

## **CHAPTER 1**

# **INTRODUCTION**

Zoo database system is a Database application which is helpful for visitors of zoo to know about the animal details and workers details. And user can search the animal by specifying the nickname. This mini-Project is implemented using HTML,CSS and PHP.

Operations supported by the application are insert, delete, update and retrieve. Admin has rights to insert, update and delete the values of table. An user just view the contents of the table and he can search for the animals by giving the nickname.

In the following sections, a brief introduction about the tools, languages and the databases used to develop the project are discussed.

### **1.1 HTML**

HTML, which stands for Hyper Text Mark-Up Language, is the language for describing structured documents as well as the language used to create web pages in the Internet. The language is based on an existing, international formatting standard SGML, Standard Generalized Mark-Up Language, which is used for text processing.

HTML documents are nothing but web pages which contains HTML tags and plain text. The purpose of a web browser is to read HTML documents and display them as web pages. The browser does not display the HTML tags, but uses the tags to interpret the content of the page.

#### **History**

HTML, which stands for Hyper Text Mark-Up Language, is the language for describing structured documents as well as the language used to create web pages in the Internet. The language is based on an existing, international formatting standard SGML, Standard Generalized Mark-Up Language, which is used for text processing. HTML is a simplified version of SGML.

Tools in order to use HTML

Tools help us in process of creating HTML document. Some are as follows

- **TEXT EDITOR:** To create the HTML code we require a text editor or a word processor. Such as, Notepad, WordPad. We are using notepad++ in developing this project.
- **WEB BROWSER:** The code created by an editor should be executed. This operation can be performed with help of a web browser. Such as Internet Explorer, Netscape navigator, Mozilla Firefox etc.
- **GRAPHICS SOFTWARE:** To include picture we require a graphic software like Adobe Photoshop.
- **Web server:** To make the document is to be available on the internet then, we will have to host it on a web server.

### **1.1.1 Significant Language Features**

HTML files are written in ASCII text, so the user can use any text editor to create his/her web page, though a browser of one sort or another is necessary to view the web page. HTML is case insensitive with its language commands. The characters within the document, however, are case sensitive. The language consists of various "tags" which are known as elements. These allow the browser to understand (and put into the desired/specified format) the layout, background, headings, titles, lists, text and/or graphics on the page. The elements are classified according to their function in the HTML document. There are head elements and body elements. The head elements identify properties of the entire document, while body elements actually mark text as content and show a change in the appearance in one way or another. Most elements have a beginning and an ending which encompass the text the user wishes to mark with the tag. All HTML documents must begin with the element and end with the element. Some of the other elements which may be used are tags to create lists--both ordered lists as well as unordered lists. The user may also create larger or smaller, bolder, italicized, or underlined text. Attributes may be used along with the elements. These perform functions such as placement of text, indication of the source files of images, and identification of links to the document or part of the document.

### 1.1.2 HTML Code

Copy and paste the following HTML code into your newly open text file. Which just displays hello world..

```
<html>
<header><title>This is title</title></header>
<body>
  This is sample text...
  <!-- We use this syntax to write comments -->
  <!-- Page content and rest of the tags here.... -->
  <!-- This is the actual area that gets shown in the browser →
Hello world
</body>
</html>
```

### 1.1.3 HTML TAGS

HTML tags are keywords surrounded by angle brackets like <html>. These are in pair format such that every first tag in pair is start tag where as second tag is end tag. These start and end tags are also called as opening tags and closing tags respectively.

### Tags Used In Project

The HTML tags are the basis, in order to do this Project. By using some of the important and basically taught tags are used in this Project. Here are some of the tags used in making the Project called AUTOMOBILE MANAGEMENT SYSTEM.

#### HTML Attributes

Attributes provide additional information about HTML elements.

- HTML elements can have **attributes**
- Attributes provide **additional information** about an element
- Attributes are always specified in **the start tag**
- Attributes come in name/value pairs like: **name="value"**

Some basic text formatting HTML tags are listed:

Tag	Description
<html>	Defines an HTML document
<body>	Defines the document's body
<h1> to <h6>	Defines header 1 to header 6
<p>	Defines a paragraph
 	Inserts a single line break
<b>	Defines bold text
<!-->	Defines a comment
<small>	Defines small text

Some of the HTML tags used to create a table are listed:

In an HTML file we can create tables with the Table tags, which in turn will render the browser to display the table in the web page.

Tag	Description
<table>	Defines a table
<th>	Defines a table header
<tr>	Defines a table row
<td>	Defines a table cell
<tbody>	Defines a table body
<tfoot>	Defines a table footer

## A Simple Form

A form in a web page allows the users to input various data online. In an HTML document; forms can be created with the Form tags. In the following table, some basic Form tags are listed:

Tag	Description
<form>	Defines a form for user input
<input>	Defines an input field
<textarea>	Defines a text-area
<label>	Defines a label to a control
<fieldset>	Defines a fieldset
<legend>	Defines a caption for a fieldset
<select>	Defines a selectable list
<optgroup>	Defines an option group
<option>	Defines an option in the drop box
<button>	Defines a push button

### Image Tags

In an HTML document we can insert and display images by using the image tags.

