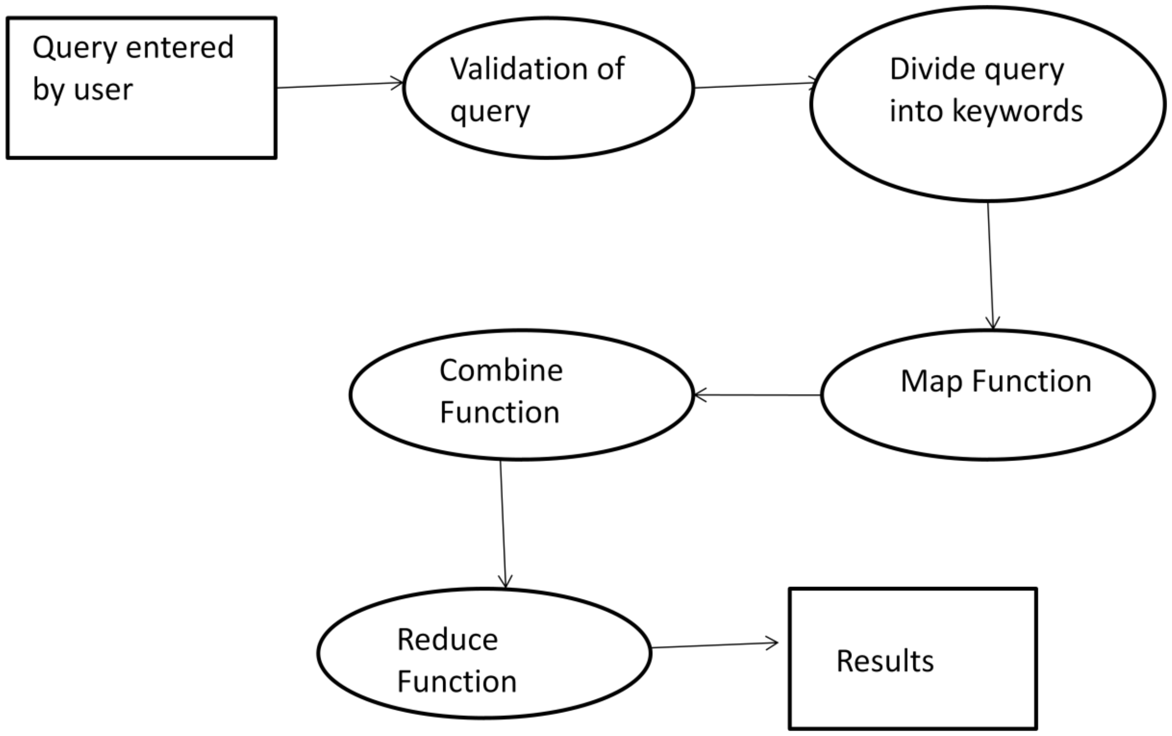


Figure: 9 Level-1 DFD

6.6) ER Diagram

An entity–relationship model (ER model) is a data model for describing the data or information aspects of a business domain or its process requirements, in an abstract way. The main components of ER models are entities (things) and the relationships that can exist among them, and databases.

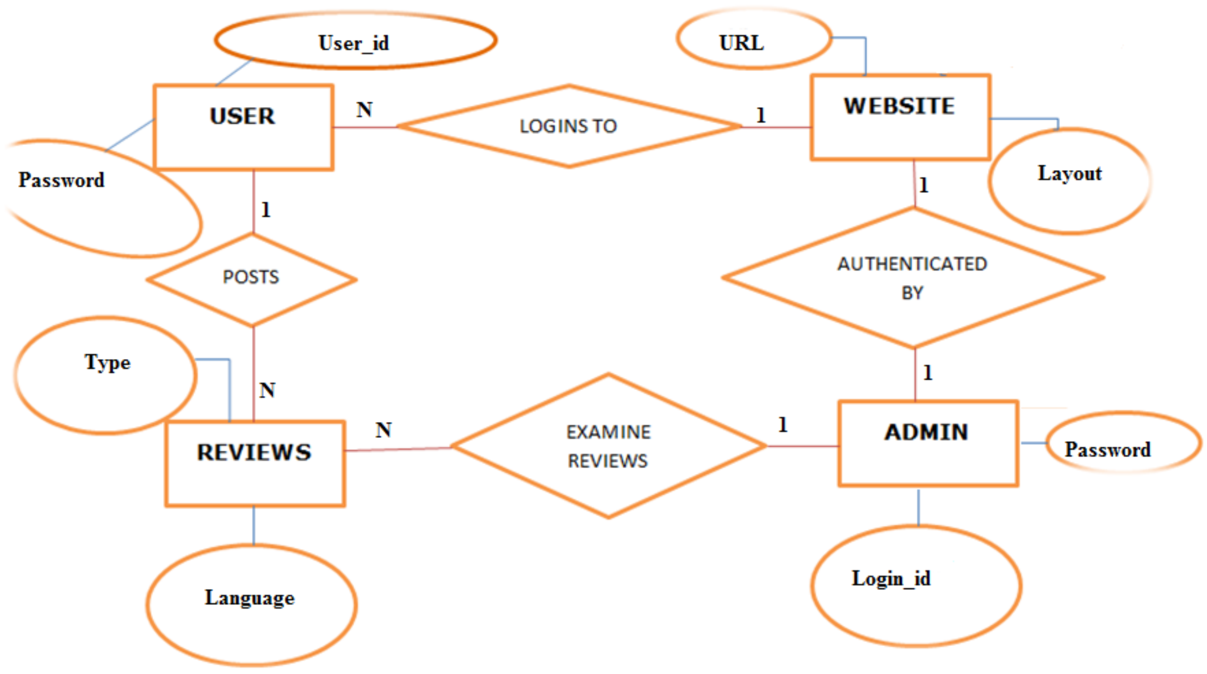


Figure: 10 ER Diagram

CHAPTER-7

Implementation

7.1) Pseudo Code

7.1.1) Map Function

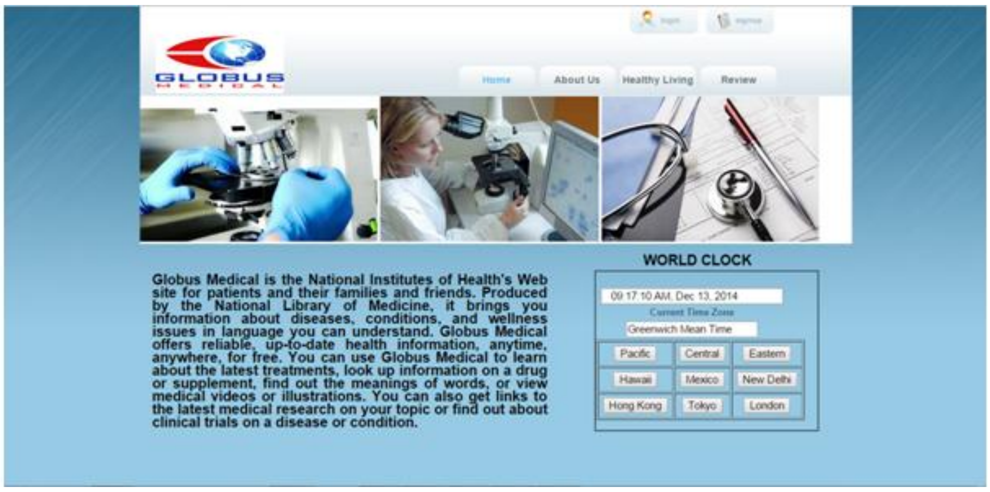
```
Begin:
        For each InputSplit generated by InputFormat,
        Replace "." and "a-z" characters by " " in the InputSplit
        Open the file which contains keywords
        do
                Read keyword from the file
                Convert InputSplit into Tokens
                do
                Save next Token every time the Token "Disease" appears
                Compare a Token with the keyword from the file
                if  Token and keyword are equal
                        Read file containing indexes of the diseases
                        Search the index previously saved
                        Print Name of Disease to output file
                        Generate an output key-value pair
                end
        end
end
```

7.1.2) Reduce Function

```
Begin:
    For every Mapper
    Copy the sorted output of each Mapper
    Sorting the input of the reduce function by Keys
    do
        Sum all the occurrences of Keys
    end
    Write the final output to the output file
end
```

7.2) Screenshots

7.2.1) Home Page



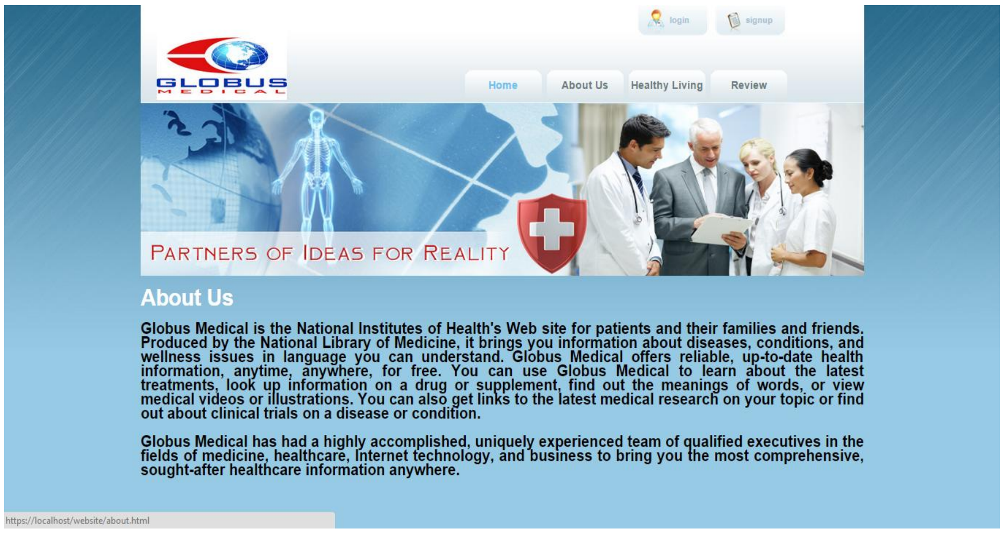
Globus Medical is the National Institutes of Health's Web site for patients and their families and friends. Produced by the National Library of Medicine, it brings you information about diseases, conditions, and wellness issues in language you can understand. Globus Medical offers reliable, up-to-date health information, anytime, anywhere, for free. You can use Globus Medical to learn about the latest treatments, look up information on a drug or supplement, find out the meanings of words, or view medical videos or illustrations. You can also get links to the latest medical research on your topic or find out about clinical trials on a disease or condition.

