Chapter 1

INTRODUCTION

The Blogging System is a web-based application. It provides a user-friendly, interactive interface based on PHP, CSS, JavaScript and HTML5 elements. All data is stored in a MySQL database. The application uses WAMP Server to communicate with the database.

1.1 Background

Interest in blogging has increased in recent years with Twitter, etc. The main motivation behind the selection of this project was to design, develop and implement a software application which will be useful for the main stake holders of a typical blogging website. Further, to make the application user interface interactive and user-friendly at the same time was challenging, using Web technologies like PHP, CSS, JavaScript and HTML5.

1.2 About the mini project

The Blogging System provides a website for users to post textual content. The posts are presented in reverse chronological order. The user must first create an account (or Sign Up), which involves creating a login username and password. Creating a User Profile involves filling of a form, with basic information that will make up the Profile. A Webcam option is provided to capture on-spot pictures, that can be used as Profile Pictures in a user's profile. At any time, a user can edit his/her profile.

Users can give titles for their posts. Other account holders can 'comment' on people's blog posts only if they 'follow' them too. A user can see the usernames of the others who are currently online, as well as, of those who have blogged that day. Every blogger is provided with a list of 'blog followers', indicating the maximum number of times followers have commented on their posts. Each user is also provided with the list of 'top followers', which are the users that have made the maximum number of user comments. The website also maintains a ranking of 'top bloggers', which will be those users with the most posts.

Chapter 2

REQUIREMENT SPECIFICATION

A **Requirement Specification** of a system includes Software Requirements Specification (SRS) and Hardware Requirements Specification. It lays out functional and non-functional requirements and may include a set of use cases that describe user interactions that the software must provide.

2.1 Hardware requirements

- 64-bit CPU
- 4-8 GB Memory
- Built-in camera

2.2 Software requirements

- OS (eg. Windows 10)
- PHP
- CSS3
- HTML5
- Apache Web server
- MySQL Database
- JavaScript

2.3 Functional and non-functional requirements

A **Functional Requirement** defines a function of a system or its component, where a function is described as a specification of behavior between outputs and inputs.

A **Non-Functional Requirement** is a requirement that specifies criteria that can be used to judge the operation of a system, rather than specific behaviors The plan for implementing non-functional requirements is detailed in the system architecture, because they are usually architecturally significant requirements.

2.3.1 Functional requirements -User

Each user has his separate account, to perform his own functionalities, identified by a user name and password. Every user is made to create an account before using the application.

2.3.2 Security requirements

It is of utmost importance to ensure that there is protection against unauthorized access to the blog posts. Users must be provided with a login ID and password which grant access only to their respective account. Similarly, every user can only view their own posts and posts of those whom they follow by them. The Admin on the other hand has privileged access in order to view all data and ensure seem less and efficient experience to all of its users.

2.3.3 Performance requirements

The PCs used must be at least be INTEL CORE i3 machines so that they can give optimum performance of the product. In addition to these requirements, the system should also embrace the following requirements:-

- **Reliability:** The system should have little or no downtime.
- **Ease of Use:** The general and administrative views should be easy to use and intuitive.

2.3.4 Design and Interface requirements

The designers must design the database is such a way that any change in the information of a client should be updated and saved effectively in the database.

The interface which is provided in this software allows students to view and comment on blog posts, and the administrator to track all the interactions of the current users. The database designed should be very easy to use and user friendly.

- An account is maintained for every user to ensure security and anonymity.
- User can view his posts and those of whom he follows.

Communication between the MySQL database and front-end is through a WAMP server. It is connected with front-end with a connection, established through PHP, which has inbuilt functions for connectivity.

Chapter 3

ANALYSIS AND DESIGN OF THE SYSTEM

System analysis is a problem-solving technique that breaks down a system into its component pieces for the purpose of the studying how well those component parts work and interact to accomplish their purpose.

3.1 Analysis of the system

The Blogging System has a scope of two levels of users.

- At the first level, the Admin maintains data about posts, comments, followers and user authentication, relevant and critical, for the application to run.
- At the second level, the Users are allowed to post textual content or comment on the posts of other users whom they follow.