In the following table, some basic Image tags are listed:

Tag	Description
<IMG SRC attributes>	Defines an image

The “src” attribute is used to display an image on a web page. “src” stands for “source”, and its value is the url of the image to be displayed on the page. The url indicates the location where the image is stored. Attributes may be height, width, align so on.

### Background colour

Using bgcolor attribute this can be done. This is body tag attribute. Six digit hexadecimal code represent the colours.

Syntax: <body text=”text\_color” bgcolor = ”background\_color”>

**Anchor tag**

Anchor tag is used to link two or more different web pages.

Ex: `<a href="next page to be opened">click here</a>` where href stands for hyper link reference.

**Areas of Application**

HTML only has one area of application at this time and that is the development of web pages. However, not all browsers support all the tags in all versions of HTML. Because of this, it is wise not to design your web page for a specific browser, because what may look fantastic on your browser has no guarantee of looking great on someone else's browser.

## 1.2 PHP

PHP is a general-purpose scripting language that is especially suited to server-side web development, in which case PHP generally runs on a web server. Any PHP code in a requested file is executed by the PHP runtime, usually to create dynamic web page content or dynamic images used on websites or elsewhere.

PHP originally stood for Personal Home Page, but it now stands for the recursive backronym PHP. Hypertext Pre-processor. PHP code may be embedded into HTML code, or it can be used in combination with various web template systems, web content management system and web frameworks.

**PHP developer**

PHP developers develop programs, applications, and web sites using the dynamic scripting language PHP. PHP is known for web development and business applications. Depending on job function, PHP developers may be classified as software developers or web developers.

**Tags Description**

`<?php` to open PHP section

`?>` to close PHP sections

`ECHO` prints the lines

## 1.3 DATABASE

A database is a collection of information that is organized so that it can easily be accessed, managed, and updated. In one view, databases can be classified according to types of content: bibliographic, full-text, numeric, and images. **Database** software systems are programmed in SQL, and examples include Microsoft SQL Server, MySQL, Oracle SAP HANA and FoxPro.

A DBMS system is also required to protect the integrity of data and provide its security. A database management system (**DBMS**) is system software for creating and managing databases. The **DBMS** provides users and programmers with a systematic way to create, retrieve, update and manage data.

## 1.4 MYSQL

MySQL is a powerful database. It's very good and free of charge. Many developers in the world selected mysql and php for developing their website.

The MySQL database has become the world's most popular open source database because of its consistent fast performance, high reliability and ease of use. It's used in more than 6 million installations ranging from large corporations to specialized embedded applications on every continent in the world. (Yes, even Antarctica!)

Not only is MySQL the world's most popular open source database, it's also become the database of choice for a new generation of applications built on the LAMP stack (Linux, Apache, MySQL, PHP / Perl / Python.) MySQL runs on more than 20 platforms including Linux, Windows, OS/X, HP-UX, AIX, Netware, giving you the kind of flexibility that puts you in control.

Whether you're new to database technology or an experienced developer or DBA, MySQL offers a comprehensive range of certified software, support, training and consulting to make you successful.

## **1.5 WAMP**

The acronym WAMP refers to a set of free (open source) applications, combined with Microsoft Windows, which are commonly used in Web server environments. The WAMP stack provides developers with the four key elements of a Web server: an operating system, database, Web server and Web scripting software. The combined usage of these programs is called a server stack. In this stack, Microsoft Windows is the operating system (OS), Apache is the Web server, MySQL handles the database components, while PHP, Python, or PERL represents the dynamic scripting languages.

## **1.6 Notepad++**

Notepad++ is a text editor and source code editor for use with Microsoft Windows. Unlike Microsoft Notepad, the built-in Windows text editor, it supports tabbed editing, which allows working with multiple open files in a single window. The project's name comes from the C increment operator.

Notepad++ is distributed as free software. At first the project was hosted on SourceForge.net, from where it has been downloaded over 28 million times.

## **1.7 Web Browser**

Google Chrome is a free web browser from Google which we are using here. With its clean design and advanced features, Chrome has quickly become one of the most popular web browsers worldwide. In this lesson, we'll talk about the features of Google Chrome, how to download and install Chrome to your computer, and how to sign in to Chrome using a Google account.



## **CHAPTER 2**

### **System Analysis and Design**

In this chapter, a complete description of the project development is discussed. The requirements of the project identified are showcased. The database design is done Using High-Level Conceptual Data Models

#### **2.1 Requirement Analysis**

Following requirements were identified during the requirement collection and analysis.

1. Admin can login by by giving appropriate login id and password.
2. After he logged in he can edit or update details.
3. After he logged in he can delete details.
4. After he logged in he can retrieve details.
5. After he logged in he can add details.
6. After he logged in he can search the animal details by giving nickname.
7. Normal user can view the details of all tables.
8. User can search the animal details by giving nickname.

#### **Feasibility Study**

The feasibility study carried out showed that the requirements that were to be included could be provided by the use of RDBMS software such as MySQL which is available as an open source and for the front end HTML pages with processing capability provided by the Scripting language such as PHP and Javascript.