09 17 10 AM, Dec 13, 2014

Greenwich Mean Time

Figure: 11 Home Page

7.2.2) About us Page:


login
signup
GLOBUS MEDICAL
Home
About Us
Healthy Living
Review
PARTNERS OF IDEAS FOR REALITY
https://localhost/website/about.html

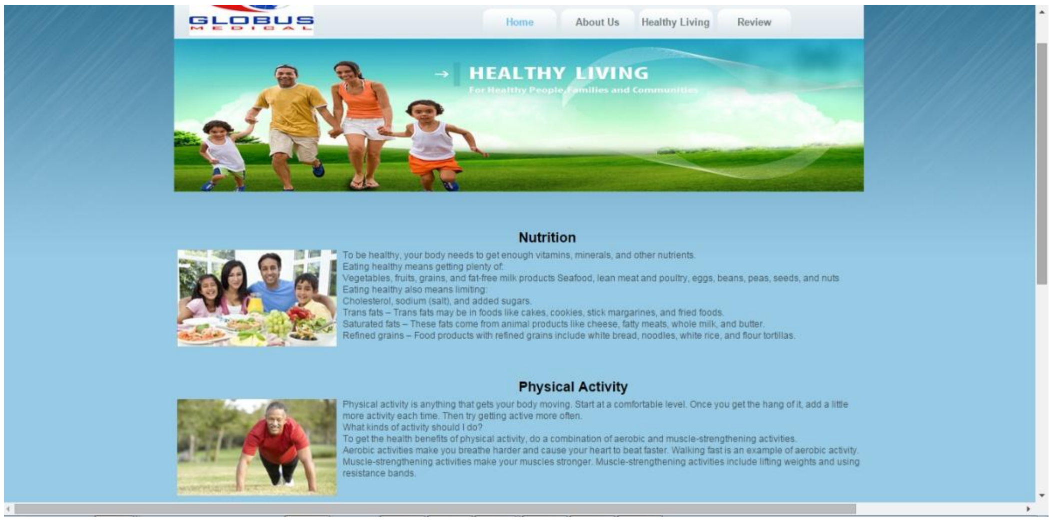
About Us

Globus Medical is the National Institutes of Health's Web site for patients and their families and friends. Produced by the National Library of Medicine, it brings you information about diseases, conditions, and wellness issues in language you can understand. Globus Medical offers reliable, up-to-date health information, anytime, anywhere, for free. You can use Globus Medical to learn about the latest treatments, look up information on a drug or supplement, find out the meanings of words, or view medical videos or illustrations. You can also get links to the latest medical research on your topic or find out about clinical trials on a disease or condition.

Globus Medical has had a highly accomplished, uniquely experienced team of qualified executives in the fields of medicine, healthcare, Internet technology, and business to bring you the most comprehensive, sought-after healthcare information anywhere.

Figure: 12 Home Page

7.2.3) Healthy Living Page:



Nutrition

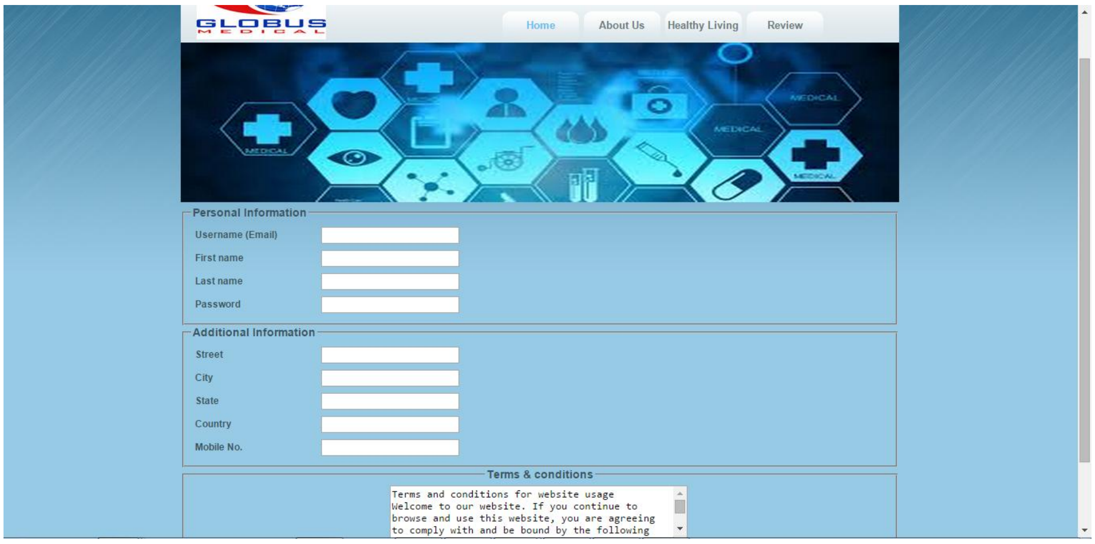
To be healthy, your body needs to get enough vitamins, minerals, and other nutrients.
Eating healthy means getting plenty of:
Vegetables, fruits, grains, and fat-free milk products Seafood, lean meat and poultry, eggs, beans, peas, seeds, and nuts
Eating healthy also means limiting:
Cholesterol, sodium (salt), and added sugars.
Trans fats – Trans fats may be in foods like cakes, cookies, stick margarines, and fried foods.
Saturated fats – These fats come from animal products like cheese, fatty meats, whole milk, and butter.
Refined grains – Food products with refined grains include white bread, noodles, white rice, and flour tortillas.

Physical Activity

Physical activity is anything that gets your body moving. Start at a comfortable level. Once you get the hang of it, add a little more activity each time. Then try getting active more often.
What kinds of activity should I do?
To get the health benefits of physical activity, do a combination of aerobic and muscle-strengthening activities.
Aerobic activities make you breathe harder and cause your heart to beat faster. Walking fast is an example of aerobic activity.
Muscle-strengthening activities make your muscles stronger. Muscle-strengthening activities include lifting weights and using resistance bands.

Figure: 13 Healthy Living Page

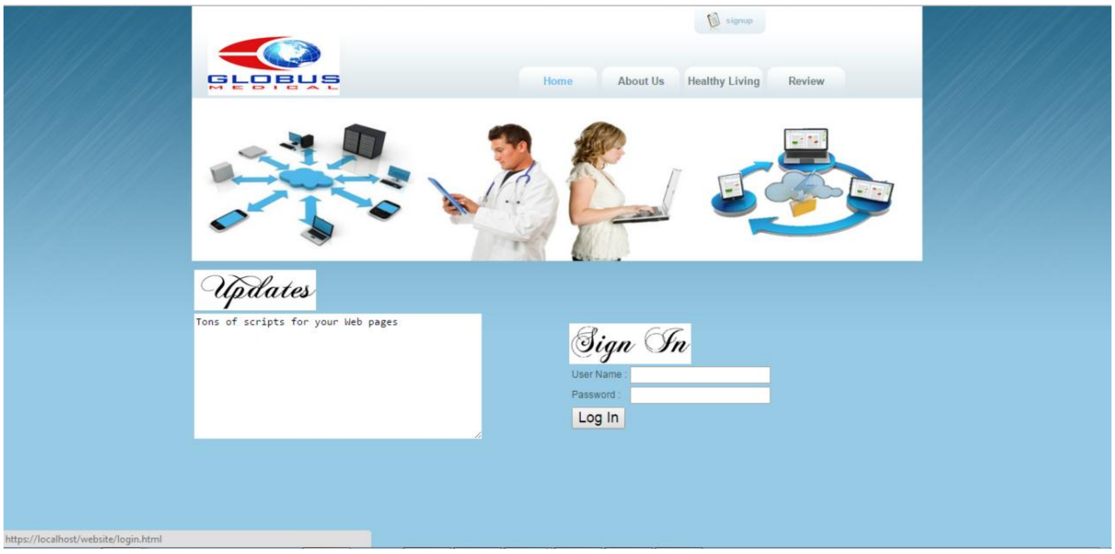
7.2.4) Signup Page:



