

BLOOD BANK MANAGEMENT SYSTEM

Mini Project Report

Submitted in partial fulfillment for the award of the degree of

BACHELOR OF TECHNOLOGY

in

ARTIFICIAL INTELLIGENCE AND DATA SCIENCE

SHIVAKUMAR

20L31A5443

SRI BINDHU

20L31A5462

HARITHA

20L31A5442

DEVI SRI PRASAD

20L31A5445

Under the guidance of

MS B.SUNAYANA

Assistant Professor



DEPARTMENT OF ARTIFICIAL INTELLIGENCE AND DATA SCIENCE

VIGNAN'S INSTITUTE OF INFORMATION TECHNOLOGY

(Autonomous) Affiliated to JNTU Kakinada & Approved by AICTE, New Delhi

Re-Accredited by NBA & NAAC (CGPA of 3.41/ 4.00)

ISO 9001:2008, ISO 14001:2004, OHSAS 18001:2007 Certified Institution

VISAKHAPATNAM – 530 039

2020-2024

VIGNAN'S INSTITUTE OF INFORMATION TECHNOLOGY
Department of Computer Science and Engineering



CERTIFICATE

This is to certify that the project report entitled "BLOOD BANK MANAGEMENT SYSTEM " is the Bonafide record of project work carried out under my supervision by **M SHIVAKUMAR (20L31A5443), SRI BINDHU (20L31A5462), HARITHA (20L31A5442), DEVI SRI PRASAD (20L31A5445)** during the academic year 2021-2022, in partial fulfillment of the requirements for the award of the degree of Bachelor of Technology in A of Jawaharlal Nehru Technological University, Kakinada. The results embodied in this project report have not been submitted to any other University or Institute for the award of any Degree or Diploma.

Head of the Department

Dr. Madhusudan Rao

Signature of Project Guide

B.SUNANYA MAM

Assistant Professor

i

DECLARATION

We hereby declare that the project report entitled “**BLOOD BANK MANAGEMENT SYSTEM**” has been written by us and has not been submitted either in part or whole for the award of any degree, diploma or any other similar title to this or any other university.

SHIVA KUMAR	20L31A5443
SRI BINDHU	20L31A5462
HARITHA	20L31A5442
DEVI SRI PRASAD	20L31A5445

Date:

Place:

ii

ACKNOWLEDGEMENT

It gives us a great sense of pleasure to acknowledge the assistance and cooperation we have received from several persons while undertaking this B. Tech. Final Year Project. We owe special debt of gratitude to **MS. B.SUNAYANA** Assistant Professor Department of COMPUTER SCIENCE AND ENGINEERING, for his constant support and guidance throughout the course of our work. His sincerity, thoroughness and perseverance have been a constant source of inspiration for us.

We also take the opportunity to acknowledge the contribution of Prof. Dr. **Dr. Madhusudan Rao**, Head, Department of artificial intelligence and data science, for his full support and assistance during the development of the project.

We also do not like to miss the opportunity to acknowledge the contribution of all faculty members of the department for their kind assistance and cooperation during the development of our project. Last but not the least, we acknowledge our friends for their contribution in the completion of the project.

ABSTRACT

The objective of this paper is that the management of the blood and detail of the donors. Online blood bank system” is a web application that allows to access the whole information about blood bank management system, readily scalable and adaptable to meet the complex need of blood bank who is the key facilitator for the health care sector it also supports all the functionalities of blood bank. It is used for maintaining information about the Donor. The project includes main modules admin, donor, blood request, blood bank and helpdesk. It maintains all the information of donor and all the record of blood requests and the available blood. The Blood Bank system helps the people who are in need of a blood by giving them all details of blood group availability or regarding the donors

CONTENTS

TITLE	PAGE NO
Declaration	i
Acknowledgement	ii
Abstract	iii
List of Figures	iv

TITLES	PAGE NO
1.i	
1.1 Introduction	
1.2 Background work	
1.3 Problem Statement	
1.4 Definition of Terms	
1.5 Why Grievance is needed	
1.6 What is Online Grievance redressal system	
1.7 Need of Grievance System	
1.8 Need of Online Student Grievance Management	
2.TECHNOLOGIES USED	
2.1 Web development	
2.1.1 Web Applications	
2.2 Front end development	
2.2.1 Languages of front end development	