## 2.2 ER-Diagram

Following is the conceptual representation of the requirements identified as an ER-Diagram.

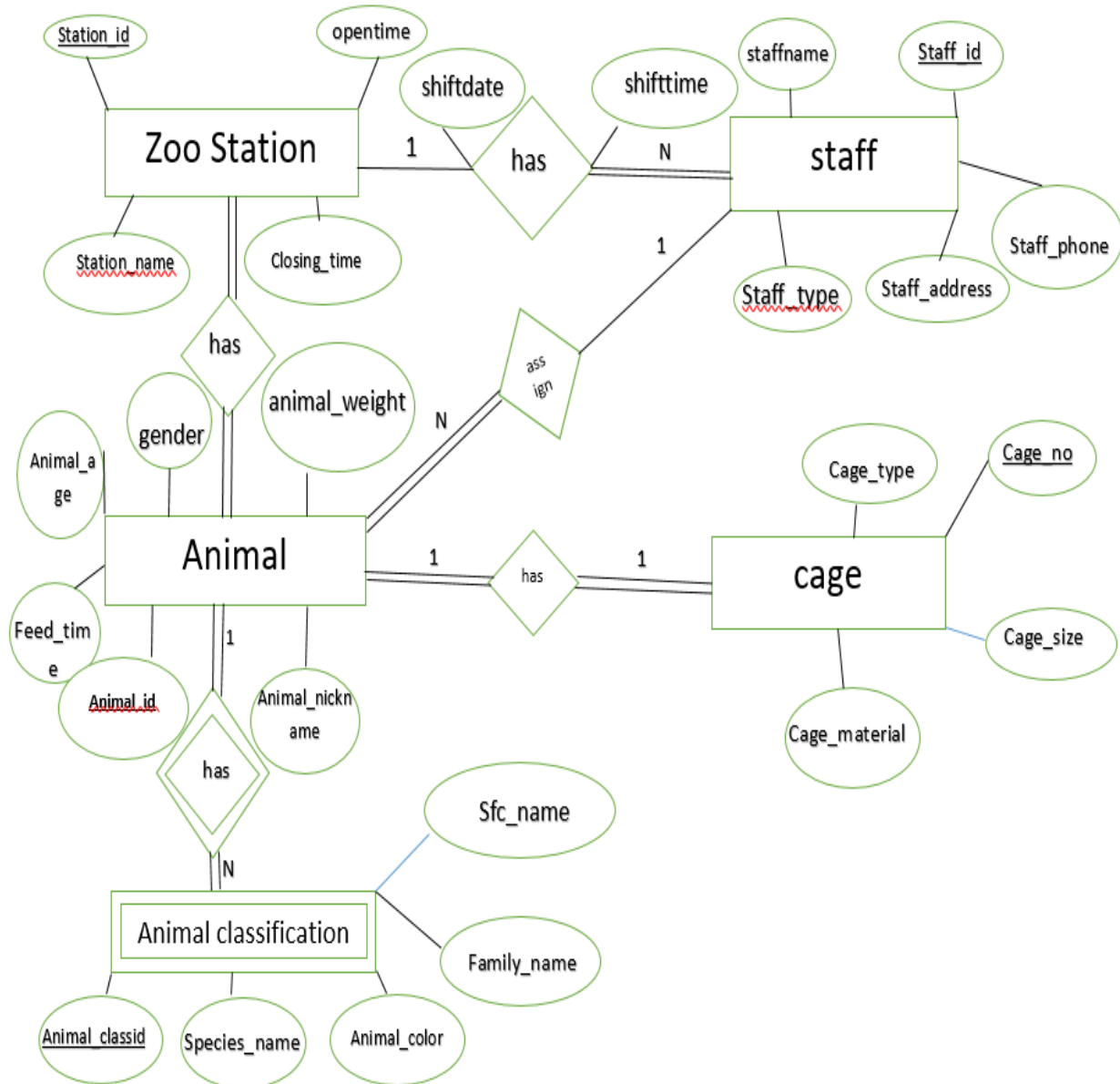
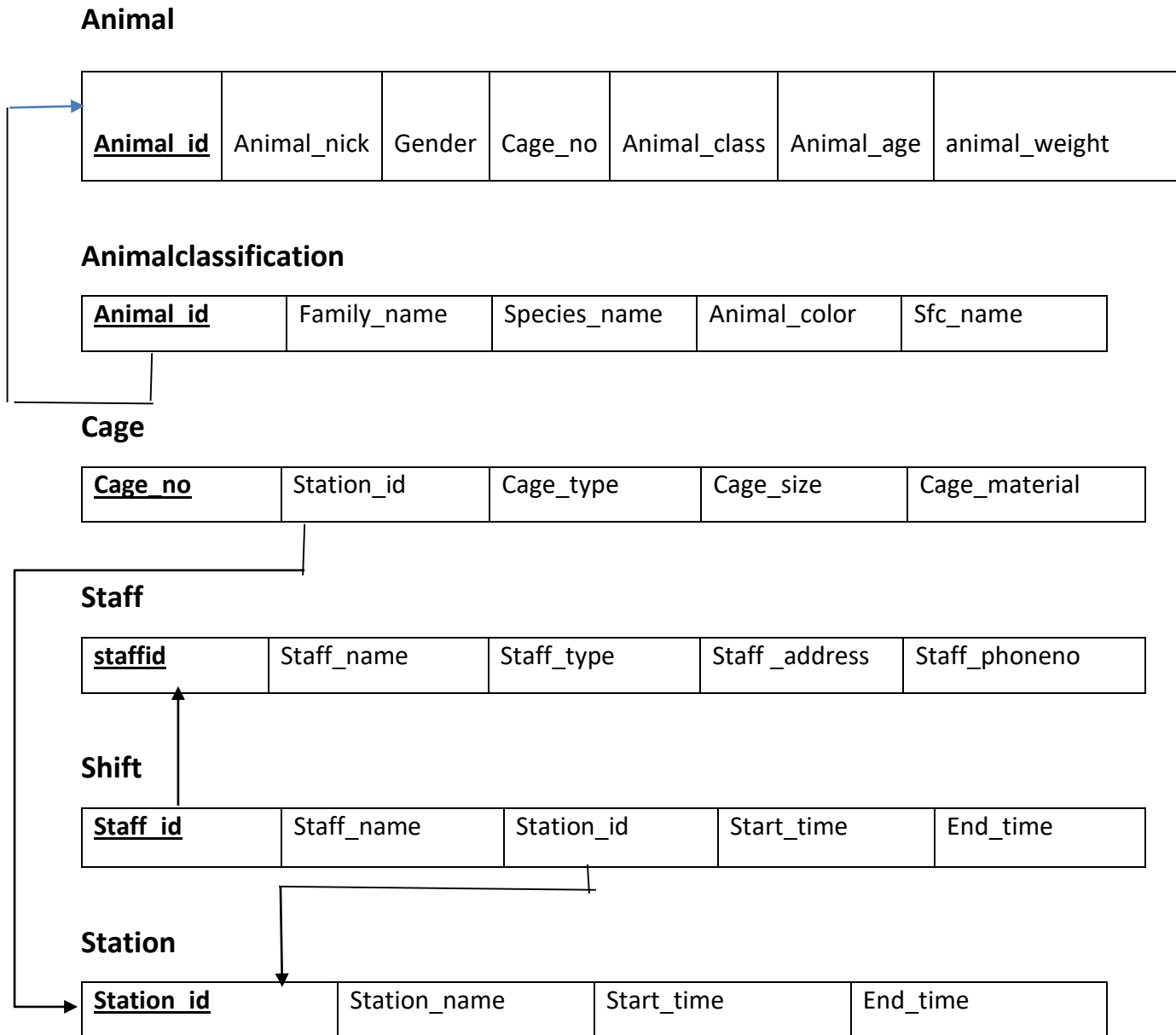