Terms and conditions for website usage
Welcome to our website. If you continue to
browse and use this website, you are agreeing
to comply with and be bound by the following

Figure: 14 Signup Page

7.2.5) Login Page:



Tons of scripts for your Web pages

Figure: 15 Login Page

7.2.6) Search Page:

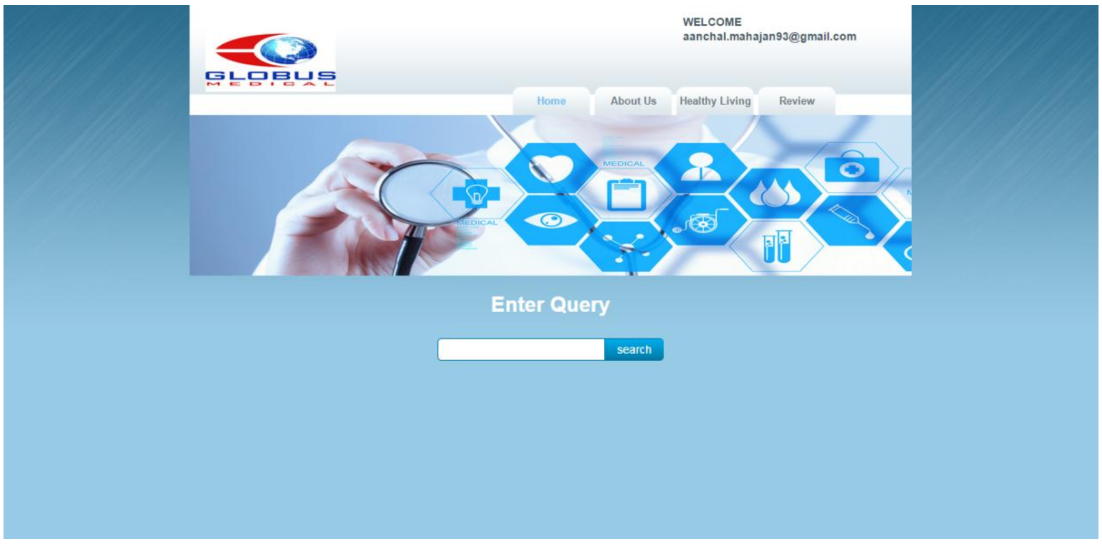


Figure: 16 Search Page

7.2.7) Review Page:

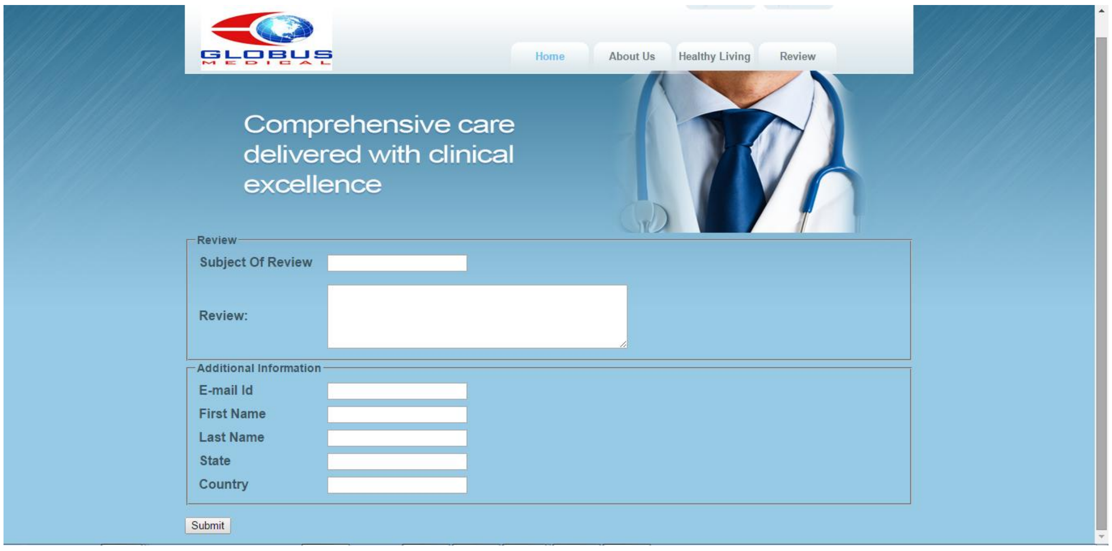
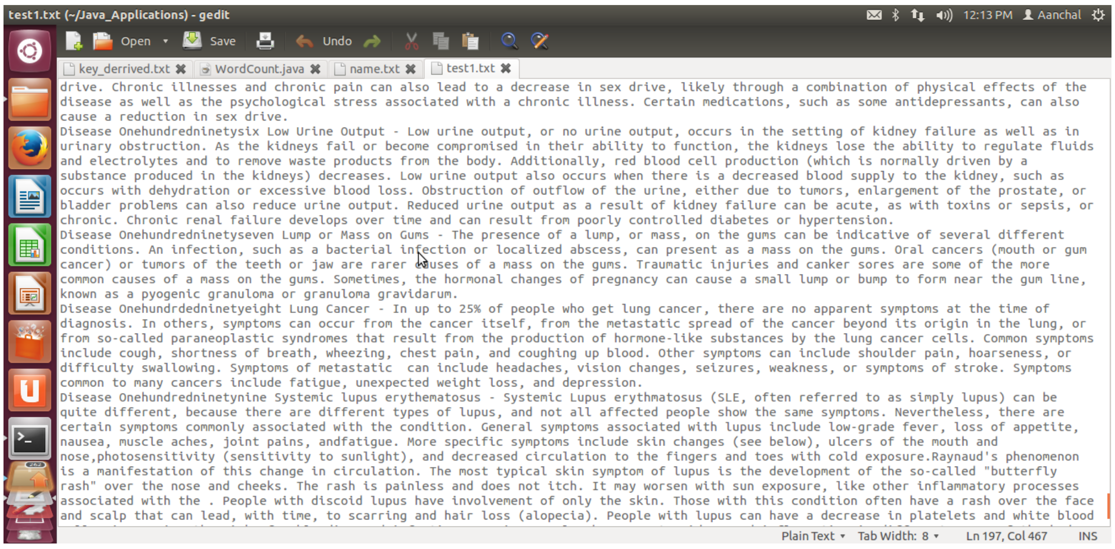