2.3 Hyper Text Markup Language

2.3.1 How does it work?

2.3.2 What are the tags up-to?

2.3.3 Is there anything HTML can't do?

2.4 Cascading Style Sheet(CSS)

2.4.1 Advantages of CSS

2.4.2 Who creates and maintains CSS

2.4.3 CSS versions

2.4.4 Rules of CSS

2.5 JavaScript

2.5.1 Ease of Use

2.5.2 Popularity

2.5.3 Productivity

2.5.4 JavaScript as both Front-end and Back-end

2.6 Back end development

2.6.1 XAMPP server

2.6.1.1 Creation of first database

2.6.2 MySQL

2.6.3 Database

2.6.4 PHP

2.6.4.1 Why to learn PHP?

2.6.4.2 Characteristics of PHP

2.6.4.3 Hello World using PHP

2.6.4.4 Applications of PHP

3.LITERATURE REVIEW

4.PROPOSED WORK

4.1 Introduction

4.2 Users/Modules

4.2.1 Admin

4.2.2 User(Student)

4.2.3 Head of Department

4.3 Functional Requirements

4.3.1 Registration

4.3.2 Login

4.3.3 Complaint

4.3.4 Status/Feedback

4.3.5 History

4.4 Non- Functional Requirements

4.4.1 Performance Requirements

4.4.2 User Characteristics

5.IMPLEMENTATION

5.1 Introduction

5.2 Code Snippets

5.2.1 Logo.php

5.2.2 Login.php

5.2.3 Index.php

5.2.4 Feedback.php

5.2.5 Database.php

5.2.6 Admin.php

5.2.7 Admin-view.php

5.2.8 Staff.php

5.2.9 Users.php

5.2.10 Complaints.php

5.2.11 Test.php

5.3 Results/Screenshots

5.3.1 Home Page

5.3.2 Staff Login

5.3.3 Admin Complaints section

5.3.4 Creating new account by Admin

5.3.5 Assign Faculties based on complaints

5.3.6 HOD Approval

5.3.7 Complaint History

5.3.8 Feedback view

5.3.9 Use case Diagram

6.SYSTEM ANALYSIS

6.1 Problem Statement

6.1.1 Existing System

6.1.1.1 Disadvantages of Existing System

6.1.2 Proposed System

6.1.2.1 Advantages

6.2 Objectives

6.3 Feasibility Study

6.3.1 Technical Feasibility

6.3.2 Economic Feasibility

6.3.3 Legal Feasibility

6.3.4 Schedule Feasibility

7.REQUIREMENT AND SPECIFICATIONS

7.1 System Specifications

7.1.1 Software Requirement

7.1.2 Hardware Requirements

7.2 Developmental Tools

7.3 Features

7.3.1 Realibility

7.3.2 Availability

7.3.3 Performance

7.3.4 Technologies

8.CONCLUSION

8.1 Conclusion

8.2 Future Scope68

9.REFERENCES

References

CHAPTER-1

INTRODUCTION

1.INTRODUCTION

The Blood Bank Management system is great project. This project is designed for successful completion of project on blood bank management system. The basic building aim is to provide blood donation service to the city recently. Blood Bank Management System (BBMS) is a browser-based system that is designed to store, process, retrieve and analyze information concerned with the administrative and inventory management within a blood bank. Bank blood donation system is planned to collect blood from many donors in short from various sources and distribute that blood to needy people who require blood. To do all this we require high quality software to manage those jobs. The government spending lot of money to develop high quality “Blood Bank management system project”. Sometimes Doctors and Blood bank project have to face the difficulty in finding the blood group Donors at right time. At any point of time the people who are in need can reach the donors through our search facility. By mobilizing people and organization who desire to make a difference in the lives of people in need. On the basis of humanity, everyone is welcome to register as a blood donor. As we know this project is quite difficult, less efficient and accurate in comparison to the computerized system. The main objectives of this project are to automate the complete operations of the blood bank. They need maintain hundreds of thousands of records. Also searching should be very faster so they can find required details instantly. To develop a web-based portal to facilitate the co-ordination between supply and demand of blood. This system makes conveniently available good quality, safe blood and other blood components, which can be provided in a sound, ethical and acceptable manner, consistent with the long-term well-being of the community. It actively encourage voluntary blood donation, motivate and maintain a well-indexed record of blood donors and educate the community on the benefits of blood donation. This will also serve as the site for interaction of best practices in reducing unnecessary utilization of blood and help the state work more efficiently towards selfsufficiency in blood. The system will provide the user the option to look at the details of the existing Donor List, Blood Group and to add a new Donor. It also allows the user to modify the record. The administrator can alter all the system data. Lot of paper work, slow data processing, not user-friendly environment, difficult to keep old records and can't update daily, due to these problems it is necessary to develop a system that address all these issues.

1.2 Problem Statement

Bank blood donation system is planned to collect blood from many donors in short from various sources and distribute that blood to needy people who require blood. To do all this we require quality software to manage those jobs [4]. The system will provide the user the option to look at the details of the existing Donor List, Blood Group and to add a new Donor. It also allows the user to modify the record. The administrator can alter all the system data [5]. Online Blood Bank system is now a day's most widely used and popular because this is less time consuming and request for the blood from any of the area. This can also save the time [6]. Online Blood Bank system provides the accuracy factor because all the data is logged into the database safely and the user is provided with the receipt number for his data. The data is then sending safely to the administration and they may interact with the client as per his requirement [7]. This web application is also very helpful because a helpdesk

1.3 Need of blood bank management system

Blood banks play an important role in the process of collecting blood and managing blood stocks, approving blood requests, updating donations and updating available blood types. Raster's web-based BBMS will address the issues and problems encountered in collecting information about donors, blood camps, inventories of blood bags, and blood transfusion services, etc, including donor screening, inventory management, blood ordering, blood usage review and compatibility testing. Blood Bank Management system will greatly increase the safety and quality of the blood supply as well as provide logistics data for the optimal supply chain management.

CHAPTER – 2
TECHNOLOGIES USED

2.1 Web Development

Web development broadly refers to the tasks associated with developing websites for hosting via intranet or internet. The web development process includes web design, web content development, client-side/server-side scripting and network security configuration, among other tasks.

Web development is also known as website development.

Web development is the coding or programming that enables website functionality, per the owner's requirements. It mainly deals with the non-design aspect of building websites, which includes coding and writing markup.

Web development ranges from creating plain text pages to complex web-based applications, social network applications and electronic business applications.

The web development hierarchy is as follows:

- Client-side coding
- Server-side coding
- Database technology

2.1.1 Web Applications

Web application development is the creation of application program that reside on remote servers and are delivered to the user's device over the Internet. A web application (web app) does not need to be downloaded and is instead accessed through a network. An end user can access a web application through a web browser such as Google Chrome, Safari, or Mozilla Firefox. A majority of web applications can be written in Javascript, Cascading Style Sheets (CSS), and HTML.

Web application development will typically have a short development life-cycle lead by a small development team. **Front-end** development for web applications is accomplished through client-side programming. Client refers to a computer application such as a web browser. **Client-side** programming will typically utilize HTML, CSS and JavaScript. HTML programming will instruct a browser how to display the on-screen content of web pages, while CSS keeps displayed information in the correct format. JavaScript will run JavaScript code on a web page, making some of the content interactive.

Server-side programming powers the client-side programming and is used to create the scripts that web applications use. Scripts can be written in multiple scripting languages such as Ruby, Java and Python. Server-side scripting will create a custom interface for the end-user and will hide the source code that makes up the interface. A database such as MySQL or MongoDB can be used to store data in web application development.

2.2 Front-end development

The part of a website that user interacts with directly is termed as front end. It is also referred to as the ‘client side’ of the application. It includes everything that users experience directly: text colors and styles, images, graphs and tables, buttons, colors, and navigation menu. **HTML, CSS, and JavaScript** are the languages used for Front End development. The structure, design, behavior, and content of everything seen on browser screen when websites, web applications, or mobile apps are opened up, is implemented by front End developers. Responsiveness and performance are two main objectives of the front End. The developer must ensure that the site is responsive i.e. it appears correctly on devices of all sizes no part of the website should behave abnormally irrespective of the size of the screen.

2.1.2 Front end Languages

The front end portion is built by using some languages which are discussed below:

- HTML
- CSS
- JavaScript

2.3 Hyper Text Markup Language

2.3.1 How does it work ?

2.3.2 What are the tags are up to?

The tags are what separate normal text from HTML code. You might know them as the words between the <angle-brackets>. They allow all the cool stuff like images and tables and stuff, just by telling your browser what to render on the page. Different tags will perform different functions. The tags themselves don’t appear when you view your page through a browser, but their effects do. The simplest tags do nothing more than apply formatting to some text, like this:

****These words will be bold****, and these will not.

2.3.3 Is there anything HTML can't do ?

Of course, but since making websites became more popular and needs increased many other supporting languages have been created to allow new stuff to happen, plus HTML is modified every few years to make way for improvements.

Cascading Stylesheets are used to control how your pages are presented, and make pages more accessible. Basic special effects and interaction is provided by **JavaScript**, which adds a lot of power to basic HTML. Most of this advanced stuff is for later down the road, but when using all of these technologies together, you have a lot of power at your disposal.

2.4 Cascading Style Sheets(CSS)

Cascading Style Sheets, fondly referred to as CSS, is a simple design language intended to simplify the process of making web pages presentable.

CSS handles the look and feel part of a web page. Using CSS, you can control the color of the text, the style of fonts, the spacing between paragraphs, how columns are sized and laid out, what background images or colors are used, layout designs, variations in display for different devices and screen sizes as well as a variety of other effects.