Figure 2.1 ER Diagram for Zoo database

## 2.3 Relational Schema

The relational schema diagram has been derived from the ER-Diagram in Figure 2.1 using the ER-Relational mapping algorithm



**Figure 2.2 Relational Schema Diagram Showing the Primary key and Foreign key relationships**

## 2.4 Functional Requirements

Functional requirements of a software project interpret the function of a part. It defines its functions, input and output. The typical functional requirements include:

Application contains 2 modules:

- Admin module
- User module

Admin module

- Admin can add details to table.
- Admin can update the details of a table.
- Admin can delete details of table.
- Admin can search an animal by giving the nickname.

User module

- Customer can view the contents of table.
- Customer can search the animals by giving the animal nickname.

## 2.5 Non- Functional Requirements

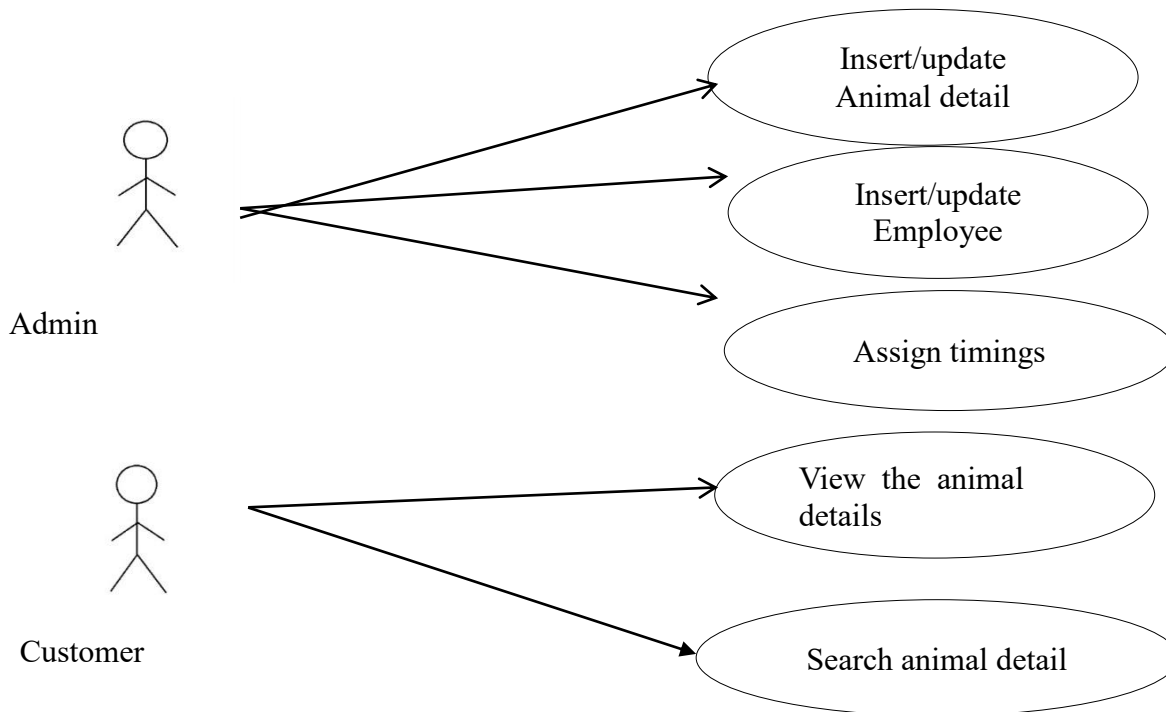
A non-functional requirement specifies the canon of the articular process not the particular judgment of the system and particular behavior of the process. Non-functional requirements define how the system works.

- This application is developed to make viewing animal information easier to customer so that it will save the time.
- This application work efficiently it works on all logical paths and independently and it should use the mobile data efficiently.
- This application is helps during zoo visiting.
- This application reduces all the complexity during viewing and searching.
- To run this application efficiently mobile network is the main important factor.

- The system should be capable to enhance with further technology in future to improve its features compared to the existing system .
- The system should be reliable and it should be related in all the condition and it should be recoverable in all the situation or condition if error occurs.

## 2.6 Use Case Diagram

The use case diagrams usually refer to behavioral diagrams helps people to understand the interaction between user and system. Use case diagram identify different users of the system. It is used to define some set of actions, which is called as use cases. Actors are the result of some valuable use cases. Use case figures are also called as unified modeling language.



**Figure 2.3 Use Case Diagram**

## CHAPTER 3

### SYSTEM IMPLEMENTATION

This chapter gives brief introduction about the project implementation first the database design, database connectivity is discussed and followed by the implementation of database operation.

#### 3.1 Database Design

-- Table structure for table `Animals`--

```
CREATE TABLE `Animals` (  
  `Cage_No` int(11) NOT NULL,  
  `Animal_ID` varchar(6) NOT NULL,  
  `Animal_Nick` varchar(30) NOT NULL,  
  `Gender` varchar(1) NOT NULL,  
  `Animal_ClassID` varchar(6) NOT NULL,  
  `Animal_age` int(3) NOT NULL,  
  `Animal_weight` int(4) NOT NULL  
);
```

-- Table structure for table `Animal class`--

```
CREATE TABLE `Animal_Class` (  
  `Animal_ClassID` varchar(6) NOT NULL,  
  `Family_Name` varchar(30) NOT NULL,  
  `Species_Name` varchar(30) NOT NULL,  
  `Animal_color` varchar(15) NOT NULL,  
  `Sfc_name` varchar(30) NOT NULL  
);
```

-- Table structure for table `Cage`--

```
CREATE TABLE `Cage` (  
  `Station_ID` varchar(30) NOT NULL,  
  `Cage_No` int(11) NOT NULL,  
  `Cage_Type` varchar(30) NOT NULL,  
  `Cage_size` varchar(15) NOT NULL,  
  `Cage_material` varchar(15) Not NULL  
);
```

-- Table structure for table `Shift\_Assignment`--

```
CREATE TABLE `Shift_Assignment` (  
  `Shift_ID` int(11) NOT NULL,  
  `Staff_ID` varchar(6) NOT NULL,  
  `Station_ID` varchar(6) NOT NULL,  
  `StartTime` int(11) NOT NULL,  
  `EndTime` int(11) NOT NULL  
);
```

-- Table structure for table `Staff`--

```
CREATE TABLE `Staff` (  
  `Staff_ID` varchar(6) NOT NULL,  
  `Staff_Name` varchar(30) NOT NULL,  
  `Staff_Type` varchar(30) NOT NULL,  
  `Staff_adress` Varchar(30) NOT NULL,  
  `Staff_pno` Varchar(15) NOT NULL  
);
```

--Table structure for table `Station`--

```
CREATE TABLE `Station` (  
  `Station_ID` varchar(6) NOT NULL,  
  `Station_Name` varchar(30) NOT NULL,  
  `Start_time` Varchar(10) NOT NULL,  
  `End_time` Varchar(10) NOT NULL  
);
```