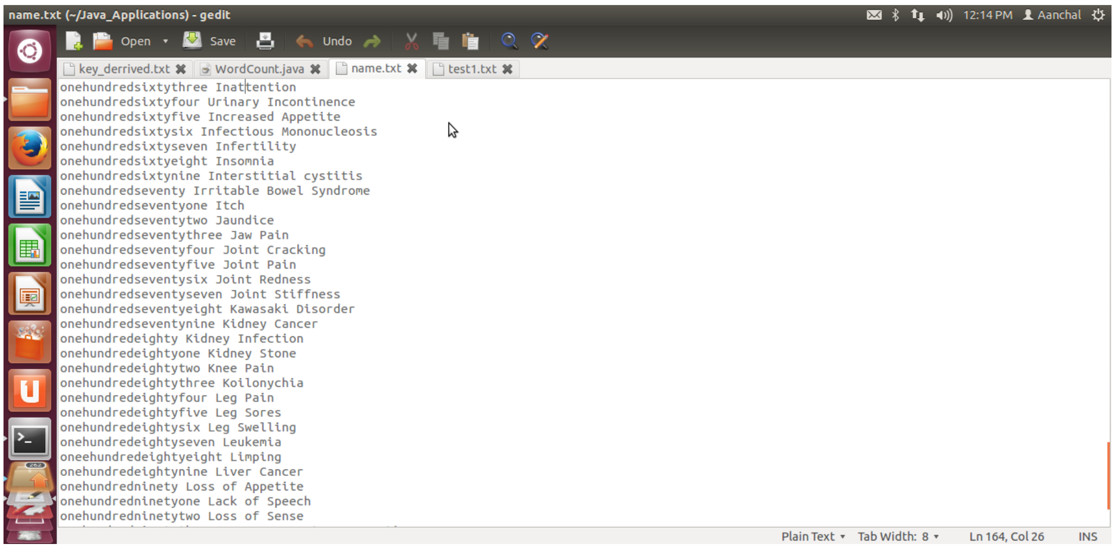
Figure: 17 Review Page

7.2.8) Details of Dataset



```
drive. Chronic illnesses and chronic pain can also lead to a decrease in sex drive, likely through a combination of physical effects of the
disease as well as the psychological stress associated with a chronic illness. Certain medications, such as some antidepressants, can also
cause a reduction in sex drive.
Disease Onehundredninetysix Low Urine Output - Low urine output, or no urine output, occurs in the setting of kidney failure as well as in
urinary obstruction. As the kidneys fail or become compromised in their ability to function, the kidneys lose the ability to regulate fluids
and electrolytes and to remove waste products from the body. Additionally, red blood cell production (which is normally driven by a
substance produced in the kidneys) decreases. Low urine output also occurs when there is a decreased blood supply to the kidney, such as
occurs with dehydration or excessive blood loss. Obstruction of outflow of the urine, either due to tumors, enlargement of the prostate, or
bladder problems can also reduce urine output. Reduced urine output as a result of kidney failure can be acute, as with toxins or sepsis, or
chronic. Chronic renal failure develops over time and can result from poorly controlled diabetes or hypertension.
Disease Onehundredninetyseven Lump or Mass on Gums - The presence of a lump, or mass, on the gums can be indicative of several different
conditions. An infection, such as a bacterial infection or localized abscess, can present as a mass on the gums. Oral cancers (mouth or gum
cancer) or tumors of the teeth or jaw are rarer causes of a mass on the gums. Traumatic injuries and canker sores are some of the more
common causes of a mass on the gums. Sometimes, the hormonal changes of pregnancy can cause a small lump or bump to form near the gum line,
known as a pyogenic granuloma or granuloma gravidarum.
Disease Onehundrdednінetyeight Lung Cancer - In up to 25% of people who get lung cancer, there are no apparent symptoms at the time of
diagnosis. In others, symptoms can occur from the cancer itself, from the metastatic spread of the cancer beyond its origin in the lung, or
from so-called paraneoplastic syndromes that result from the production of hormone-like substances by the lung cancer cells. Common symptoms
include cough, shortness of breath, wheezing, chest pain, and coughing up blood. Other symptoms can include shoulder pain, hoarseness, or
difficulty swallowing. Symptoms of metastatic  can include headaches, vision changes, seizures, weakness, or symptoms of stroke. Symptoms
common to many cancers include fatigue, unexpected weight loss, and depression.
Disease Onehundredninetynine Systemic lupus erythematosus - Systemic Lupus erythmatosus (SLE, often referred to as simply lupus) can be
quite different, because there are different types of lupus, and not all affected people show the same symptoms. Nevertheless, there are
certain symptoms commonly associated with the condition. General symptoms associated with lupus include low-grade fever, loss of appetite,
nausea, muscle aches, joint pains, andfatigue. More specific symptoms include skin changes (see below), ulcers of the mouth and
nose,photosensitivity (sensitivity to sunlight), and decreased circulation to the fingers and toes with cold exposure.Raynaud's phenomenon
is a manifestation of this change in circulation. The most typical skin symptom of lupus is the development of the so-called "butterfly
rash" over the nose and cheeks. The rash is painless and does not itch. It may worsen with sun exposure, like other inflammatory processes
associated with the . People with discoid lupus have involvement of only the skin. Those with this condition often have a rash over the face
and scalp that can lead, with time, to scarring and hair loss (alopecia). People with lupus can have a decrease in platelets and white blood
```

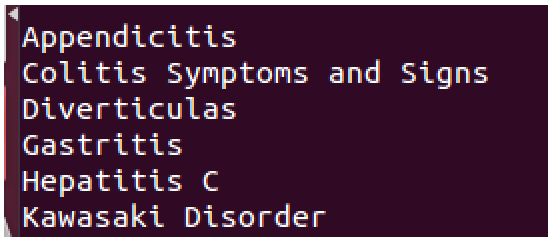
Figure 18: Dataset



```
onehundredsixtythree Inattention
onehundredsixtyfour Urinary Incontinence
onehundredsixtyfive Increased Appetite
onehundredsixtysix Infectious Mononucleosis
onehundredsixtyseven Infertility
onehundredsixtyeight Insomnia
onehundredsixtynine Interstitial cystitis
onehundredseventy Irritable Bowel Syndrome
onehundredseventyone Itch
onehundredseventytwo Jaundice
onehundredseventythree Jaw Pain
onehundredseventyfour Joint Cracking
onehundredseventyfive Joint Pain
onehundredseventysix Joint Redness
onehundredseventyseven Joint Stiffness
onehundredseventyeight Kawasaki Disorder
onehundredseventynine Kidney Cancer
onehundredeighty Kidney Infection
onehundredeightyone Kidney Stone
onehundredeightytwo Knee Pain
onehundredeightythree Koilonychia
onehundredeightyfour Leg Pain
onehundredeightyfive Leg Sores
onehundredeightysix Leg Swelling
onehundredeightyseven Leukemia
oneehundredeightyeight Limping
onehundredeightynine Liver Cancer
onehundredninety Loss of Appetite
onehundredninetyone Lack of Speech
onehundredninetytwo Loss of Sense
```

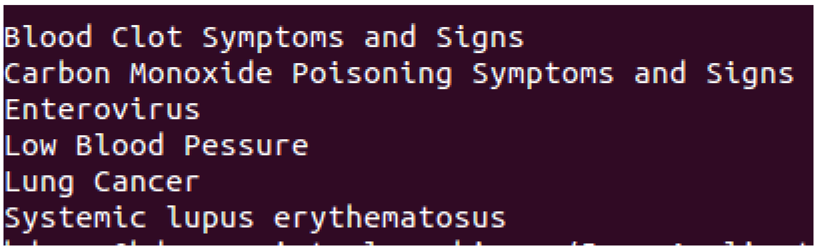
Figure 19: Index File

7.2.9) Output of MapReduce



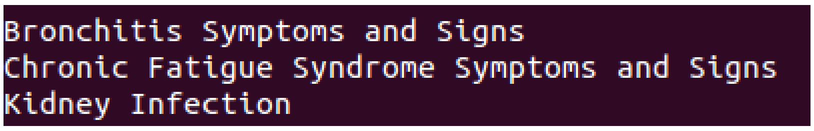
```
Appendicitis
Colitis Symptoms and Signs
Diverticulas
Gastritis
Hepatitis C
Kawasaki Disorder
```

Figure 20: Sample Output 1



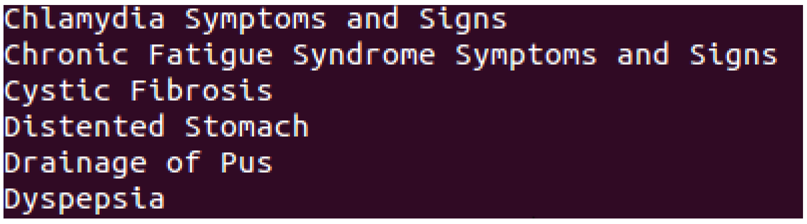
```
Blood Clot Symptoms and Signs
Carbon Monoxide Poisoning Symptoms and Signs
Enterovirus
Low Blood Pessure
Lung Cancer
Systemic lupus erythematosus
```

Figure 21: Sample Output 2



```
Bronchitis Symptoms and Signs
Chronic Fatigue Syndrome Symptoms and Signs
Kidney Infection
```

Figure 22: Sample Output 3



```
Chlamydia Symptoms and Signs
Chronic Fatigue Syndrome Symptoms and Signs
Cystic Fibrosis
Distented Stomach
Drainage of Pus
Dyspepsia
```

Figure 23: Sample Output 4

Conclusion and Future Work

The demand for E- Medicare systems are rising these days because around the world, every health care system is struggling with rising costs and uneven quality despite the hard work of well-intentioned, well-trained clinicians. Health care leaders and policy makers have tried countless incremental fixes—attacking fraud, reducing errors, enforcing practice guidelines, making patients better “consumers.”

The project “Managing E-Medicare Database Using Hadoop Distributed File System” involves user to successfully register and login to the website and queries data related to medical field. The user can get the name of the diseases displayed based on the symptoms he has entered and will get the names of the recommended doctors for the same.

In Future the project can be extended to process reviews and posts, such that meaningful data can be extracted for providing better medical services. The user feedback can also be considered to optimize the results.

References

Reference Links

[1] Hadoop 1.2.1 Documentation, HDFS Architecture Guide [online] 2013, http://hadoop.apache.org/docs/r1.2.1/hdfs_design.html (Accessed: 17th August 2014)

[2] Hadoop 1.2.1 Documentation, MapReduce Tutorial [online] 2013, http://hadoop.apache.org/docs/r1.2.1/mapred_tutorial.html (Accessed: 25th August 2014)

[3] Hadoop Installation, Installation Instructions [online] 2013 http://wiki.efrontlearning.net/index.php?title=Installation_instructions&revision=19 (Accessed: 4th September 2014)

[4] Hadoop 1.2.1 Documentation, Single Node Setup [online] 2013 http://hadoop.apache.org/docs/r1.2.1/single_node_setup.html (Accessed: 7th September 2014)

Research Papers

[1] Kyuseok Shim, MapReduce Algorithms for Big Data Analysis, DNIS 2013, LNCS 7813, pp. 44–48, 2013

[2] DunrenChe, MejdlSafran, and ZhiyongPeng, From Big Data to Big Data Mining: Challenges, Issues, and Opportunities, DASFAA Workshops 2013, LNCS 7827, pp. 1–15, 2013.

Reference Book

Tom White, Hadoop: The definitive guide, 3d. ed. O'Reilly Media, 2012.