CSS is easy to learn and understand but it provides powerful control over the presentation of an HTML document. Most commonly, CSS is combined with the markup languages HTML or XHTML.

2.4.1 Advantages of CSS

- **CSS saves time** – You can write CSS once and then reuse same sheet in multiple HTML pages. You can define a style for each HTML element and apply it to as many Web pages as you want.
- **Pages load faster** – If you are using CSS, you do not need to write HTML tag attributes every time. Just write one CSS rule of a tag and apply it to all the occurrences of that tag. So less code means faster download times.
- **Easy maintenance** – To make a global change, simply change the style, and all elements in all the web pages will be updated automatically.
- **Superior styles to HTML** – CSS has a much wider array of attributes than HTML, so you can give a far better look to your HTML page in comparison to HTML attributes.

- **Multiple Device Compatibility** – Style sheets allow content to be optimized for more than one type of device. By using the same HTML document, different versions of a website can be presented for handheld devices such as PDAs and cell phones or for printing.
- **Global web standards** – Now HTML attributes are being deprecated and it is being recommended to use CSS. So it's a good idea to start using CSS in all the HTML pages to make them compatible to future browsers.

2.4.2 who Creates and Maintains CSS?

CSS is created and maintained through a group of people within the W3C called the CSS Working Group. The CSS Working Group creates documents called specifications. When a specification has been discussed and officially ratified by the W3C members, it becomes a recommendation.

Vignan's Institute of Information Technology (2017-2020)

page | 10

These ratified specifications are called recommendations because the W3C has no control over the actual implementation of the language. Independent companies and organizations create that software.

NOTE – The World Wide Web Consortium, or W3C is a group that makes recommendations about how the Internet works and how it should evolve.

2.4.3 CSS Versions

Cascading Style Sheets level 1 (CSS1) came out of W3C as a recommendation in December 1996. This version describes the CSS language as well as a simple visual formatting model for all the HTML tags.

CSS2 became a W3C recommendation in May 1998 and builds on CSS1. This version adds support for media-specific style sheets e.g. printers and aural devices, downloadable fonts, element positioning and tables.

2.4.4 Rules of CSS

A CSS comprises of style rules that are interpreted by the browser and then applied to the corresponding elements in your document. A style rule is made of three parts –

- **Selector** – A selector is an HTML tag at which a style will be applied. This could be any tag like <h1> or <table> etc.
- **Property** – A property is a type of attribute of HTML tag. Put simply, all the HTML attributes are converted into CSS properties. They could be *color*, *border* etc.
- **Value** – Values are assigned to properties. For example, *color* property can have value either red or #F1F1F1 etc.

You can put CSS Style Rule Syntax as follows :

selector { property: value }

Fig -1 CSS syntax

2.5 JavaScript

JavaScript reached the height of its popularity when it entered the fields of cross-platform mobile application development and server-side development. Since Node.js framework can be used for both frontend and backend development; JavaScript became highly popular among web developers. With the rise of React Native by Facebook, also mobile app developers started to move towards JavaScript. As a result, JavaScript remains the most popular programming language to this day.

2.5.1 Ease of Use

JavaScript has been around longer in the industry and is a mature and stable language. JavaScript is very easy to use. It has numerous frameworks and libraries available online, so developers can use the existing code for developing apps faster. However, in order to learn the JavaScript syntax, we need to have a basic technical knowledge of programming in general.

Dart is a fairly new language for most of the developers outside of Google. Although Google has put a lot of effort into documenting the Dart programming language, it's still hard for developers to find solutions for specific problems. In terms of coding style and syntax, it has Java-like syntax, so developers from OOPS background can master and use Dart easily once they learn the basics.

2.5.2 Popularity

At the moment, JavaScript is everywhere. There is no device in the world that doesn't run JavaScript. There are many companies that are using JavaScript frameworks for developing web and mobile applications. JavaScript can also be used for writing server-side applications and backends, so more and more developers have got hooked on JavaScript as a language during their career.

Before Google announced Flutter, Dart was nowhere to be found. However, since the announcement of Flutter in Google I/O, Dart has got drastic attention among the mobile developers as an alternative to React Native. The developers who didn't like JavaScript as the programming

language found Dart as an alternative option. As big companies like Google, Alibaba etc. adopted Flutter, the popularity of Dart raised considerably, but it's not near as popular as JavaScript.

2.5.3 Productivity

JavaScript has countless frameworks and new JavaScript frameworks land in the market almost every year. As there is a need for developers to share and distribute code, there are thousands of JS packages available online. With the use of an existing package and experience of the developers, it's easy for new developers to learn and adopt the JavaScript programming language. As JavaScript is a fast, lightweight and dynamic programming language, it boosts the developer productivity. Solutions to common problems can be found online easily, which is another reason that developers prefer JavaScript over other programming languages. Although JavaScript has countless libraries and frameworks available, not all of them are equally good. Also new JavaScript frameworks are released after a regular interval, so the developers constantly need to learn new frameworks, which hinders their productivity.

2.5.4 JavaScript as both Frontend And Backend

JavaScript was originally used for frontend web development with HTML and CSS. However, with the rise of the Node.js framework, JavaScript is now widely used for server-side and backend development as well.

2.6 Back-end Development

Backend development (also stylized as back-end or back end development) is the skill that powers the web. Yet it does it modestly, without fanfare—allowing people to browse their favorite sites without even knowing about all the work put in by the backend developer or team.

Backend development languages handle the ‘behind-the-scenes’ functionality of web applications. It's code that connects the web to a database, manages user connections, and powers the web application itself. Backend development works in tandem with the front end to deliver the final product to the end user. Backend developers are primarily focused on how a website works. They write code that focuses on the functionality and logic powering the application they're working on, and the technology they work on is never directly seen by users. The tech of the **back end is a combination of servers, applications, and databases.**

Responsibilities of backend programmers could involve writing APIs, writing code to interact with a database, creating libraries, working on business processes and data architecture, and much more. It often depends on the specific role and company.

2.6.1 XAMPP Server

XAMPP is an open source software developed by Apache friends. XAMPP software package contains Apache distributions for Apache server, MariaDB, PHP, and Perl. And it is basically a local host or a local server. This local server works on your own desktop or laptop computer. The use of XAMPP is to test the clients or your website before uploading it to the remote web server. This XAMPP server software gives you the suitable environment for testing MYSQL, PHP, Apache and Perl projects on the local computer. XAMPP has been designed to be the easiest way to install and run a development server on your local computer.

- In the web browser , You can easily download XAMPP from <https://www.apachefriends.org/index.html> Since it is an open source you can download it for free.
- Once the software is downloaded, you have to install by double click the .exe file.
- Once the file is executed, a setup window appears. In the setup file, select the components that are required. For eg (if you want to install WordPress on XAMPP, the required components are MySQL, Apache, PHPMyAdmin).
- Next step is to choose the folder where the file is to be located. It is recommended to choose the default “C drive” and then click “Next” button.
- Clicking on the next your installation process will be started. The setup wizard will unpack and install all the selected components and will save them to their designated directory. The installation process takes a couple of minutes to complete.
- Once the installation process is completed, click on the “Finish” button.
- Once done, the XAMPP icon will appear on your desktop or start menu. By double-clicking the XAMPP icon, XAMPP control panel window appears.
- In the XAMPP control panel, click the necessary modules that are required for you to work. You can start the modules by clicking the “Start” button under “Actions”. You will be able to see the initiated modules highlighted in green.