3.2 Design of the system

Systems design is the process of defining the architecture, modules, interfaces, and data for a system to satisfy specified requirements. It is the application of systems theory to product development. There is some overlap with system analysis, system architecture and system engineering.

3.2.1 Architecture of the system

A **Data Flow Diagram** (**DFD**) is a graphical representation of the "flow" of data through an information system, modelling its *process* aspects. A DFD is often used as a preliminary step to create an overview of the system without going into great detail, which can later be elaborated. DFDs can also be used for the visualization of data processing (structured design).

A DFD shows what kind of information will be input to and output from the system, how the data will advance through the system, and where the data will be stored. It does not show information about process timing or whether processes will operate in sequence or in parallel, unlike a traditional structured flowchart which focuses on control flow, or a UML activity workflow diagram, which presents both control and data flows as a unified model.

Figure 3.1 shows the data flow diagram for the Blogging System. It illustrates the flow of data from and to the blogger in the system. Bloggers can Sign Up by creating a user profile, Login by providing usernames and passwords for authentication, Post textual content, Comment on others' posts, Follow other bloggers and Edit their user profiles. The system uses a MySQL database to store all the required information in the form of relational tables. In response to user input, the system uses the database to store profile details of users after Sign Up, bloggers' Posts, their Followers, their Comments on others' posts, and uses the data stored to validate users on Login and display some Statistics related t the site.

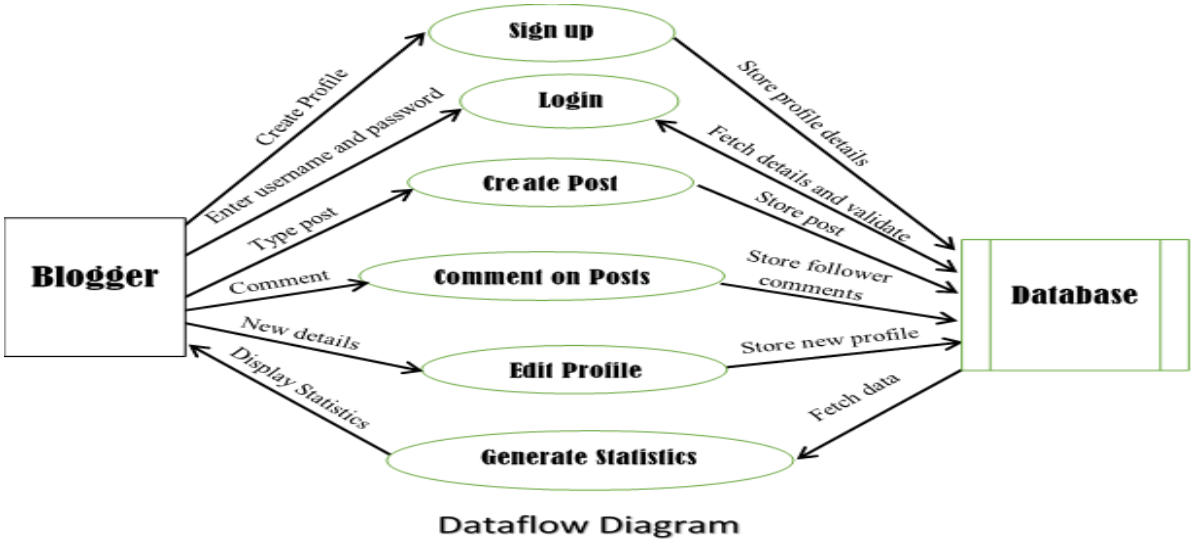

Figure 3.1 System Data Flow Diagram

3.2.2 Use-Case Diagram

A **Use-Case Diagram** (**UCD**) at its simplest is a representation of a user's interaction with the system that shows the relationship between the user and the different use-cases in which the user is involved. A use-case diagram can identify the different types of users of a system and the different use-cases and will often be accompanied by other types of diagrams as well.

Figure 3.2 shows the use-case diagram for the Blogging System. It illustrates the different actions that can be performed by bloggers in order to interact with the application. Users can Sign Up by providing details for mandatory form fields, Login using the chosen username and password, Post textual content, Follow other bloggers, Comment on their posts, Edit their user profiles and even use a Webcam to snap a picture for their profiles.