Appendix – A

Installation of Hadoop

Assumptions

1. You're running 64-bit Windows
2. Your laptop has more than 4 GB of RAM

Download List (No specific order)

1 VMWare Player – allows you to run virtual machines with different operating systems

2 Ubuntu 12.04 LTS – Linux operating system with a nice user interface

Instructions to Install Hadoop

1. Install VMWare Player
2. Create a new virtual machine
3. Point the installer disc image to the ISO file (Ubuntu) that you just downloaded
4. User name should be hduser
5. Hard disk space 40 GB Hard drive (more is better, but you want to leave some for your Windows machine)
6. Customize hardware
    a. Memory: 2 GB RAM
    b. Processors: 2
7. Launch your virtual machine (all the instructions after this step will be performed in Ubuntu)
8. Login to hduser
9. Open a terminal window with Ctrl + Alt + T (you will use this keyboard shortcut a lot)
10. Install Java JDK 7
    a. Download the Java JDK
    b. Unzip the file

tar -xvf jdk-7u21-linux-x64.tar.gz

c. Now move the JDK 7 directory to /usr/lib

   sudomkdir -p /usr/lib/jvm

   sudo mv ./jdk1.7.0_21 /usr/lib/jvm/jdk1.7.0

d. Now run

   sudo update-alternatives --install "/usr/bin/java" "java" "/usr/lib/jvm/jdk1.7.0/bin/java"

   sudo update-alternatives --install "/usr/bin/javac" "javac" "/usr/lib/jvm/jdk1.7.0/bin/javac"

   sudo update-alternatives --install "/usr/bin/javaws" "javaws" "/usr/lib/jvm/jdk1.7.0/bin/javaws"

e. Correct the file ownership and the permissions of the executables:

   sudochmoda+x /usr/bin/java

   sudochmoda+x /usr/bin/javac

   sudochmoda+x /usr/bin/javaws

   sudochown -R root:root /usr/lib/jvm/jdk1.7.0

f. Check the version of you new JDK 7 installation:

   java-version

11. Install SSH Server

    sudo apt-get install openssh-client

    sudo apt-get install openssh-server

12. Configure SSH

    su - hduser

    ssh-keygen -t rsa -P ""

```
cat $HOME/.ssh/id_rsa.pub >> $HOME/.ssh/authorized_keys

sshlocalhost
```

13. Disabling IPv6 – Run the following command in the extended terminal (Alt + F2)

```
gksudogedit /etc/sysctl.conf
```

14. Add the following lines to the bottom of the file

```
# disable ipv6
net.ipv6.conf.all.disable_ipv6 = 1
net.ipv6.conf.default.disable_ipv6 = 1
net.ipv6.conf.lo.disable_ipv6 = 1
```

15. Save the file and close it
16. Restart your Ubuntu
17. Download Apache Hadoop 1.1.2 and store it the Downloads folder
18. Unzip the file (open up the terminal window)

```
cd Downloads

sudo tar xzf hadoop-1.1.2.tar.gz

cd /usr/local

sudo mv /home/hduser/Downloads/hadoop-1.1.2 hadoop

sudoaddgrouphadoop

sudochown -R hduser:hadoophadoop
```

19. Open your .bashrc file in the extended terminal (Alt + F2)

```
gksudogedit .bashrc
```

20. Add the following lines to the bottom of the file:

```
# Set Hadoop-related environment variables
export HADOOP_HOME=/usr/local/hadoop
export PIG_HOME=/usr/local/pig
export PIG_CLASSPATH=/usr/local/hadoop/conf

# Set JAVA_HOME (we will also configure JAVA_HOME directly for Hadoop later on)
export JAVA_HOME=/usr/lib/jvm/jdk1.7.0/

# Some convenient aliases and functions for running Hadoop-related commands
unaliasfs&> /dev/null
alias fs="hadoopfs"
unaliashls&> /dev/null
alias hls="fs -ls"

# If you have LZO compression enabled in your Hadoop cluster and
# compress job outputs with LZOP (not covered in this tutorial):
# Conveniently inspect an LZOP compressed file from the command
# line; run via:
#
# $ lzohead /hdfs/path/to/lzop/compressed/file.lzo
#
# Requires installed 'lzop' command.
#
lzohead () {
hadoopfs -cat $1 | lzop -dc | head -1000 | less
}

# Add Hadoop bin/ directory to PATH
export PATH=$PATH:$HADOOP_HOME/bin
export PATH=$PATH:$PIG_HOME/bin
```

21. Save the .bashrcfile and close it
22. Run

    gksudogedit/usr/local/hadoop/conf/hadoop-env.sh

23. Add the following lines

```
# The java implementation to use.  Required.
export JAVA_HOME=/usr/lib/jvm/jdk1.7.0/
```

24. Save and close file
25. In the terminal window, create a directory and set the required ownerships and permissions

    sudomkdir -p /app/hadoop/tmp

    sudochownhduser:hadoop /app/hadoop/tmp

    sudochmod 750 /app/hadoop/tmp

26. Run

    gksudogedit /usr/local/hadoop/conf/core-site.xml

27. Add the following between the <configuration> … </configuration> tags

```
<property>
<name>hadoop.tmp.dir</name>
<value>/app/hadoop/tmp</value>
<description>A base for other temporary directories.</description>
</property>


<property>
<name>fs.default.name</name>
<value>hdfs://localhost:54310</value>
<description>The name of the default file system.  A URI whose
```

```
scheme and authority determine the FileSystem implementation.  The
uri's scheme determines the config property (fs.SCHEME.impl) naming
theFileSystem implementation class.  The uri's authority is used to
  determine the host, port, etc. for a filesystem.</description>
</property>
```

28. Save and close file
29. Run

```
gksudogedit /usr/local/hadoop/conf/mapred-site.xml
```

30. Add the following between the <configuration> … </configuration> tags

```
<property>
<name>mapred.job.tracker</name>
<value>localhost:54311</value>
<description>The host and port that the MapReduce job tracker runs
at.  If "local", then jobs are run in-process as a single map
and reduce task.
</description>
</property>
```

31. Save and close file
32. Run

```
gksudogedit /usr/local/hadoop/conf/hdfs-site.xml
```

33. Add the following between the <configuration> … </configuration> tags

```
<property>
<name>dfs.replication</name>
<value>1</value>
<description>Default block replication.
  The actual number of replications can be specified when the file is created.
  The default is used if replication is not specified in create time.
```

```
</description>
</property>
```

34. Format the HDFS

```
/usr/local/hadoop/bin/hadoopnamenode -format
```

35. Press the Start button and type Startup *Applications*
36. Add an application with the following command:
    /usr/local/hadoop/bin/start-all.sh
37. Restart Ubuntu and login

Run a Simple MapReduce Program

1. Get the Gettysburg Address and store it on your Downloads folder
2. Copy the Gettysburg Address from the name node to the HDFS

```
cd Downloads

hadoopfs -put gettysburg.txt /user/hduser/getty/gettysburg.txt
```

3. Check that the Gettysburg Address is in HDFS

```
hadoopfs -ls /user/hduser/getty/
```

4. Delete Gettysburg Address from your name node

```
rm gettysburg.txt
```

5. Download a jar file that contains a WordCount program into the Downloads folder
6. Execute the WordCount program on the Gettysburg Address (the following command is one line)

```
hadoop jar chiu-wordcount2.jar WordCount /user/hduser/getty/gettysburg.txt
/user/hduser/getty/out
```