- Then by clicking the Apache or any “Admin” button in the XAMPP control panel, you will be able to configure each module settings separately.
- That's it, now in your web browser type localhost and you will be able to see a splash screen of XAMPP. Here you have to choose a language which you understand.
- By clicking the English button, you will be at the admin page of XAMPP. Now you have successfully installed XAMPP on your system.

2.6.1.1 Creation of first Database

- With XAMPP up and running, it's time to teach you how to install WordPress, but before we do that, you will need to create a database first. Open your XAMPP Control Panel and click on the “Admin” button of the MySQL section, which will lead you to the phpMyAdmin page.
- Alternatively, you can reach this page by typing “<http://localhost/phpmyadmin/>” (without the quotation marks) into your browser.
- Find “Databases” link on the top of the page, and right under it you will find “Create Database”. Enter the desired name and click “Create”.

2.6.2 MySQL

MySQL is an open source relational database management system (RDBMS) with a client-server model that uses Structured Query Language (SQL). RDBMS is a software or service used to create and manage databases based on a relational model.

SQL is the most popular language for adding, accessing and managing content in a database. It is most noted for its quick processing, proven reliability, ease and flexibility of use. MySQL is an essential part of almost every open source PHP application. Good examples for PHP & MySQL-based scripts are Wordpress, Joomla, Magento and Drupal.

MySQL and SQL are not the same. Be aware that MySQL is one of the most popular RDBMS software's brand names, which implements a client-server model. So, how do the client and server communicate in an RDBMS environment? They use a domain-specific language – Structured Query Language (SQL). If you ever encounter other names that have SQL in them, like PostgreSQL and Microsoft SQL server, they are most likely brands which also use Structured Query Language syntax. RDBMS software is often written in other programming languages, but always use SQL as their primary language to interact with the database. MySQL itself is written in C and C++.

Computer scientist Ted Codd developed SQL in the early 1970s with an IBM based relational model. It became more widely used in 1974 and quickly replaced similar, then-outdated languages, ISAM and VISAM. History aside, SQL tells the server what to do with the data. It is similar to your WordPress password or code. You input it into the system to gain access to the dashboard area. In this case, SQL statements can instruct the server to perform certain operations:

- Data query: requesting specific information from the existing database.
- Data manipulation: adding, deleting, changing, sorting, and other operations to modify the data, the values or the visuals.
- Data identity: defining data types, e.g. changing numerical data to integers. This also includes defining a schema or the relationship of each table in the database.

- Data access control: providing security techniques to protect data, this includes deciding who can view or use any information stored in the database.

2.6.3 Database

A database is simply a collection of structured data. Think of taking a selfie: you push a button and capture an image of yourself. Your photo is data, and your phone's gallery is the database. A database is a place in which data is stored and organized. The word "relational" means that the data stored in the dataset is organized as tables. Every table relates in some ways. If the software doesn't support the relational data model, just call it DBMS. things like analytics, authentication, databases, configuration, file storage, push messaging, and the list goes on. The services are hosted in the cloud, and scale with little to no effort on the part of the developer.

2.6.4 PHP

The **PHP Hypertext Preprocessor (PHP)** is a programming language that allows web developers to create dynamic content that interacts with databases. PHP is basically used for developing web based software applications. This tutorial helps you to build your base with PHP.

2.6.4.1 Why to Learn PHP?

PHP started out as a small open source project that evolved as more and more people found out how useful it was. Rasmus Lerdorf unleashed the first version of PHP way back in 1994.

PHP is a MUST for students and working professionals to become a great Software Engineer specially when they are working in Web Development Domain. I will list down some of the key advantages of learning PHP:

- PHP is a recursive acronym for "PHP: Hypertext Preprocessor".
- PHP is a server side scripting language that is embedded in HTML. It is used to manage dynamic content, databases, session tracking, even build entire e-commerce sites.
- It is integrated with a number of popular databases, including MySQL, PostgreSQL, Oracle, Sybase, Informix, and Microsoft SQL Server.
- PHP is pleasingly zippy in its execution, especially when compiled as an Apache module on the Unix side. The MySQL server, once started, executes even very complex queries with huge result sets in record-setting time.
- PHP supports a large number of major protocols such as POP3, IMAP, and LDAP. PHP4 added support for Java and distributed object architectures (COM and CORBA), making n-tier development a possibility for the first time.
- PHP is forgiving: PHP language tries to be as forgiving as possible.
- PHP Syntax is C-Like.

2.6.4.2 Characteristics of PHP

Five important characteristics make PHP's practical nature possible –

- Simplicity
- Efficiency
- Security
- Flexibility
- Familiarity

2.6.4.3 Hello World using PHP.

Just to give you a little excitement about PHP, I'm going to give you a small conventional PHP Hello World program, You can try it using Demo link.

```
<html>

  <head>
    <title>Hello World</title>
  </head>

  <body>
    <?php echo "Hello, World!";?>
  </body>

</html>
```

2.6.4.4 Applications of Python

As mentioned before, PHP is one of the most widely used language over the web. I'm going to list few of them here:

- PHP performs system functions, i.e. from files on a system it can create, open, read, write, and close them.
- PHP can handle forms, i.e. gather data from files, save data to a file, through email you can send data, return data to the user.
- You add, delete, modify elements within your database through PHP.

- Access cookies variables and set cookies.
- Using PHP, you can restrict users to access some pages of your website.
- It can encrypt data.

CHAPTER – 3

LITERATURE REVIEW

Bank blood donation system is planned to collect blood from many donators in short from various sources and distribute that blood to needy people who require blood. To do all this we require quality software to manage those jobs . The system will provide the user the option to look at the details of the existing Donor List, Blood Group and to add a new Donor. It also allows the user to modify the record. The administrator can alter all the system data . Online Blood Bank system is now a day's most widely used and popular because this is less time consuming and request for the blood from any of the area. This can also save the time [6]. Online Blood Bank system provides the accuracy factor because all the data is logged into the database safely and the user is provided with the receipt number for his data. The data is then sending safely to the administration and they may interact with the client as per his requirement . This web application is also very helpful because a helpdesk system is available for the user where the user get fully help either they want any information, or they want to submit any complain or feedback [8]. The system will provide the user the option to look at the details of the existing Donor List, Blood Group and to add a new Donor. It also allows the admin to modify the record. The administrator can alter all the system data . It also allow the user to see the blood request and donate blood according to the blood request

CHAPTER – 4

PROPOSED WORK

The proposed system is a web based online application that provides the user with the facility of detailed information of Donor or Blood request. This system clears the factor of confusion from user mind about information. This system makes conveniently available good quality, safe blood and other blood components, which can be provided in a sound, ethical and acceptable manner, consistent with the long-term wellbeing of the community [. This application is built such a way that it should suits for all type of blood banks in future. So every effort is taken to implement this project in this blood bank, on successful implementation in this blood bank It actively encourage voluntary blood donation, motivate and maintain a well-indexed record of blood donors and educate the community on the benefits of blood donation . The proposed system is efficient and better than existing system because of the following reason: • This system provide a detailed information platform to the user • The manual system is replaced with the online system • All the data and the record are stored safely in the database in this system

4. PROPOSED WORK

4.1 INTRODUCTION

The main purpose of a . It is the description of the software system to be developed. An SRS lists the necessary requirements that are needed for the project. To obtain the requirements, the products to be developed should be clearly understood. This can be achieved by the continuous communication between the developer and customer.