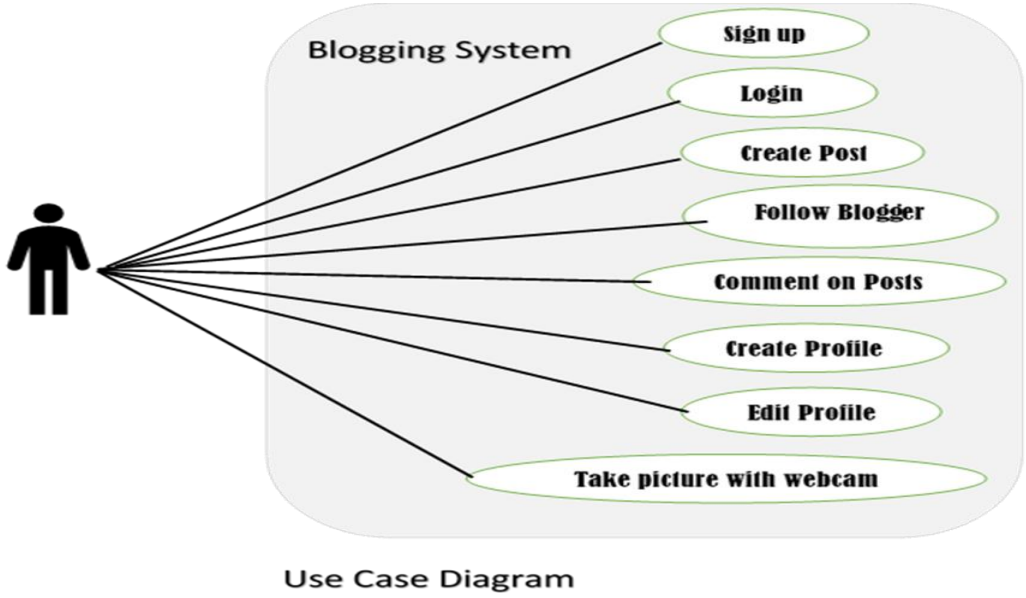

Figure 3.2 System Use-Case Diagram

Chapter 4

IMPLEMENTATION

Implementation is the process of defining how the system should be built, ensuring that it is operational and meets quality standards. It is a systematic and structured approach for effectively integrating a software-based service or component into the requirements of end users.

4.1 Front-end and back-end used

The front-end is everything involved with what the user sees. The back-end, or the "server-side", is basically how the site works, updates and changes. This refers to everything the user can't see in the browser, like databases and servers.

4.1.1 Features of front-end

HTML code along with CSS is used for styling while JavaScript, is used for validation at frontend. PHP is a server-side scripting language designed for Web development, but also used as a general-purpose programming language [9]. PHP code is embedded into HTML.

PHP code is usually processed by a PHP interpreter implemented as a module in the web server or as a Common Gateway Interface (CGI) executable. The web server combines the results of the interpreted and executed PHP code, which may be any type of data, including images, with the generated web page. PHP code may also be executed with a command-line interface (CLI) and can be used to implement standalone graphical applications.

4.1.2 Features of back-end

WampServer refers to a software stack for the Microsoft Windows operating system, created by Romain Bourdon and consisting of the Apache web server, OpenSSL for SSL support, MySQL database and PHP programming language.

The **Apache HTTP Server**, colloquially called **Apache**, is a free and open-source cross-platform web server, released under the terms of Apache License 2.0. Apache is developed and maintained by an open community of developers under the auspices of the Apache Software Foundation. The Apache HTTP Server is cross-platform with Version 2.0 improved support for non-Unix operating systems such as Windows.

MySQL is an open-source relational database management system (RDBMS). MySQL is a central component of the WAMP open-source web application software stack (and other "AMP" stacks). MySQL is written in C and C++. MySQL works on many system platforms, including Linux, macOS, Microsoft Windows, etc.

MySQL performs extremely well in the average case, the developer interfaces are there, and the documentation is very, very good. It has also been tested to be a fast, stable and true multi-user, multi-threaded SQL database server.

4.2 Discussion of code segments

This section includes the segments of code used to provide various user functionalities.