7. Check MapReduce results

```
hadoopfs -cat /user/hduser/getty/out/part-r-00000
```

Appendix- B

Source Code

E-Medicare Website

- **Home Page**

```
<!Doctype html>
<head>
<title>Demo</title>
<link rel="stylesheet" href="css/ex.css" type="text/css">
<link rel="stylesheet" type="text/css" href="style.css" media="screen" />
<script src="js/dw_event.js" type="text/javascript"></script>
<script src="js/dw_rotator.js" type="text/javascript"></script>
<script type="text/javascript">
var rotator1 = {
   path:   'images/',
   speed:  3000, // default is 4500
   id:   'r1',
   images:["download.jpg", "images.jpg","images (11).jpg","images (2).jpg"]
}
var rotator2 = {
   path:   'images/',
   speed:  3000, // default is 4500
   id:   'r2',
   images:["images (3).jpg", "images (4).jpg","images (5).jpg","images
(6).jpg"]
}
var rotator3 = {
   path:   'images/',
   speed:  3000, // default is 4500
   id:   'r3',
   images:  ["images  (7).jpg",  "images  (8).jpg","images  (9).jpg","images
(10).jpg"]
}
function initRotator() {
   dw_Rotator.setup(rotator1);
   dw_Rotator.setup(rotator2);
   dw_Rotator.setup(rotator3);
}
dw_Event.add( window, 'load', initRotator);
</script>
<SCRIPT LANGUAGE="JavaScript">
var timerRunning = false;
var timezone = "Greenwich Mean Time";  // what time zone are you in ?
var adjust = 0;
function timeCheck(tzone, diff) {
if (timerRunning) {
clearTimeout(updatetime);
timerRunning = false; }
gmtOffset=eval(diff+adjust);
timezone = tzone;
checkDateTime();
}
function checkDateTime () {
var today = new Date();
var year = today.getYear() + 1900;
var month = today.getMonth()+1;
var date = today.getDate();
var day = today.getDay();
var hour = today.getHours();
var minute = today.getMinutes();
var second = today.getSeconds();
var lastSat = date - (day+1);
while (lastSat < 32) lastSat+=7;
```

```
if (lastSat > 31) lastSat+=-7;
var firstSat = date - (day+1);
while (firstSat > 0) firstSat+=-7;
if (firstSat < 1) firstSat+=7;
if ((((month == 4) && (date >= firstSat)) || month > 4) &&
(month < 11 || ((month == 10) && day <= lastSat))) adjust += 60;
yourOffset = (new Date()).getTimezoneOffset();
yourOffset = yourOffset + adjust;
var xx = navigator.appName
var xy = navigator.appVersion;
xy = xy.substring(0,1);
if ((xy == 4) && (xx == "Netscape")) yourOffset = yourOffset+adjust;
if ((((month == 4) && (date > 20)) || month > 4) && (month < 11 || ((month
== 10) &&
day < 30))) adjust -= 60;
ourDifference = eval(gmtOffset - yourOffset);
var half = eval(ourDifference % 60);
ourDifference = Math.round(ourDifference / 60);
hour = eval(hour - ourDifference);
var m = new Array("",
"Jan","Feb","Mar",
"Apr","May","Jun",
"Jul","Aug","Sept",
"Oct","Nov","Dec");
var leap = eval(year % 4);
if ((half == -30) || (half == 30)) minute += 30;
if (minute > 59) minute -= 60, hour++;
if (minute < 0) minute += 60, hour--;
if (hour > 23) hour -= 24, date += 1;
if (((month == 4) || (month == 6) ||
(month == 9) || (month == 11)) && (date==31)) date = 1, month ++;
if (((month == 2) && (date > 28)) && (leap != 0)) date = 1, month ++;
if ((month == 2) && (date > 29)) date = 1, month++;
if (hour < 0) hour += 24, date --;
if ((date == 32) && (month == 12)) month = m[1], date = 1, year++;
if (date == 32) date = 1, month++;
if ((date < 1) && (month == 1)) month= m[12], date = 31, year--;
if (date < 1) date = 31, month --;
if (((month == 4) || (month == 6) ||
(month== 9) || (month == 11)) && (date == 31)) date = 30;
if ((month == 2) && (date > 28)) date = 29;
if (((month == 2) && (date > 28)) && (leap != 0)) date=28;
for (i=1; i<13; i++) {
if (month == i) {
month = m[i]; break;
 }
}
var dateTime = hour;
dateTime = ((dateTime < 10) ? "0":"") + dateTime;
dateTime = "   " + dateTime;
dateTime += ((minute < 10) ? ":0" : ":") + minute;
dateTime += ((second < 10) ? ":0" : ":") + second;
dateTime += (hour >= 12) ? " PM, " : " AM, ";
dateTime += month + " " + date + ", " + year;
document.clock.zonetime.value = dateTime;
document.clock.zonename.value = timezone;
updatetime=setTimeout("checkDateTime()", 900);
timerRunning = true;
}
</script>
</head>
<body OnLoad="timeCheck(timezone, 0)">
<div id="main_container">
<div class="header">
<div id="logo"><img src="images/images (12).jpg" alt="" width="162"
height="80" border="0" /></div>
<div class="right_header">
```

```
<div class="top_menu"> <a href="login.html" class="login">login</a> <a
href="signup.php" class="sign_up">signup</a> </div>
<div id="menu">
<ul>
    <li><a class="current" href="#">Home</a></li>
    <li><a href="about.html">About Us</a></li>
    <li><a href="healthyliving.html">Healthy Living</a></li>
    <li><a href="contact.php">Review</a></li>
</ul>
</div>
</div>
</div>
<frameset cols="25%,50%,25%">
<frame>
<img id="r1" src="C:\Users\disha\Desktop\website\images\download.jpg"
border="0">
</frame>
<frame >
<img id="r2" src="C:\Users\disha\Desktop\website\images\images (3).jpg">
</frame>
<img id="r3" src="C:\Users\disha\Desktop\website\images\images (7).jpg">
</frame>
</frameset>
<frameset cols="25%,75%">
<frame>
</frame>
<frame >
</frame>
</frameset>
</div>
<divstyle="position:absolute;top:50%;left:15%;width:500px;height:200px;">
<br>
<br>
<h2><p>
```

Globus Medical is the National Institutes of Health's Web site for patients and their families and friends. Produced by the National Library of Medicine, it brings you information about diseases, conditions, and wellness issues in language you can understand. Globus Medical offers reliable, up-to-date health information, anytime, anywhere, for free.

You can use Globus Medical to learn about the latest treatments, look up information on a drug or supplement, find out the meanings of words, or view medical videos or illustrations. You can also get links to the latest medical research on your topic or find out about clinical trials on a disease or condition.

```
</p>
</h2>
</div>
<divstyle="position:absolute;top:50%;left:65%;width:280px;height:200px;">
<h2><t>            WORLD CLOCK</t></h2>
</div>
<divstyle="position:absolute;top:55%;left:60%;width:280px;height:200px;bo
rder: 2px solid;">
<table>
<tr><td>
</td>
<form name=clock>
<tr>
<td>
</td>
<td><center>
<input type=text name=zonetime size=31><br>
<center></td>
</tr>
<tr>
<td>
</td>
<td><b><center>Current Time Zone</center></b></td>
</tr>
```

```
<tr>
<td>
</td>
<td><center>
<input type=text name=zonename size=21>
</center></td>
</tr>
<br>
<table border=1 cellpadding=5>
<tr>
<td align=center><input type=button value="Pacific"
onClick="timeCheck(this.value, +480)"></td>
<td align=center><input type=button value="Central"
onClick="timeCheck(this.value, +420)"></td>
<td align=center><input type=button value="Eastern"
onClick="timeCheck(this.value, +300)"></td>
</tr>
<tr>
<td align=center><input type=button value="Hawaii"
onClick="timeCheck(this.value, +600)"></td>
<td align=center><input type=button value="Mexico"
onClick="timeCheck(this.value, +360)"></td>
<td align=center><input type=button value="New Delhi"
onClick="timeCheck(this.value, -330)"></td>
</tr>
<tr>
<td align=center><input type=button value="Hong Kong"
onClick="timeCheck(this.value, -480)"></td>
<td align=center><input type=button value="Tokyo"
onClick="timeCheck(this.value, -540)"></td>
<td align=center><input type=button value="London"
onClick="timeCheck(this.value, +0)"></td>
</tr>
</table>
```

```
 </div>
 </table>
 </form>
  </body>
  </html>
```

- **About me**

```
<!DOCTYPE html>
<html>
<head>
<link rel="stylesheet" href="css/ex.css" type="text/css">
<link rel="stylesheet" type="text/css" href="style.css" media="screen" />
</head>
<body>
<div id="main_container">
<div class="header">
<div id="logo"><img src="images/images (12).jpg" alt="" width="162"
height="80" border="0" /></div>
<div class="right_header">
<div class="top_menu"> <a href="login.html" class="login">login</a> <a
href="signup.php" class="sign_up">signup</a> </div>
<div id="menu">
<ul>
        <li><a class="current" href="a11.htm">Home</a></li>
        <li><a href="#">About Us</a></li>
        <li><a href="healthyliving.html">Healthy Living</a></li>
        <li><a href="contact.php">Review</a></li>
 </ul>
</div>
</div>
<img src="images\banner1.jpg" style="vertical-align:left" width="900"
height="200">
<h1> About Us</h1>
```

```
<h2><p>
Globus Medical is the National Institutes of Health's Web site for patients and their families and friends. Produced by the National Library of Medicine, it brings you information about diseases, conditions, and wellness issues in language you can understand. Globus Medical offers reliable, up-to-date health information, anytime, anywhere, for free.
You can use Globus Medical to learn about the latest treatments, look up information on a drug or supplement, find out the meanings of words, or view medical videos or illustrations. You can also get links to the latest medical research on your topic or find out about clinical trials on a disease or condition.
<br>
<br>
Globus Medical has had a highly accomplished, uniquely experienced team of qualified executives in the fields of medicine, healthcare, Internet technology, and business to bring you the most comprehensive, sought-after healthcare information anywhere.
</h2>
</p>
</div>
</div>
</body>
</html>
```