4.2 USERS/MODULES

“Students Grievance Redressal System” is focusing on three users: Admin, Users and Head of Department (HOD). Admin has the responsibility to add various departments to the system. Users can register their grievance to the site. The Department head can view the grievances and take action.

4.2.1 Admin

The admin part handles the administration part of the whole system. The admin can handle the overall responsibility of the system. The admin user should be able to:

- Add HOD: Admin can add HODs for the various departments of the system to which the users can register their complaint.
- View Users: Admin can view the details of all the users who are registered in the system.
- View Complaints: Admin can view all the grievances registered by the public users.
- Generate report: Admin can view report of complaints that are registered and solved.

4.2.2 User (Student)

The User module handles the activities of the users registered in the system. The user is able to registers complaint to a particular department and checks the status. The user should able to:

- Register complaint: Users can register their complaints to a particular department.
- Check complaint status: Users are able to view their complaint status to know the action taken.
- Reopen complaint: Users can reopen their complaint which has been closed by the department head after solved.
- Send feedback: Users can send feedback to different departments.

4.2.3 Head of Department

The HOD of each department can login to the system using username and password. They can view the complaints that are registered to their department and approve the complaints. The department

head can change the status of the complaint to modify the progress of complaint. This user should be able to:

- View grievance: The department HOD can view all the complaints received by their department.
- Approve complaint: HOD has to view and approve the complaint before processing on it.
- Change status: The HOD can change the status of each complaint based on the action taken.
- Close complaint: The HOD can close any complaints that have been solved.
- View feedback: The Department head can view the feedback sent by the users.

4.3 FUNCTIONAL REQUIREMENTS

Functional Requirements of “Students Grievance Redressal System” are:

4.3.1 Registration

Registration module is used for registering the user(only for Admin and HOD) account into the system.

- **Input:** Name, dob, username, password, phone, address.
- **Output:** User is registered to system.
- **Processing:** All fields are entered as per the constraints

4.3.2 Login

Login is used so that the authorized person can enter into the system. Only registered users can login.

- **Input:** username, password.
- **Output:** User is logged in and authorized page is displayed.
- **Processing:** Login is not allowed if wrong username and password is entered.

4.3.3 Complaint

This is used by the Students to register their grievances through this website.

- **Input:** Department and Complaint.
- **Output:** Complaint is registered successfully.
- **Processing:** Complaint is saved in the particular department.

4.3.4 Status/Feedback

This is used by the user to check the status of their registered complaint.

- **Input:** Complaint id
- **Output:** Status is displayed.
- **Processing:** Valid complaint id should be entered to view status.

4.3.5 History

It is used for generating complaint report by both in the admin and HOD Home page.

- **Input:** start date, end date, status
- **Output:** History of complaint is generated.

4.4.1 Performance requirements

The performance of the proposed system must be good. It should be fast and reliable.

The system should have built-in error checking and correction techniques.

To run this application, the requirements needed are:

- An internet connection.
- A Web browser.

CHAPTER – 5

IMPLEMENTATION

5. IMPLEMENTATION

5.1 Code Snippets

LOGIN DETAILS:

```
<?php
```

```
require 'functions/functions.php';  
require_once 'connection.php';
```

```
if (isset($_SESSION['user'])) {  
    header("location:admin1.html");  
}  
session_destroy();  
session_start();  
ob_start();  
?>
```

```
<?php
```

```
$user= $pass="";  
if(isset($_POST['sub']))  
{  
  
    $user=$_POST['user'];  
    $pass=$_POST['pass'];  
    $q=$db->prepare("SELECT * from admin where user='$user' && pass='$pass'");  
    $q->execute();  
    $res=$q->fetchAll(PDO::FETCH_OBJ);  
    if($res)  
    {  
        $_SESSION['user']=$user;  
        header("Location:admin1.html");  
    }  
    else  
    {  
        echo"<script>alert('You are not an admin')</script>";  
    }  
}
```

```
function test_input($data){  
    $data=trim($data);  
    $data=stripslashes($data);
```

```

        $data=htmlspecialchars($data);
        return $data;

    }

}
?>

```

Adminhome:

```

<?php
require 'functions/functions.php';
session_start();
// Check whether user is logged on or not
if (!isset($_SESSION['user'])) {
    header("location:index.php");
}
$user = $_SESSION['user'];
session_destroy();
session_start();
$_SESSION['user'] = $user;
ob_start();

$conn = connect();
?>

```

```

<!DOCTYPE>
<html>
<head>
<title>Admin Home</title>
    <link rel = "icon" href =
"images/logo.png"
        type = "image/x-icon">
    <style type="text/css">
</style>

```

```
body
{
margin: 0;
padding: 0;
color:red;
background-image:url("seamless.jpg");
}
```

```
ul
{
list-style-type: none;
background-color: red;
height: 40px;
width:100%;
```

```

}
a
{
text-decoration: none;
font-size: 25px;
color:white;
position: relative;
top: 4px;
}
```

```
li
{
display: inline;
font-style: 15px;
padding: 38px;
margin-left: 20    px;
position: relative;
left: 70px;
```

```
}
```

```
p
{
text-align: justify;
max-width: 1000px;
```

```

    max-height: 80px;
    position: absolute;
    left: 60px;
    font-size: 20px;
    top: 350px;
    margin-left: 95px;
}
h6
{
    position: relative;
    top: 180px;
    left: 60px;
    font-size: 20px;
    margin-left: 95px;
}

.bloodquery
{
    position: relative;
    top: 160px;
    left: 60px;
    font-size: 20px;
    margin-left: 95px;
}

.carousel-inner > .item > img,
.carousel-inner > .item > a > img {
    width: 70%;
    margin: auto;
}
.carousel-inner > .item > img,
.carousel-inner > .item > a > img {
    width: 50%;
    margin: auto;
}
.ongoingcamps
{
    position: relative;
    top: 180px;
    left: 60px;
    font-size: 20px;