4.2.1 Code segment to insert a Blog post

Figure 4.1 shows the code segment to insert a Blog post into the database.

```
else
{
//$blogdate = test_input($_POST["blogdate"]);
$mydate=date("Y:m:d H:i:s");
mysqli_select_db($con,'socntwk')
or die("Unable to select database: " . mysqli_error($con));
if (!$con) {mysqli_fatal_error($con,"Invalid username or password");}
$sql1="INSERT INTO socntwk.user_messages VALUES
(NULL,'$uname','$msgtext','$mydate','$msgtype','$blogtitle')";
$result1 = mysqli_query($con,$sql1) or
die(mysqli_fatal_error($con,"Could not insert the record"));
mysqli_close($con);
$msgtypeflag = $msgtextflag =false;
if(!isset($_SESSION)) session_start();
$error="$uname's Message has been successfuly posted";
$_SESSION['error']=$error;
header('Location:postmessage.php');
exit();
}
} // successfuly posted
```

Figure 4.1 Code segment to insert a Blog post

4.2.2 Code segment to drop a Follower

Figure 4.2 shows the code segment to delete an entry from the list of Followers of a blogger in the database.

```
if (!$loggedin) {
header('Location:login.php');
exit();
};
echo "<div class='mycontentbox'>";
if (isset($_GET['add'])){
$add=sanitizeString($con,$_GET['add']);
if (!mysqli_num_rows(queryMysql($con,"SELECT * FROM socntwk.USER_FRIENDS
WHERE user_name='$add' AND friend_name='$uname'")))
queryMysql($con,"INSERT INTO socntwk.USER_FRIENDS VALUES(null,'$add', '$uname')");
}
elseif (isset($_GET['remove']))
{
$remove = sanitizeString($con,$_GET['remove']);
queryMysql($con,"DELETE FROM socntwk.USER_FRIENDS
WHERE user_name='$remove' AND friend_name='$uname'");
}
$result = queryMysql($con,"SELECT * FROM socntwk.user_profile
where user_verified='Y' ORDER BY user_name");
$num = mysqli_num_rows($result);
```

Figure 4.2 Code segment to drop a Follower

4.2.3 Code segment to retrieve list of Top Bloggers

Figure 4.3 shows the code segment used to retrieve and display a list of Top Bloggers from the database.

```php
<?php
echo "<aside class='asidebox'>";
$sql1=queryMysql($con,"select count(b.user_message) topbloggers,a.use
$numtop = mysqli_num_rows($sql1);
echo  "<fieldset><legend>Top bloggers</legend>";
for ($k=0 ; $k < $numtop; ++$k)
{
$result1 = mysqli_fetch_row($sql1);
$blogcount=$result1[0];
$bloguname=$result1[1];
$blogfname=$result1[2];
$bloglname=$result1[3];
echo "<p>" .$blogfname ."" .$bloglname ."(".$blogcount    .")</p>";
}
echo  "</fieldset></aside>";
?>
```

Figure 4.3 Code segment to retrieve list of Top Bloggers

4.2.4 Code segment to update User Profile

Figure 4.4 shows the code segment to update the details of a user's profile in the database.

```php
/*****************************************************/
if($addressflag and $emailflag and $phoneflag and $mobileflag and $dobflag and $lnameflag
{ // all data are in valid format
//echo "all flags are true";
mysqli_select_db($con,'socntwk')  or die("Unable to select database: " . mysqli_error($cor
if (!$con) {mysqli_fatal_error($con,"Invalid username or password");}
$sql1="UPDATE socntwk.user_profile
SET user_email='$email',user_mobile=$mobile,user_phone=$phone,user_address='$address'
WHERE
user_name='$uname' AND user_fname='$fname' AND user_lname='$lname'  AND
user_dob=STR_TO_DATE('$dob','%d/%m/%Y')";
$result1=mysqli_query($con,$sql1) or die("Unable to update the database: " . mysqli_error
$unameflag = $fnameflag =$lnameflag = $pswflag = $cpwflag = $dobflag = $phoneflag = $mobil
if(!isset($_SESSION)) session_start();
header('Location:editprofile.php');
exit();
}
// profile updated successfully
```

Figure 4.4 Code segment to update User Profile

4.3 Applications of project work

- Personal Diary
- Corporate Newsletter
- Informal Engagements
- Rich Media Portfolios
- Building a Community

Chapter 5

TESTING AND RESULTS

5.1 Testing

Software testing is conducted to provide stakeholders with information about the quality of the software product or service under test. Software testing can also provide an objective, independent view of the software to allow the business to appreciate and understand the risks of software implementation.