- **Healthy Living Website**

```
<!DOCTYPE html>
<html>
<head>
<link rel="stylesheet" href="css/ex.css" type="text/css">
<link rel="stylesheet" type="text/css" href="style.css" media="screen" />
</head>
<body>
<div id="main_container">
<div class="header">
```

```
 <div id="logo"><img src="images/images (12).jpg" alt="" width="162"
 height="80" border="0" /></div>
 <div class="right_header">
 <div class="top_menu"> <a href="login.html" class="login">login</a> <a
 href="signup.php" class="sign_up">signup</a> </div>
 <div id="menu">
 <ul>
        <li><a class="current" href="a11.htm">Home</a></li>
        <li><a href="about.html">About Us</a></li>
        <li><a href="#">Healthy Living</a></li>
        <li><a href="contact.php">Review</a></li>
</ul>
</div>
</div>
<img src="images\banner3.jpg" style="vertical-align:left" width="900"
height="200">
<div style="position:absolute;top:60%;left:17%;">
<img src="images\eathealthy.jpg" > </div>
<div style="position:absolute;top:55%;left:50%;width:900px ;height:200px;">
<h2 >Nutrition</h2>
</div>
<div style="position:absolute;top:60%;left:33%;width:900px ;height:200px;">
 To be healthy, your body needs to get enough vitamins, minerals, and other
 nutrients.<br> Eating healthy means getting plenty of:
<br>
 Vegetables, fruits, grains, and fat-free milk products
 Seafood, lean meat and poultry, eggs, beans, peas, seeds, and nuts
 <br>
 Eating healthy also means limiting:
 <br>Cholesterol, sodium (salt), and added sugars.
 <br>Trans fats – Trans fats may be in foods like cakes, cookies, stick
 margarines, and fried foods.
 <br>Saturated fats – These fats come from animal products like cheese, fatty
 meats, whole milk, and butter.
```

```
<br>Refined grains – Food products with refined grains include white bread,
noodles, white rice, and flour tortillas.
</div>
<div style="position:absolute;top:90%;left:17%;">
<img src="images\getactive.jpg" > </div>
<divstyle="position:absolute;top:85%;left:50%;width:900px;height:200px;">
<h2 >Physical Activity</h2>
</div>
<divstyle="position:absolute;top:90%;left:33%;width:900px;height:200px;">
Physical activity is anything that gets your body moving. Start at a
comfortable level. Once you get the hang of it, add a little<br> more activity
each time. Then try getting active more often.
<br>
What kinds of activity should I do?<br>
To get the health benefits of physical activity, do a combination of aerobic
and muscle-strengthening activities.
<br>
Aerobic activities make you breathe harder and cause your heart to beat
faster. Walking fast is an example of aerobic activity.
<br>Muscle-strengthening activities make your muscles stronger. Muscle-
strengthening activities include lifting weights and using <br>resistance
bands.
</div>
<div style="position:absolute;top:125%;left:17%;">
<img width="210" height="110" src="images\family.jpg" > </div>
<divstyle="position:absolute;top:115%;left:50%;width:900px;height:200px;>
<h2 >Mental Health</h2>
</div>
<divstyle="position:absolute;top:120%;left:33%;width:900px;height:200px;>
The activities you engage in and the daily choices you make affect the way
you feel physically and emotionally.
<br>
Get enough rest-To have good mental and emotional health, it’s important to
take care of your body. That includes getting enough<br> sleep. Most people
```

```
need seven to eight hours of sleep each night in order to function
optimally.<br>
Learn about good nutrition and practice it- The subject of nutrition is
complicated and not always easy to put into practice. But the<br> more you
learn about what you eat and how it affects your energy and mood, the better
you can feel.<br>
Exercise to relieve stress and lift your mood- Exercise is a powerful antidote
to stress, anxiety, and depression. Look for small ways<br> to add activity to
your day, like taking the stairs instead of the elevator or going on a short
walk. To get the most mental health benefits,<br> aim for 30 minutes or more
of exercise per day.<br>
Get a dose of sunlight every day-Sunlight lifts your mood, so try to get at
least 10 to 15 minutes of sun per day. This can be done while<br> exercising,
gardening, or socializing.<br>
Limit alcohol and avoid cigarettes and other drugs. These are stimulants that
may unnaturally make you feel good in the short term,<br> but have long-
term negative consequences for mood and emotional health.
</div>
<div style="position:absolute;top:175%;left:17%;">
<img width="210" height="110" src="images\download (1).jpg" > </div>
<divstyle="position:absolute;top:160%;left:53%;width:900px;height:200px;>
<h2 >Safety</h2>
</div>
<divstyle="position:absolute;top:165%;left:33%;width:900px;height:200px;>
These 10 drug DOs and DON'Ts can help you make sure that your
medication works safely to improve your health.
<br>
5 Drug DOs...
<br>
DO take each medication exactly as it has been prescribed.
<br>DO make sure that all your doctors know about all your medications.
<br>DO let your doctors know about any other over-the-counter medications,
vitamins and supplements, or herbs that you use.
```

```
<br>DO try to use the same pharmacy to fill all your prescriptions, so that they can help you keep track of everything you're taking.
<br>DO keep medications out of the reach of children.
<br>5 Drug DON'Ts...

<br>DON'T change your medication dose or schedule without talking with your doctor.
<br>DON'T use medication prescribed for someone else.
<br>DON'T crush or break pills unless your doctor instructs you to do so.
<br>DON'T use medication that has passed its expiration date.
<br>DON'T store your medications in locations that are either too hot or too cold. For example, the bathroom cabinet may not be the best place for your medication
</div>
</div>
</body>
</html>
```

- **Review**

```
<html>
<head>
<link rel="stylesheet" href="css/ex.css" type="text/css">
<link rel="stylesheet" type="text/css" href="style.css" media="screen" />
</head>
<body>
<div id="main_container">
<div class="header">
<div id="logo"><img src="images/images (12).jpg" alt="" width="162" height="80" border="0" /></div>
<div class="right_header">
<div class="top_menu"> <a href="login.html" class="login">login</a> <a href="signup.php" class="sign_up">signup</a> </div>
<div id="menu">
```

```
  <ul>
        <li><a class="current" href="a11.htm">Home</a></li>
        <li><a href="about.html">About Us</a></li>
        <li><a href="healthyliving.html">Healthy Living</a></li>
        <li><a href="#">Review</a></li>
  </ul>
  </div>
  </div>
  <img src="images\home_page_banner_2.png" style="vertical-align:left"
width="900" height="200">
<form action="contactdetails.php" method="post">
<fieldset>
<legend><b><h3>Review</h3></b></legend>
<table cellpadding="3">
<tr><td><strong>Subject Of Review</strong></td><td><input type="text"
name="sub"></td></tr>
<tr><td></td>
</tr>
<tr><td width="150px"><strong>
Review:
</td>
<td>
<textarea cols="50" rows="5" name="comm">
</textarea>
</strong>
</td>
</tr>
 </table>
 </fieldset>
<fieldset>
<legend><b><h3>Additional Information</h3></b></legend>
<table cellpadding="3">
<tr><td><strong>E-mail Id</strong></td><td><input type="email"
name="data1"></td></tr>
```

```
<tr><td><strong>First Name</strong></td><td><input type="text" name="data2"></td></tr>
<tr><td><strong>Last Name</strong></td><td><input type="text" name="data3"></td></tr>
<tr><td><strong>State</strong></td><td><input type="text" name="data4"></td></tr>
<tr><td width="150px"><strong>Country</strong></td><td><input type="text" name="data5"></td></tr>
</table>
</fieldset>
<?php
if(array_key_exists('cumulativeErrorMessage', $_POST) && $_POST['cumulativeErrorMessage'] != '') {
?>
<fieldset style="color: #ff0000;">
<legend>There were errors</legend>
 <?php echo $_POST['cumulativeErrorMessage']?>
 </fieldset>
 <?php
}
?>
<br/>
<input type="submit" value="Submit">
</form>
</div>
</div>
</body>
</html>
```

- **Login Page**

```
<!DOCTYPE html>
<html>
<head>
```

```
<link rel="stylesheet" href="css/ex.css" type="text/css">
<link rel="stylesheet" type="text/css" href="style.css" media="screen" />
<script type="text/javascript" src="altTextBox.js"></script>
</head>
<body>
<div id="main_container">
<div class="header">
<div id="logo"><img src="images/images (12).jpg" alt="" width="162"
height="80" border="0" /></div>
<div class="right_header">
<div class="top_menu">  <a href="signup.php" class="sign_up">signup</a>
</div>
<div id="menu">
<ul>
        <li><a class="current" href="a11.htm">Home</a></li>
        <li><a href="about.html">About Us</a></li>
        <li><a href="healthyliving.html">Healthy Living</a></li>
        <li><a href="contact.php">Review</a></li>
 </ul>
 </div>
 </div>
 <table>
 <col width="500">
 <col width="500">
 <tr>
 <th colspan="2"></th>
<img src="images\banner_3.jpg" style="vertical-align:left" width="900"
height="200">
</tr>
<tr>
<td>
<img src="images\up.png" style="vertical-align:middle" width="150"
height="50"/>
</td>
```

```
<td>
</td>
</tr>
<tr>
<td>
<form name="change">
<textarea rows="10" cols="20" class="test" name="descript" wrap="virtual">
</textarea>
</form>
</td>
<td>
<img src="images\sign.png" style="vertical-align:middle" width="150" height="50"/>
<div id="login_form">
<form name="myform" method="post" action="logincheck.php" id="myform">
<table>
<tr>
<td class="f1_label">User Name :</td><td><input id="username" type="text" name="username" value="" />
</td>
</tr>
<tr>
<td class="f1_label">Password :</td><td><input id="password" type="password" name="password" value=""  />
</td>
</tr>
<tr>
<td>
<input type="submit" name="submit" value="Log In" style="font-size:18px;/>
</td>
</tr>
```

```
</table>
</form>
 </div>
 </td>
 </tr>
 </table>
 </div>
 </div>
 </body>
 </html>
```