## 3.2 Database Connectivity

```
<?php  
  $host = "localhost";  
  $user = "root";  
  $pass = "";  
  $db = "Test1";  
  $dbconn = mysqli_connect($host,$user,$pass,$db) or die("Could not connect to  
database!");  
  
  mysqli_select_db($dbconn,'Staff');  
  
  $query="SELECT * FROM Staff";  
  $result=mysqli_query($dbconn,$query);  
  $rownum=mysqli_num_rows($result);  
  ?>
```



### 3.3 Implementation of Database Operations

#### INSERT

Following code snippets shows Insertion operation done in Cage table.

```
$add="INSERT INTO `Animals` (`Cage_No`, `Animal_ID`, `Animal_Nick`, `Gender`,  
`Animal_ClassID`, `Animal_age`, `Animal_weight`) VALUES ('$Cage_No', '$Animal_ID',  
'$Animal_Nick', '$Gender', '$Animal_ClassID', '$Animal_age', '$Animal_weight');  
mysqli_query($dbconn,$add);  
$affectedrows = mysqli_affected_rows($dbconn);  
if($affectedrows==1){  
header("Location:animal.php");
```

#### UPDATE

Following code snippets shows Updation operation done in Cage table.

```
$update_Query = "UPDATE Animals SET Cage_No='$Cage_No',  
Animal_Nick='$Animal_Nick', Gender='$Gender', Cage_No='$Cage_No',  
Animal_ClassID='$Animal_ClassID',  
Animal_age='$Animal_age',Animal_weight='$Animal_weight' WHERE  
Animal_ID='$Animal_ID';  
mysqli_query($dbconn,$update_Query);  
$affectedrows = mysqli_affected_rows($dbconn);  
header("Location:animal.php");  
}else {  
$edit_Animal_ID = $_GET["animalid"];  
$edit_Query="SELECT * FROM Animals WHERE Animal_ID='$edit_Animal_ID';  
$edit_Pass_Query = mysqli_query($dbconn, $edit_Query);  
$edit_Results = mysqli_fetch_assoc($edit_Pass_Query);  
}?>
```

## DELETE

Following code snippets shows Deletion operation done in Cage table.

```
<html>
<body>
<?php
$dbconn = mysqli_connect("localhost","root","","Test1");
mysqli_select_db($dbconn,'Animals');
$Animal_ID=$_GET["animalid"];
mysqli_query($dbconn,"DELETE FROM Animals WHERE Animal_ID='$Animal_ID'");
$affectedrows = mysqli_affected_rows($dbconn);
if($affectedrows==1){
header("Location:animal.php");
}
?>
</body>
</html>
```

## RETRIEVAL

Following code snippets shows Select operation done in Cage table.

```
<html>
<body>
<?php
$host = "localhost";
$user = "root";
$pass = "";
$db = "Test1";
$dbconn = mysqli_connect($host,$user,$pass,$db)
```

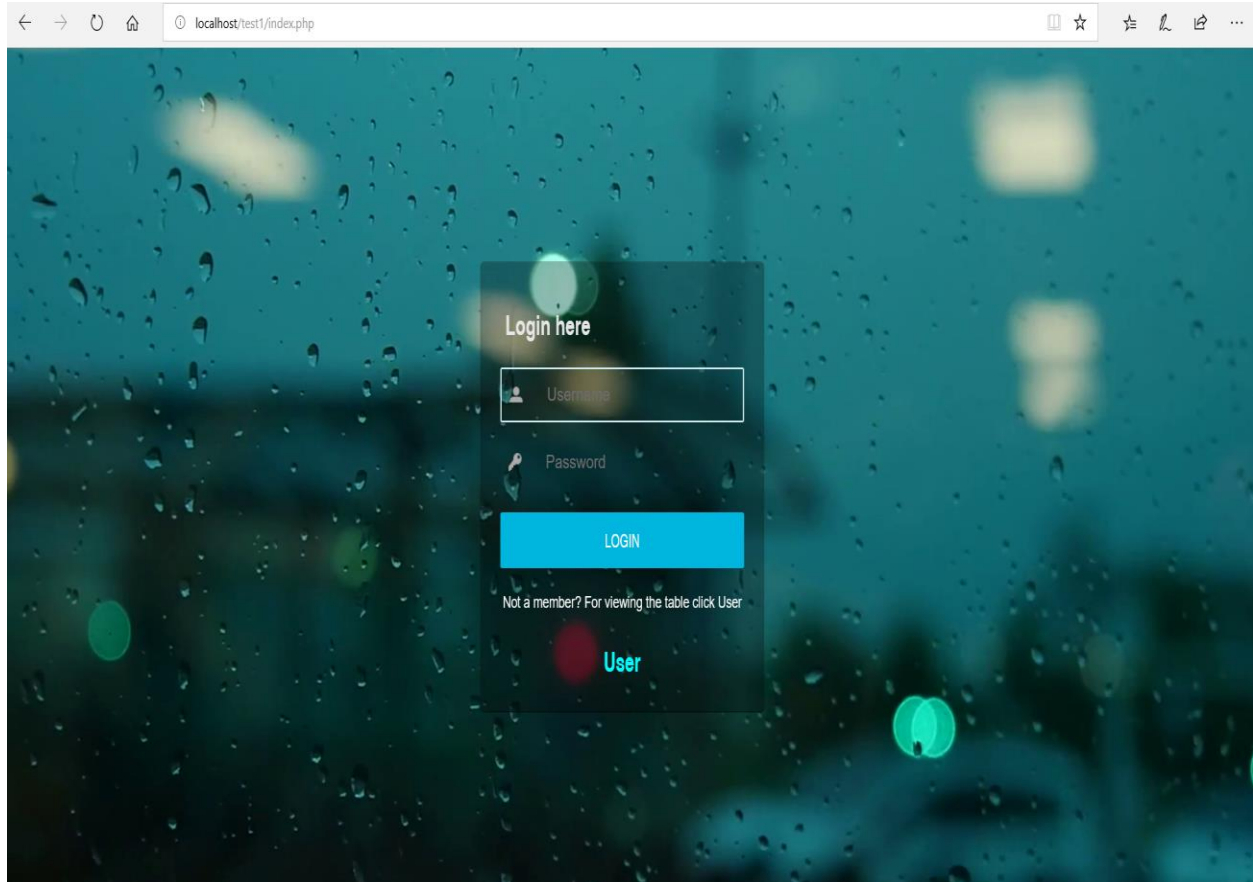
or

```
die("Could not connect to database!");  
mysqli_select_db($dbconn,'Animals');  
$query="SELECT * FROM Animals";  
$result=mysqli_query($dbconn,$query);  
$rownum=mysqli_num_rows($result);  
?>
```