5.1.1 Unit testing

Unit testing is a software testing method by which individual units of source code, sets of one or more computer program modules together with associated control data, usage procedures, and operating procedures, are tested to determine whether they are fit for use.

5.1.1.1 Unit test case 1

This test case checks to see if signed up users are able to login or not, as shown in Table 5.1 and Figure 5.1.

Table 5.1 Unit test case for Login check

Sl No. of test case:	1
Name of test:	Login check
Item / Feature being tested:	Login
Sample Input:	Enter username and password fields and click on Submit button.
Expected output:	Message ‘(username), you are logged in’ is displayed.
Actual output:	Message ‘(username), you are logged in’ is displayed.
Remarks:	Test succeeded

Figure 5.1 Unit test case for Login check

5.1.2 Integration testing

Integration testing is the phase in software testing in which individual software modules are combined and tested as a group. It occurs after unit testing and before validation testing

5.1.2.1 Integration test case 1

This test case checks to see if the Webcam is properly configured and working, as shown in Table 5.2, and in Figure 5.2 and 5.3.

Table 5.2 Integration test case for Webcam check

Sl No. of test case:	1
Name of test:	Webcam check
Item / Feature being tested:	Webcam
Sample Input:	Click on ‘Webcam’ option and take a snapshot
Expected output:	Snapshot is uploaded to profile
Actual output:	Snapshot is uploaded to profile
Remarks:	Test succeeded

Figure 5.2 Snapshot with Webcam

Figure 5.3 Profile updated

5.1.2.2 Integration test case 2

This test case checks to see if following another user, updates the list of users being followed by blogger, as shown in Table 5.3, and in Figure 5.4 and 5.5.

Table 5.3 Integration test case for Following check

Sl No. of test case:	2
Name of test:	Following check
Item / Feature being tested:	Follow option
Sample Input:	Click on Follow option beside a user's name.
Expected output:	'You are following' list is updated with new user being followed.
Actual output:	New user being followed is added to 'You are following' list.
Remarks:	Test succeeded.

Figure 5.4 Following option clicked

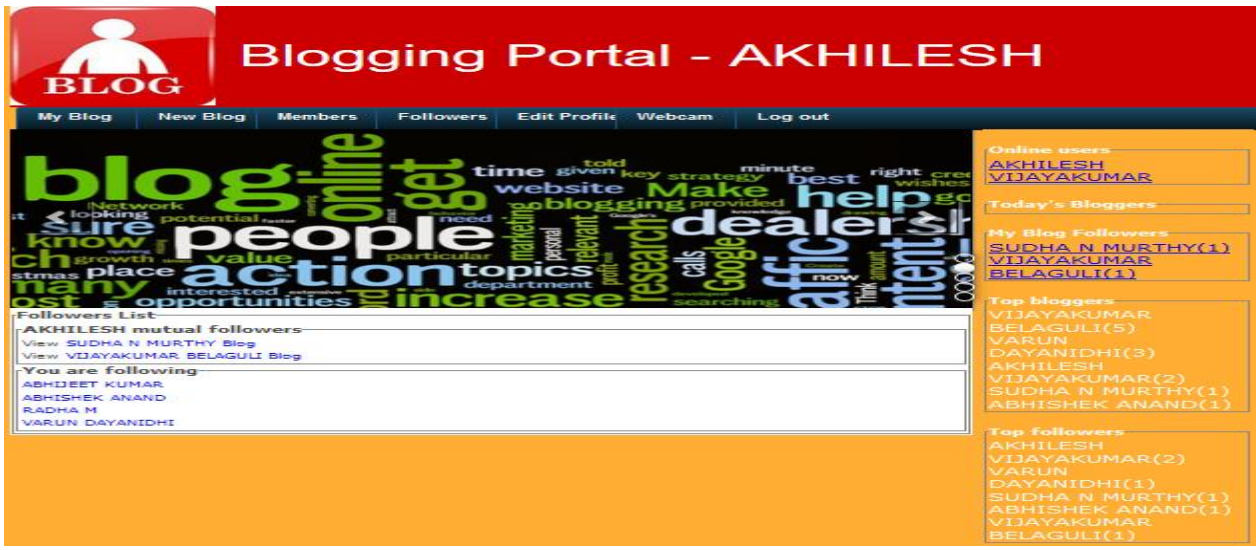

Figure 5.5 'You are Following' list updated

5.1.3 System testing

System testing is testing conducted on a complete integrated system to evaluate the system's compliance with its specified requirements.