- **Signup Page**

```
<!DOCTYPE html>
<html>
<head>
<link rel="stylesheet" href="css/ex.css" type="text/css">
<link rel="stylesheet" type="text/css" href="style.css" media="screen" />
<script type="text/javascript" src="altTextBox.js"></script>
</head>
<body>

<div id="main_container">
<div class="header">
<div id="logo"><img src="images/images (12).jpg" alt="" width="162"
height="80" border="0" /></div>
<div class="right_header">
<div class="top_menu">  <div class="top_menu"> <a href="login.html"
class="login">login</a></div>
<div id="menu">
<ul>
        <li><a class="current" href="a11.htm">Home</a></li>
        <li><a href="about.html">About Us</a></li>
        <li><a href="healthyliving.html">Healthy Living</a></li>
        <li><a href="contact.php">Review</a></li>
</ul>
</div>
</div>
</div>
<img  src="images\images (13).jpg"  width="900" height="200">
<form action="signupdetails.php" method="post">
<fieldset>
<legend><b><h3>Personal Information</h3></b></legend>
 <table cellpadding="3">
 <tr><tdwidth="150px"><strong>Username
(Email)</strong></td><td><input                    type="email"
name="username"></td></tr>
<tr><td><strong>First     name</strong></td><td><input     type="text"
name="firstname"></td></tr>
<tr><td><strong>Last     name</strong></td><td><input     type="text"
name="lastname"></td></tr>
 <tr><td><strong>Password</strong></td><td><input     type="password"
name="password"></td></tr>
 </table>
 </fieldset>
  <fieldset>
    <legend><b><h3>Additional Information</h3></b></legend>
    <table cellpadding="3">
   <tr><td><strong>Street</strong></td><td><input          type="text"
name="data1"></td></tr>
   <tr><td><strong>City</strong></td><td><input            type="text"
name="data2"></td></tr>
   <tr><td><strong>State</strong></td><td><input           type="text"
name="data3"></td></tr>
   <tr><td><strong>Country</strong></td><td><input         type="text"
name="data4"></td></tr>
  <tr><td   width="150px"><strong>Mobile   No.</strong></td><td><input
type="number" name="data5"></td></tr>
```

```html
</table>
</fieldset>
<fieldset style="text-align: center">
<legend><h3><b>Terms & conditions</b></h3></legend>
 <textarea cols="50" rows="5">Terms and conditions for website usage
Welcome to our website. If you continue to browse and use this website, you
are agreeing to comply with and be bound by the following terms and
conditions of use, which together with our privacy policy govern [business
name]'s relationship with you in relation to this website. If you disagree with
any part of these terms and conditions, please do not use our website. The
term '[business name]' or 'us' or 'we' refers to the owner of the website whose
registered office is [address]. Our company registration number is [company
registration number and place of registration]. The term 'you' refers to the
user or viewer of our website. The use of this website is subject to the
following terms of use: The content of the pages of this website is for your
general information and use only. It is subject to change without notice.
This website uses cookies to monitor browsing preferences. If you do allow
cookies to be used, the following personal information may be stored by us
for use by third parties: [insert list of information].
Neither we nor any third parties provide any warranty or guarantee as to the
accuracy, timeliness, performance, completeness or suitability of the
information and materials found or offered on this website for any particular
purpose. You acknowledge that such information and materials may contain
inaccuracies or errors and we expressly exclude liability for any such
inaccuracies or errors to the fullest extent permitted by law.
Your use of any information or materials on this website is entirely at your
own risk, for which we shall not be liable. It shall be your own responsibility
to ensure that any products, services or information available through this
website meet your specific requirements.
This website contains material which is owned by or licensed to us. This
material includes, but is not limited to, the design, layout, look, appearance
and graphics. Reproduction is prohibited other than in accordance with the
copyright notice, which forms part of these terms and conditions.
```

```
All trade marks reproduced in this website which are not the property of, or
licensed to, the operator are acknowledged on the website.
Unauthorised use of this website may give rise to a claim for damages and/or
be a criminal offence.
From time to time this website may also include links to other websites.
These links are provided for your convenience to provide further information.
They do not signify that we endorse the website(s). We have no responsibility
for the content of the linked website(s).
Your use of this website and any dispute arising out of such use of the
website is subject to the laws of England, Northern Ireland, Scotland and
Wales.</textarea>
 <br/>
I confirm that I agree with terms & conditions <input type="checkbox"
name="agreeWithTerms" value="Y">
<br/><br/>
<?php
if(array_key_exists('cumulativeErrorMessage', $_POST) &&
$_POST['cumulativeErrorMessage'] != '') {
?>
<fieldset style="color: #ff0000;">
<legend>There were errors</legend>
<?php echo $_POST['cumulativeErrorMessage']?>
</fieldset>
 <?php
}
?>
<br/>
<input type="submit" value="add">
</form>
</div>
</body>
</html>
```

- **Search Page**

```
<html>
<head>
<link rel="stylesheet" href="css/ex.css" type="text/css">
<link rel="stylesheet" type="text/css" href="style.css" media="screen" />
</head>
<body>
<div id="main_container">
<div class="header">
<div   id="logo"><img   src="images/images   (12).jpg"   alt=""   width="162"
height="80" border="0" /></div>
<div class="right_header">
<div class="top_menu"><h3> WELCOME
<?php
session_start();
$myusername = $_SESSION['myusername'];
echo$myusername;
?>
</h3> </div>
<div id="menu">
 <ul>
          <li><a class="current" href="#">Home</a></li>
          <li><a href="about.html">About Us</a></li>
          <li><a healthyliving.html">Healthy Living</a></li>
          <li><a href="contact.php">Review</a></li>
 </ul>
 </div>
 </div>
 </div>
<img      src="images\banner-1.jpg"    style="vertical-align:left"    width="900"
height="200">
</div>
```

```
</div>
<br>
<center><h1>Enter Query</h1></center>
<div>
<form id="tfnewsearch" method="post" action="file.php">
<input type="text" class="tftextinput" name="q" size="21"
maxlength="120"><input type="submit" value="search" class="tfbutton">
</form>
<div class="tfclear"></div>
</div>
</body>
</html>
```

- **MapReduce Code**

```
import java.io.*;

import java.util.StringTokenizer;

import org.apache.hadoop.conf.Configuration;

import org.apache.hadoop.fs.Path;

import org.apache.hadoop.io.IntWritable;

import org.apache.hadoop.io.LongWritable;

import org.apache.hadoop.io.Text;

import org.apache.hadoop.mapreduce.Job;

import org.apache.hadoop.mapreduce.Mapper;

import org.apache.hadoop.mapreduce.Reducer;

import org.apache.hadoop.mapreduce.Reducer.Context;

import org.apache.hadoop.mapreduce.lib.input.FileInputFormat;

import org.apache.hadoop.mapreduce.lib.input.TextInputFormat;
```

```
        import org.apache.hadoop.mapreduce.lib.output.FileOutputFormat;

        import org.apache.hadoop.mapreduce.lib.output.TextOutputFormat;

        public class WordCount

        {

            private static final Text ttwo = new Text();

            private static final IntWritable two = new IntWritable();

            public static class Map extends Mapper<LongWritable, Text, Text,
            IntWritable>

            {

                private static final IntWritable one = new IntWritable(1);

                private Text word = new Text();

               public void map(LongWritable key, Text value, Mapper<LongWritable,
Text, Text, IntWritable>.Context context throws IOException, InterruptedException

            {

            RandomAccessFile br1,br;

            String line = value.toString();

            line = line.toLowerCase();

            line = line.replaceAll("'", "");

            line = line.replaceAll("[^a-zA-Z]", " ");

            try {

                        String sCurrentLine,save,dd,s,fin,checked;

            br =new RandomAccessFile("/home/hduser/KEY/key_derrived.txt","rw");

br1 = new RandomAccessFile("/home/hduser/Java_Applications/name.txt", "rw");
```

```
dd =new String("disease");
fin=new String(" ");
int len=0,len1=0,count=0;
save = new String(" ");
checked= new String(" ");
StringTokenizer tokenizer = new StringTokenizer(line);
while (tokenizer.hasMoreTokens())
{
        String token = tokenizer.nextToken();
        if(dd.equals(token))
        {
        if (tokenizer.hasMoreTokens())
        {
                count=0;
                save = tokenizer.nextToken();
                len=save.length();
                checked=" ";
        }
        }
        while ((sCurrentLine = br.readLine()) != null)
        {
        if(sCurrentLine.equals(token) && checked.contains(token)==false)
        {
```

```
{               count++;

                checked=checked+token;

          }

          }

          br.seek(0);

          if(count>=3)

          {

          while ((s = br1.readLine()) != null)

          {

          if(s.contains(save))

          {

                len1=s.length();

                word.set(s.substring(len+1));

                context.write(this.word, two);

                break;

          }

          }

          }

          br1.seek(0);

          }

}

catch (IOException e)

{
```

```
            e.printStackTrace();
}
}
}
  public static class Reduce extends Reducer<Text, IntWritable, Text, IntWritable>
  {
 public void reduce(Text key, Iterable<IntWritable> values, Reducer<Text,
IntWritable, Text, IntWritable>.Context context) throws IOException,
InterruptedException
    {
            context.write(key, null);
    }
  }
  public static void main(String[] args) throws Exception
  {
            Configuration conf = new Configuration();
            Job job = new Job(conf, "WordCount");
            job.setJarByClass(WordCount.class);
            job.setOutputKeyClass(Text.class);
            job.setOutputValueClass(IntWritable.class);
            job.setMapperClass(WordCount.Map.class);
            job.setReducerClass(WordCount.Reduce.class);
            job.setInputFormatClass(TextInputFormat.class);
```

```
        job.setOutputFormatClass(TextOutputFormat.class);

        FileInputFormat.addInputPath(job, new Path(args[0]));

        FileOutputFormat.setOutputPath(job, new Path(args[1]));

        job.waitForCompletion(true);

    }

    }
```