## CHAPTER 4

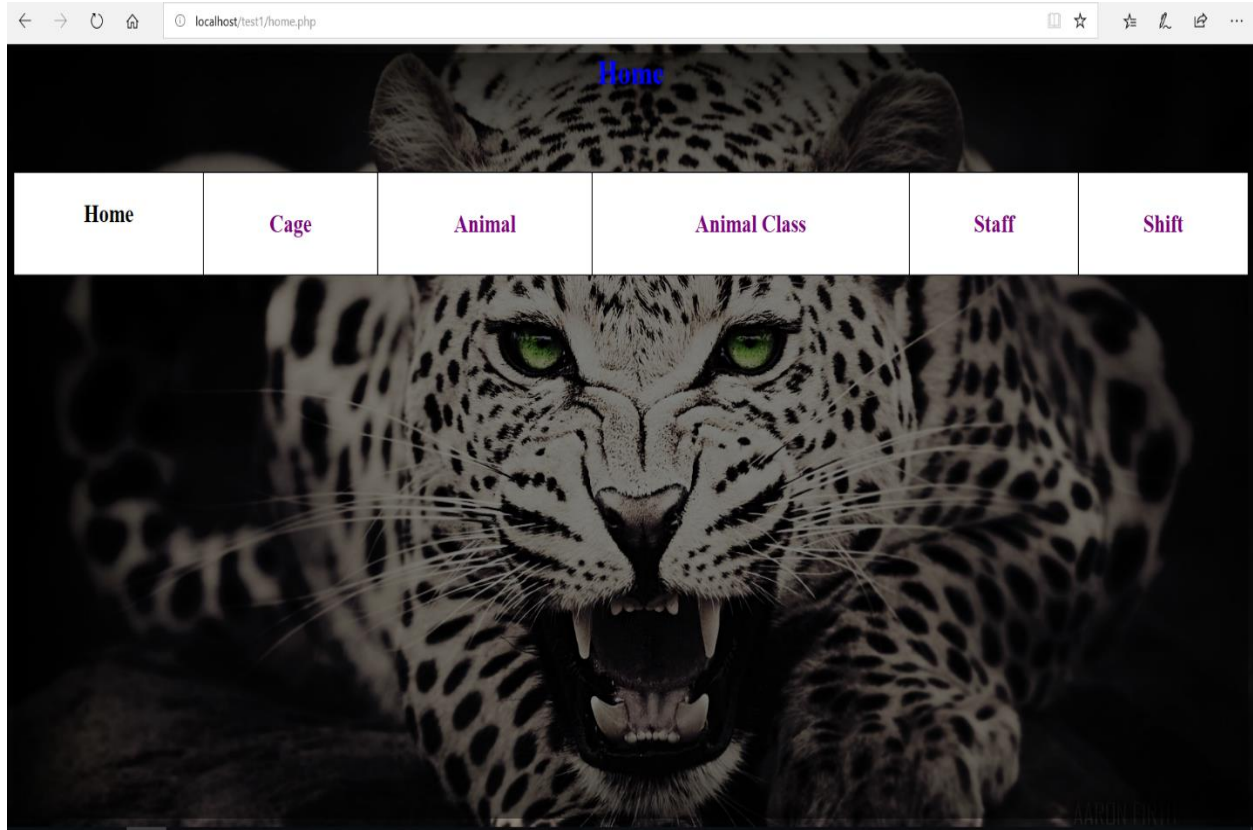
### Results and Discussion

In this chapter the results of the project are discussed. The snapshot of the project showing various functionalities like insert, delete, update and retrieval are showcased.