5.1.3.1 System test case 1

This test case checks to see if statistics of the site is displayed or not. This checks to see if the lists showing the Online users, Today's Bloggers, Top bloggers and Top threads are displayed correctly or not, as shown in Table 5.4 and Figure 5.6.

Table 5.4 System test case for Site statistics

Sl No. of test case:	1
Name of test:	Site statistics test.
Items / Features being tested:	Online users, Today's Bloggers, Top bloggers and Top threads lists.
Sample Input:	Login to valid user account.
Expected output:	Online users, Today's Bloggers, Top bloggers and Top threads lists display current statistics of site.
Actual output:	Current statistics of site displayed in Online users, Today's Bloggers, Top bloggers and Top threads lists.
Remarks:	Test succeeded.

Figure 5.6 System test case for Site statistics

5.2 Discussion of results

The outcomes of test results for a variety of user interactions with the application are discussed in the following sections of the chapter.

5.2.1 Home page

Figure 5.7 shows the Home page of the Blogging Portal displayed before users login.

Figure 5.7 Home page

5.2.2 Sign up page

Figure 5.8 shows the page of the Blogging Portal where users sign up (registers) by giving profile details.

Figure 5.8 Sign up page

5.2.3 Login page

Figure 5.9 shows the Login page of the Blogging Portal displayed for users to login.

Figure 5.9 Login page

5.2.4 My Blog page

Figure 5.10 shows the page of the Blogging Portal where a blogger can view all his previous posts.

Figure 5.10 My Blog page

5.2.5 New Blogpost page

Figure 5.11 shows the page of the Blogging Portal to type and submit a new and titled blogpost.