```

```

    margin-left: 95px;

}

.saap{
    position: relative;
    top:160px;
    left: 60px;
    font-size: 20px;
    margin-left:40px;
    color:black;
}

footer{
    position: relative;
    bottom: 10;
}

@media (max-height:100px){
    footer { position: static; }
    header { padding-top:20px; }
}

.footer-distributed{
    background-color: #2c292f;
    box-sizing: border-box;
    width: 100%;
    text-align: left;
    font: bold 16px sans-serif;
    padding: 20px 20px 20px 20px
    margin-top:100px;
    height:220px;
}

.footer-distributed .footer-left,
.footer-distributed .footer-center,
.footer-distributed .footer-right{
    display: inline-block;
    vertical-align: bottom;
}

```

```
.footer-distributed .footer-left{  
    width: 100%;  
    height:100px;  
}
```

```
.footer-distributed h3{  
    color: #ffffff;  
    font: normal 36px 'Cookie', cursive;  
    margin: 0;  
}
```

```
.footer-distributed .footer-left img{  
    width: 30%;  
    height:20px;  
}
```

```
.footer-distributed h3 span{  
    color: #e0ac1c;  
}
```

```
.footer-distributed .footer-links{  
    color: #ffffff;  
    margin: 20px 0 12px;  
}
```

```
.footer-distributed .footer-links a{  
    display:inline-block;  
    line-height: 1.8;  
    text-decoration: none;  
    color: inherit;  
}
```

```
.footer-distributed .footer-company-name{  
    color: #8f9296;  
    font-size: 10px;
```

```

    font-weight: normal;
    margin: 0;
}
.footer-distributed .footer-center{
    width: 30%;
    height: 20px;

}
.footer-distributed .footer-center
{
    background-color:#33383b;
    color: #ffffff;
    font-size: 25px;
    width: 10px;
    height: 10px;
    border-radius: 50%;
    text-align: center;
    line-height: 20px;
    margin: 10px 15px;
    vertical-align: middle;
}
.footer-distributed .footer-center i.fa-envelope{
    font-size: 17px;
    line-height: 20px;
}
.footer-distributed .footer-center p{
    display: inline-block;
    color: #ffffff;
    vertical-align: middle;
    margin:0;
}
.footer-distributed .footer-center p span{
    display:block;
    font-weight: normal;
    font-size:14px;
    line-height:20px;
}
.footer-distributed .footer-center p a{
    color: #e0ac1c;
    text-decoration: none;;
}

```



```
.footer-distributed .footer-right{  
    width: 30%;  
}
```

```
.footer-distributed .footer-company-about{  
    line-height: 20px;  
    color: #92999f;  
    font-size: 13px;  
    font-weight: normal;  
    margin: 0;  
}
```

```
.footer-distributed .footer-company-about span{  
    display: block;  
    color: #ffffff;  
    font-size: 18px;  
    font-weight: bold;  
    margin-bottom: 20px;  
}
```

```
.footer-distributed .footer-icons{  
    margin-top: 25px;  
}
```

1

```
.footer-distributed .footer-icons a{  
    display: inline-block;  
    width: 35px;  
    height: 35px;  
    cursor: pointer;  
    background-color: #33383b;  
    border-radius: 2px;  
  
    font-size: 20px;  
    color: #ffffff;  
    text-align: center;  
    line-height: 35px;  
  
    margin-right: 3px;  
    margin-bottom: 5px;  
}
```

```
.footer-distributed .footer-left,
```

```

.footer-distributed .footer-center,
.footer-distributed .footer-right{
    display: block;
    width: 100%;
    margin-bottom: 40px;
    text-align: center;
}

.footer-distributed .footer-center i{
    margin-left: 0;
}

}
</style>
</head>

<body>

<div>

<p style="text-align:center;font-size:30px;font-family:cursive;color:black;text-decoration:underline;border-radius:0.5px;margin-left:585px;margin-top:-362">Welcome Admin</p>


<ul>

<li><a href="user.php" style="font-size:20px">Users LoggedIn Details</a></li>
<li><a href="bloodupdate.php" style="font-size:20px">Update Blood Details For User</a></li>
<li><a href="camps.html" style="font-size:20px">Update Camps details for donar</a></li>
<li><a href="viewrequest.php" style="font-size:20px">See Who Request for Blood</a></li>
<li><a href="viewrequest.php" style="font-size:20px"></a></li>ect
<li><a href="logout.php" style="font-size:20px">Logout</a></li>
</ul>

```

<p style="color:blue;font-style: italic;margin-top:120px;"> Blood is universally recognized as the most precious elements that sustain life. It saves innumerable lives across the world in a variety of condition. The need of blood is great- on any given day, approximately 39,000 units of Red blood cells are needed. More than 29 million units of blood component are transfused every year. Donate Blood Despite the increase in the number of donors, blood remains in short supply during emergencies, mainly attributed to lack of information and accessibility. We positively belive this tool can overcome most of these challenges by effectively connecting the blood donors with blood recipients. </p>

*
*

<div class="bloodquery">

</div>

</body>

</html>

USER DETAILS:

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

<title>Blood Request</title>
  <link rel = "icon" href =
"images/logo.png"
    type = "image/x-icon">
  <style type="text/css">
  <style type="text/css">
  body{

      color:black;
      background-image: url("css/bg.jpg");
      text-align: center;
      font-size:10px;

      }
```

```
div
{
    position:relative;
    left:7cm;

}
```

```
table, td, th {
  border: 5px solid black;
  text-align: center;
  font-size:20px;
}
```

```
table {
  border-collapse: collapse;
  width: 100%;
}
```

```

th, td {
    padding: 20px;
}
h1{

    color:black;
    font-size:30px;
    text-align:left;
}
</style>
<title>Blood Details</title>
</head>

<body>

<div>
<?php

session_start();

$conn= mysqli_connect('localhost', 'root', '');
mysqli_select_db($conn,'bloodbank');
echo "<h1 style=position:center;>List of Users who Logged In and their Details</h1>";
$stmt = "select * from login;";
$result = mysqli_query($conn, $stmt);
if(mysqli_num_rows($result)>0)
{
    while($rows = mysqli_fetch_assoc($result))
    {
        echo "<table style=position:relative;width:15cm;>";
        echo "<tr>";
        echo "<th>user</th>";
        echo "<th>useremail</th>";
        echo "<th>bloodgroup</th>";
        echo "</tr>";
        echo "<tr style=position:relative;width:13cm;>";
        $user= $rows['user'];
        echo "<td>$user</td>";
        echo "<br>";
        $useremail= $rows['useremail'];
        echo "<td>$useremail</td>";
    }
}

```

```

        echo "<br>";
        $bloodgroup= $rows['bloodgroup'];
        echo "<td>$bloodgroup</td>";
        echo "<br>";

    }
}
else
{
    echo "No Blood Available";

}

?>

</div>
</body>
</html>

```

VIEW REQUESTS:

```

<?php
session_start();
include '../config.php';

// for deleting user
if(isset($_GET['id']))
{
    $adminid=$_GET['id'];
    $msg=mysqli_query($con,"delete from requestblood where id='$adminid'");
    if($msg)
    {
        echo "<script>alert('Data deleted');</script>";
    }
}
?>
<!DOCTYPE>
<html>
<head>