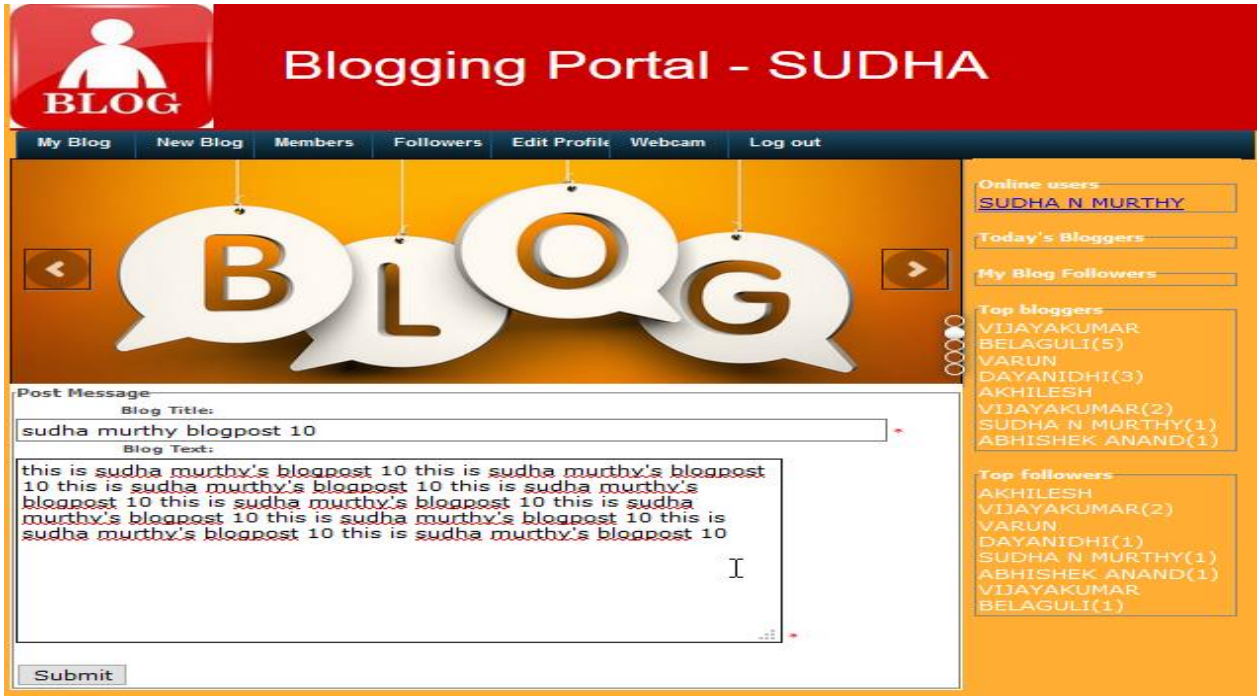

Figure 5.11 New Blogpost page

5.2.6 Members page

Figure 5.12 shows the page of the Blogging Portal where all other members (registered users) of the system can be viewed by a blogger.

Figure 5.12 Members page

5.2.7 Followers page

Figure 5.13 shows the page of the Blogging Portal where the followers of a blogger and users being followed by the blogger can be viewed.

Figure 5.13 Followers page

5.2.8 Edit Profile page

Figure 5.14 shows the page of the Blogging Portal used to edit a user's profile details.

Figure 5.14 Edit Profile page

5.2.9 Webcam page

Figure 5.15 shows the page of the Blogging Portal for taking a snapshot using webcam, for a user's profile picture.

Figure 5.15 Webcam page

Chapter 6

INSTALLATION INSTRUCTIONS

This chapter provides step-by-step instructions to install the WAMP Server required to run PHP applications in a Windows environment.

6.1 Installing WAMP Server

- To start the installation process, you need to open the folder where you saved the file, and **double-click the installer file**. A security warning window will open, asking if you are sure you want to run this file. **Click Run** to start the installation process.
- Next you will see the Welcome To The WampServer Setup Wizard screen. **Click Next** to continue the installation.
- The next screen you are presented with is the License Agreement. Read the agreement, check the radio button next to **I accept the agreement**, then **click Next** to continue the installation.
- Next you will see the Select Destination Location screen. Unless you would like to install WampServer on another drive, you should not need to change anything. **Click Next** to continue.
- The next screen you are presented with is the Select Additional Tasks screen. You will be able to select whether you would like a Quick Launch icon added to the taskbar or a Desktop icon created once installation is complete. Make your selections, then **click Next** to continue.
- Next you will see the Ready To Install screen. You can review your setup choices, and change any of them by **clicking Back** to the appropriate screen, if you choose to. Once you have reviewed your choices, **click Install** to continue.
- WampServer will begin extracting files to the location you selected.
- Once the files are extracted, you will be asked to select your default browser. WampServer defaults to Internet Explorer upon opening the local file browser window. If your default browser isn't IE, then look in the following locations for the corresponding .exe file:

    - **Opera:** C:\Program Files (x86)\Opera\opera.exe
    - **Firefox:** C:\Program Files (x86)\Mozille Firefox\firefox.exe
    - **Safari:** C:\Program Files (x86)\Safari\safari.exe
    - **Chrome:** C:\Users\xxxxx\AppData\Local\Google\Chrome\Application\chrome.exe

- Select your default browser's .exe file, then **click Open** to continue.
- A Windows Security Alert window will open, saying that Windows Firewall has blocked some features of the program. Check whether you want to allow Apache HTTP Server to communicate on a private or public network, then **click Allow Access**.
- The Setup screen will appear next, showing you the status of the installation process.
- Once the progress bar is completely green, the PHP Mail Parameters screen will appear. Leave the SMTP server as **localhost**, and change the email address to one of your choosing. **Click Next** to continue.
- The Installation Complete screen will now appear. **Check the Launch WampServer Now** box, then **click Finish** to complete the installation.
- You should see the WampServer icon appear in the systray on the right side of your taskbar. If the icon is green, then everything is working properly. If the icon is orange, then there are issues with one of the services. If the icon is red, then both Apache and MySQL services aren't running. You will need to resolve those issues before continuing.

Chapter 7

CONCLUSION AND FUTURE ENHANCEMENTS

The Blogging System focuses on providing the users the ability to post blogs. The Admin configures the database with critical data prior to the use of the Blogging System application. The system is tested and re-tested with varying constraints to ensure its effectiveness and provide error free functionality to the end user. It assures saving of time in performing maintenance operations in case of blogging.

Hence, the Blogging System serves as an interesting tool that caters to all user's requirements when they are blogging and provides accurate and up-to-date statistics about the system's users.

- The option to edit and delete posts can be made available to bloggers.
- Audio and video blogging features can be added.

REFERENCES

1. Raghu Ramakrishnan and Johannes Gehrke , Database Management Systems , McGRAW HILL , 3rd Edition.
2. Ramez Elmasri and Shamkant B. Navathe , Fundamentals of Database Systems , Pearson , 7th Edition
3. Randy Connolly, Ricardo Hoar, **"Fundamentals of Web Development",** 1stEdition, Pearson Education India.
4. Robin Nixon, **"Learning PHP, MySQL &JavaScript with jQuery, CSS and HTML5",** 4thEdition, O'Reilly Publications, 2015.
5. Luke Welling, Laura Thomson, **"PHP and MySQL Web Development",** 5th Edition, Pearson Education, 2016.
6. Nicholas C Zakas, **"Professional JavaScript for Web Developers",** 3rd Edition, Wrox/Wiley India, 2012.
7. www.php.net
8. www.javascript.com
9. www.w3schools.com

ABBREVIATIONS

CSS	Cascading Style Sheets
DFD	Data Flow Diagram
HTML5	HyperText Markup Langauge version 5
HTTP	HyperText Transfer Protocol
OS	Operating System
PHP	Hypertext Preprocessor
SRS	Software Requirements Specification
SQL	Structured Query Language
UCD	Use Case Diagram
WAMP	Windows, Apache, MySQL and PHP

ABSTRACT

Interest in Blogging has increased in recent years with Twitter, etc. The Blogging System provides a website for users to post textual content. The posts are presented in reverse chronological order. The user must first create an account (or Sign Up), which involves creating a login username and password. Creating a User Profile involves filling of a form, with basic information that will make up the Profile. A Webcam option is provided to capture on-spot pictures, that can be used as Profile Pictures in a user's profile. At any time, a user can edit his/her profile.

Users can give titles for their posts. Other account holders can 'comment' on people's blog posts only if they 'follow' them too. A user can see the usernames of the others who are currently online, as well as, of those who have blogged that day. Every blogger is provided with a list of 'blog threads', indicating the number of users who have commented on each of their posts. Each user is also provided with the list of top 5 blog threads, which are the posts with most user comments. The website also maintains a ranking of top 5 bloggers, which will be those users with the most posts.

ACKNOWLEDGMENT

The fulfillment and rapture that go with the fruitful finishing of any assignment would be inadequate without the specifying the people who made it conceivable, whose steady direction and support delegated the endeavors with success.

I would like to profoundly thank **Management** of **RNS Institute of Technology** for providing such a healthy environment to carry out this Project work.

I would like to thank our beloved Director **Dr. H N Shivashankar** for his confidence feeling words and support for providing facilities throughout the course.

I would like to express my thanks to our Principal **Dr. M K Venkatesha** for his support and inspired me towards the attainment of knowledge.

I wish to place on record my words of gratitude to **Dr. M V Sudhamani,** Professor and Head of the Department, Information Science and Engineering, for being the enzyme and master mind behind my Project work.

I would like to express my profound and cordial gratitude to my Coordinator **Mr. Santosh Kumar**, Assistant Professor, Department of Information Science and Engineering for his valuable guidance, constructive comments and continuous encouragement throughout the Project work.

I would like to express my profound and cordial gratitude to my Faculty Incharge **Mrs. Asha M V**, Assistant Professor, Department of Information Science and Engineering for her valuable guidance in preparing Project report

I would like to thank all other teaching and non-teaching staff of Information Science & Engineering who have directly or indirectly helped me to carry out the project work.

And lastly, I would hereby acknowledge and thank my parents who have been a source of inspiration and also instrumental in carrying out this Project work.

AKHILESH. V

USN: 1RN15IS009