<title>Blood Request</title>

```

```
<link rel = "icon" href =  
"images/logo.png"  
    type = "image/x-icon">  
<style type="text/css">  
body{  
    background-image:url("seamless.jpg");  
}  
<!-- /* body{  
  
    color:black;  
    background-image:url("seamless.jpg");  
    text-align: center;  
    font-size:10px;  
  
    }  
}
```

```
div  
{  
    position:relative;  
    left:7cm;  
  
}
```

```
table, td, th {  
    border: 5px solid black;  
    text-align: center;  
    font-size:20px;  
}
```

```
table {  
    border-collapse: collapse;  
    width: 100%;  
}
```

```
th{  
  
    font-size: 24px;  
    color:blue;
```

```

        font-style: bold;
    }
    td {
        padding: 20px;
    }
    h1{

        color:black;
        font-size:30px;
    }
    */ --></style>
<link rel="stylesheet" type="text/css" href="css/bootstrap.min.css">
</head>

<body>
<p style="text-align:center;font-size:60px;font-family:cursive;color:black;text-decoration:underline;border-radius:0.5px">BBMS</p>
<section id="main-content">
    <section class="wrapper">
        <div class="row">
            <div class="col-md-12">
                <div class="content-panel">
                    <table class="table table-striped table-advance table-hover">
                        <h4><i class="fa fa-angle-right"></i> All User Details </h4>
                        <hr>
                        <thead>
                            <tr>
                                <th>Id</th>
                                <th class="hidden-phone">First Name</th>
                                <th> Address</th>
                                <th> Bloodgroup</th>
                                <th>Contact no.</th>
                                <th>unit</th>
                                <th>What date & time </th>
                                <th> time</th>
                            </tr>
                        </thead>
                        <tbody>
                            <?php

                                $ret=mysqli_query($con,"select * from requestblood");

```



```

$cnt=1;
while($row=mysqli_fetch_array($ret))
{?>

<tr>
<td><?php echo $cnt;?></td>
    <td><?php echo $row['user'];?></td>
    <td><?php echo $row['Address'];?></td>
    <td><?php echo $row['bloodgroup'];?></td>
    <td><?php echo $row['phno'];?></td>
    <td><?php echo $row['unit'];?></td>
    <td><?php echo $row['time-for-flood'];?></td>
    <td><?php echo $row['time'];?></td>
    <td>

        <a href="message.php?uid=<?php echo $row['id'];?>">
            <button class="btn btn-primary btn-xs"><i class="fa fa-
pencil"></i></button></a>
            <a href="viewrequest.php?id=<?php echo $row['id'];?>">
                <button class="btn btn-danger btn-xs" onClick="return confirm('Do you really
want to delete');"><i class="fa fa-trash-o "></i></button></a>
            </td>
        </tr>
    <?php $cnt=$cnt+1; }?>

</tbody>
</table>
</div>
</div>
</div>
</div>
</section>
</div>
</div>
<div class="footer-distributed">
<center>
<br>
<br>
<h3 style="color:black;font-size: 20px">&copy 2019 BBMS</h3>
<a href="adminhome.php" style="color:black;font-size: 20px">Back to home</a>
<br>
<br>
</center>

```

</body>
</html>

CHAPTER – 6

SYSTEM ANALYSIS

5.SYSTEM ANALYSIS

6.1 Problem Statement

The current system is dealing on the manual basis. People have to go directly to the concerned departments to give their complaint. It needs huge amount of paper works to maintain the complaint details. A huge expenditure and lots of time is spent in the existing system. Tracking and retrieving of data from bulk of papers is a difficult process.

6.1.1 Existing System

The existing system is the manual system and for that if the any request for the blood go in the request for blood page and check the blood request .Than he came to know that the blood request is available or not.

6.1.1.1 Disadvantages

- Inaccessibility
- Lower quality data
- Limited flexibility
- Non eco-friendly
- Lengthy process

6.1.2 Proposed System

The proposed system is a web based online application that provides the user with the facility of detailed information of Donor or Blood request. This system clears the factor of confusion from user mind about information. This system makes conveniently available good quality, safe blood and other blood components, which can be provided in a sound, ethical and acceptable manner, consistent with the long-term wellbeing of the community [11]. This application is built such a way that it should suits for all type of blood banks in future. So every effort is taken to implement this project in this blood bank, on successful implementation in this blood bank It actively encourage voluntary blood donation, motivate and maintain a well-indexed record of blood donors and educate the community on the benefits of blood donation [12]. The proposed system is efficient and better than existing system because of the following reason: • This system provide a detailed information platform to the user • The manual system is replaced with the online system • All the data and the record are stored safely in the database in this system
Vignan's Institute of Information Technology (2017-2020)

6.1.2.1 Advantages

The web complaint management software allows the user to record the complaint and loads it to the complaints database.

- It offers a strategic method for online lodging and tracking of customer concerns and issues.
- You can access each complaint's history and check the status update. It has the details of how a complaint is being progressed and what actions are being taken.
- After a complaint is recorded by a user, it is reviewed by an Administrator and then led to the complaints workflow.
- The admin can assign it to a particular service engineer for resolving the issue.
- The Service engineer then acts on it and updates the status.

6.2 Objectives

The main objective of the Blood Bank Management System is to manage the details of Blood ,Donor,Blood Group,Blood Bank,Stock. It manages all the information about Blood , Blood Cell, Stock, Blood . The project is totally built at administrative end and thus only the administrator is guaranteed the access

6.3 Feasibility Study

- Feasibility study is made to see if the project on completion will serve the purpose of the organization for the amount of the work, effort and the time that spend on it. Feasibility study lets the developer for see the future of the project and the usefulness. A feasibility study of the system proposal is according to its workability, which is the impact on the organization, ability to meet their user needs and effective use of resources. As the name implies, a feasibility study is used to determine the viability of an idea, such as ensuring a project is legally and technically feasible as well as economically justifiable. It tells us whether a project is worth the investment in some cases, a project may not be doable. Feasibility studies allow companies to determine and organize all the details to make a business work. A feasibility study helps identify logistical problems, and nearly all business related problems and their solutions. Feasibility studies can also lead to the

development of marketing strategies that convince investors or a bank that investing in the business is a wise choice.

6.3.1 Technical feasibility

The system should be technically feasible. The system should be designed and developed by the person with good technical knowledge. The Student Grievance Redressal System meets all technical aspects need for the system. It uses front end as HTML,CSS,JavaScript , back end as MySQL. It uses Notepad++ as the IDE and Windows 8 as operating system. This application needs an internet connection to use processor Pentium 4 and RAM with 1GB or more.

6.3.2 Economic feasibility

The system should be economically feasible. The cost for the development of the system is low for the organization. The development process is achieved by using the needed resources, so organizations need not invest more and so, the system can be considered as economically feasible. Student Grievance Redressal system is a low cost application that is developed for the purpose of public users. It is useful for the citizens.

6.3.3 Legal feasibility

The system is legally feasible. It checks whether the proposed system conflicts the legal requirements. The security level of the application is checked.

6.3.4 Operational feasibility

This system describes all the operational aspects in each level. Since it is a web based system, it can be operated anywhere with internet connection.

6.3.5 Schedule feasibility

Schedule feasibility is mainly based on the time period taken to complete the development of the system. Student Grievance Redressal System have been completed within the decided time period.

CHAPTER – 7
REQUIREMENT AND SPECIFICATION

7.REQUIREMENTS AND SPECIFICATION

7.1 System Specifications

7.1.1 Software Requirements

- Operating system
- Installed XAMPP
- Notepad
- Notepad++
- Visual Studio.

7.1.2 Hardware Requirements

At Server side

- 16 GB RAM
- 512 GB SDD

7.2 Development Tools

- We made our model using Web development.
- Languages used: HTML,CSS,JAVASCRIPT,PHP,MYSQL.
- We used Visual Studio and Notepad as editor.

7.3 Features

- Inexpensive.
- User Friendly.
- Fully Informative.
- Reduced Human efforts.
- Increased accuracy and reliability.

8.CONCLUSION

8.1 Conclusion

This project has given me an ample opportunity to design, code, test and implements an application. This has helped in putting into practice of various Software Engineering principles and Database Management concepts like maintaining integrity and consistency of data. Further, this has helped me to learn more about PHP, bootstrap, css, JAVASCRIPT ASP 2.0, HTML, VB Script, Adobe Photoshop 7.0 and Personal Web Server. I thank my guide for his invaluable contribution in guiding me throughout the project. I also thank my parents and friends who have supported and motivated me to complete this project successfully. Extensibility: The other features, which the Blood bank services provide, can also be incorporated into this Blood Bank. The Encryption standards can also be used to make the transactions more secure. The Socket Secure Layer protocol can also used in implementing the system, which gives highest security in the Internet. Future Enhancement: As there was a little number of contact person's information given, some people may face difficulty in getting blood fast. So i like to gather more information regarding the contact persons in other cities as well as villages and will provide much more services for the people and help everyone with humanity.

8.2 Future Scope

This application is developed in such a way that any future detection can be done easily..

- This application is created only for benchmark datasets. In future, this can be expanded to multiple states.
- Mobile application can be developed for this system.
- The system can be upgraded by including more features like video facilities etc.

CHAPTER – 9

REFERENCES

9.REFERENCES

- [1]Fiji Accommodation. onlinetravel.com. Retrieved April 13, 2013, from <http://www.tourismfiji.com/>
- [2]B. Brown and M. Chalmers, "Tourism and mobile technology," pp. 339, 2003.
- [3]Gordon M., Paprzycki M., "Designing Agent Based Travel Support System"
- [4]G. M and P. M., "Designing Agent Based Travel Support System," in Parallel and Distributed Computing, 2005.
- [5]Maes P., "Agents that Reduce Work and Information Overload," Communications of the ACM, 37(7), 1994, 31-40
- [6]Paprzycki M., Abraham A., "Agent Systems Today; Methodological Considerations," in: Proceedings of 2003 International Conference on Management of e-Commerce and e-Government, Jangxi Science and Technology Press, Nanchang, China, 2003, pp. 416-421
- [7]Nwana H., Ndumu D. (1999) "A perspective on software agents research," The Knowledge Engineering Review, 14 (2), pp. 1-18
- [8]Ndumu, D., Collins, J., Nwana, H. (1998) "Towards Desktop Personal Travel Agents," BT Technological Journal, 16 (3), pp. 69-78
- [9]Suarez J. N., O'Sullivan D., Brouchoud H., Cros P. (1999) "Personal Travel Market: Real-Life Application of the FIPA Standards." Technical Report, BT, Project AC317
- [10] M. Umlauft, G. Pospischil, G. Niklfeld, E. Michlmayr: LoL@, a Mobile Tourist Guide for UMTS. Journal on Information Technology & Toursim, H. Werthner and E. Veit (ed.) Cognizant, Vol. 5, No. 3; pp. 151-164, March 2003.
- [11] W. Schwinger, C. Grün, B. Pröll, W. Retschitzegger and A. Schauerhuber, "Context-awareness in mobile tourism guides–A comprehensive survey," Rapport Technique. Johannes Kepler University Linz, 2005.

- [12] H. K. Eden and U. Gretzel, "A taxonomy of mobile application in tourism," E-review of Tourism Research, vol. 10, no. 2, pp. 47-50, 2012.
- [13] Kenteris M., Gavalas D., Economou D., "Evaluation of Mobile Tourist Guides"
- [14] Jian Meng, Neng Xu, "A Mobile Tourist Guide System Based on Mashup Technology" ISBN 978-1-4244-7618-3 / 10 ©2010 IEEE.
- [15] H. K. Eden and U. Gretzel, "A taxonomy of mobile application in tourism," E-review of Tourism Research, vol. 10, no. 2, pp. 47-50, 2012

- [16] Andress, M., The road to secure web services. InfoWorld, 2002. 24(2): p. 52.
- [17] E. Cuervo, A. Balasubramanian, D.-k. Cho, A. Wolman, S. Saroiu, R. Chandra and P. Bahl, "Making Smartphones Last Longer with Code Offload," Proceedings of the 8th international conference on Mobile systems, applications, and services, 2010.
- [18] R. Prikladnick, T. Duarte, T. Conte, F. Calefato and F. Lanubile, "RealTime Machine Translation for Software Development Teams," pp. 1-4, 2013.
- [19] Phillip Miller, F. L. K., Sushil K. Sharma A Flexible Services Architecture Based Translator Web Services. International Journal of Mobile Communications, 1, 3 2003), 301-311
- [20] R. Jafri, "Smart Travel Planner: A mashup of travel-related web services," Current Trends in Information Technology (CTIT), pp. 181 - 185 , 2013.
- [21] I. R. Floyd, M. C. Jones, D. Rathi, and M. B. Twidale. Web mash-ups and patchwork prototyping: User-driven technological innovation with web 2.0 and open source software. System Sciences, 2007. HICSS 2007. 40th Annual Hawaii International Conference on, pages 86 –86, jan. 2007.
- [22] Singhal, Manav and A. Shukla, "Implementation of location based services in Android using GPS and Web services," IJCSI International Journal of Computer Science Issues, vol. 9, no. 1, pp. 237-242, January 2012.
- [23] Google Maps. (n.d.). Retrieved May 21, 2013, from <https://maps.google.com>
- [24] Google Places API. (n.d.). Retrieved May 20, 2013, from Supported Place Types: https://developers.google.com/places/documentation/supported_types
- [25] M. H. Goadrich and M. P. Rogers, "Smart Smartphone Development: iOS versus Android," Proceedings of the 42nd ACM technical symposium on Computer science education, 2011.
- [26] Google APIs Console. (n.d.). Retrieved May 21, 2013, from <https://code.google.com/apis/console/>
- [27] google-api-java-client. (n.d.). Retrieved May 20, 2013, from Google APIs Client Library for Java: <http://code.google.com/p/google-api-java-client/wiki/Setup>