**Fig. 4.1 Login page of project**

Figure 4.1 shows the login page of the project. Here it has a option for admin login. For normal user also it has one link. If person log in the page then he can edit the values of the table. If he press the user link then he can just view the contents of the table.



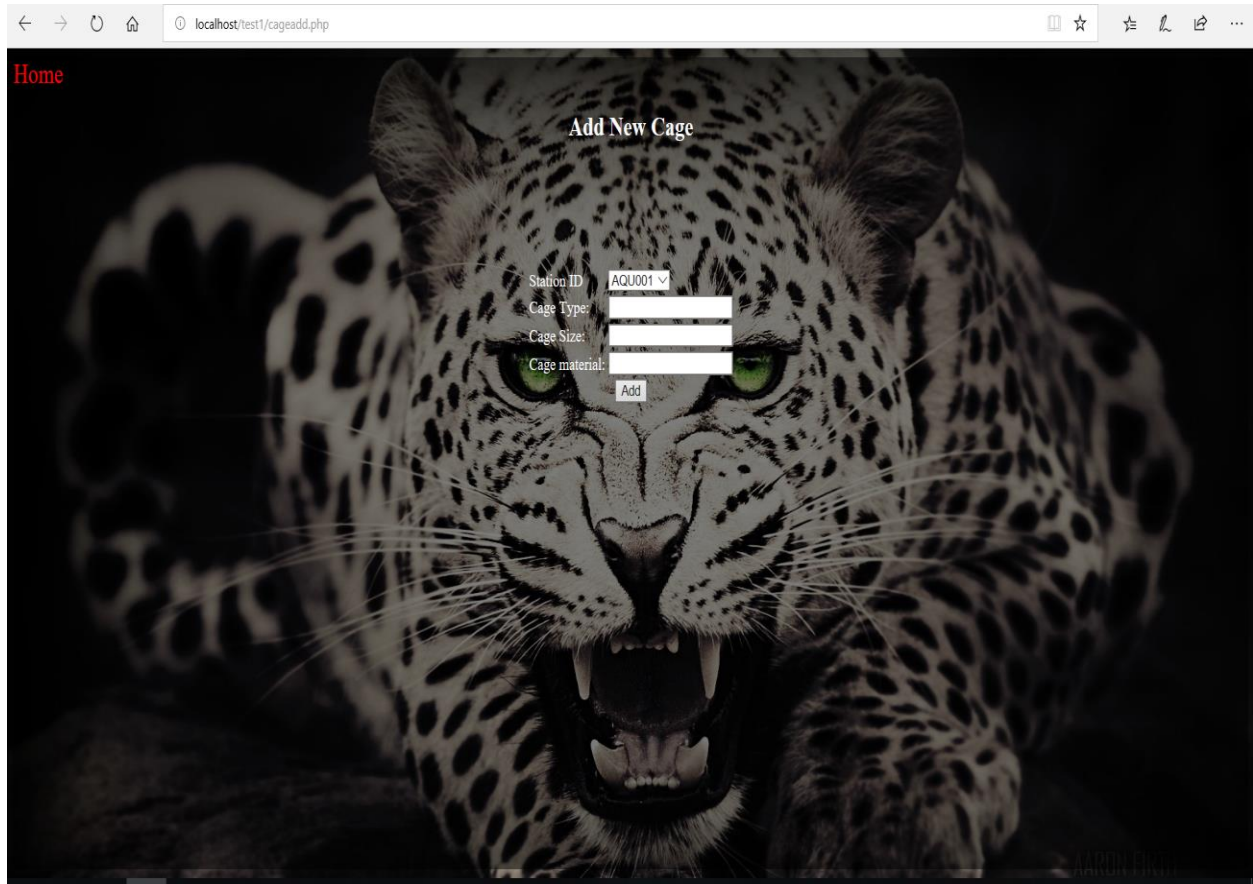
**Fig. 4.2 Welcome page of project.**

Figure 4.1 shows the welcome page of the project. In this page we have some options. Those are the links of every table. By clicking those links we can get enter into that page.

Cage No	Station ID	Cage Type	Cage size	Cage Material	Options
1	TER001	Lion's Den	Big	Iron	<a href="#">Update Delete</a>
2	TER001	Dog Cage	Medium	Wood	<a href="#">Update Delete</a>
3	JNG001	Extra Swing Set Cage	Medium	Metal	<a href="#">Update Delete</a>
4	JNG001	High Gate Cage	Medium	Metal	<a href="#">Update Delete</a>
5	AVR001	Dove Lane	Small	Steel	<a href="#">Update Delete</a>
6	AVR001	Eagle Farm	medium	Iron	<a href="#">Update Delete</a>
7	AQU001	Dolphin Tank	Big	Glass	<a href="#">Update Delete</a>
8	AQU001	Shark Tank X2	Big	Glass	<a href="#">Update Delete</a>
9	JNG001	Jungle Maternity Ward	Big	Metal Net	<a href="#">Update Delete</a>
10	AQU001	Bangus Net	Medium	Iron net	<a href="#">Update Delete</a>

**Fig. 4.3 Retriving the information of the table**

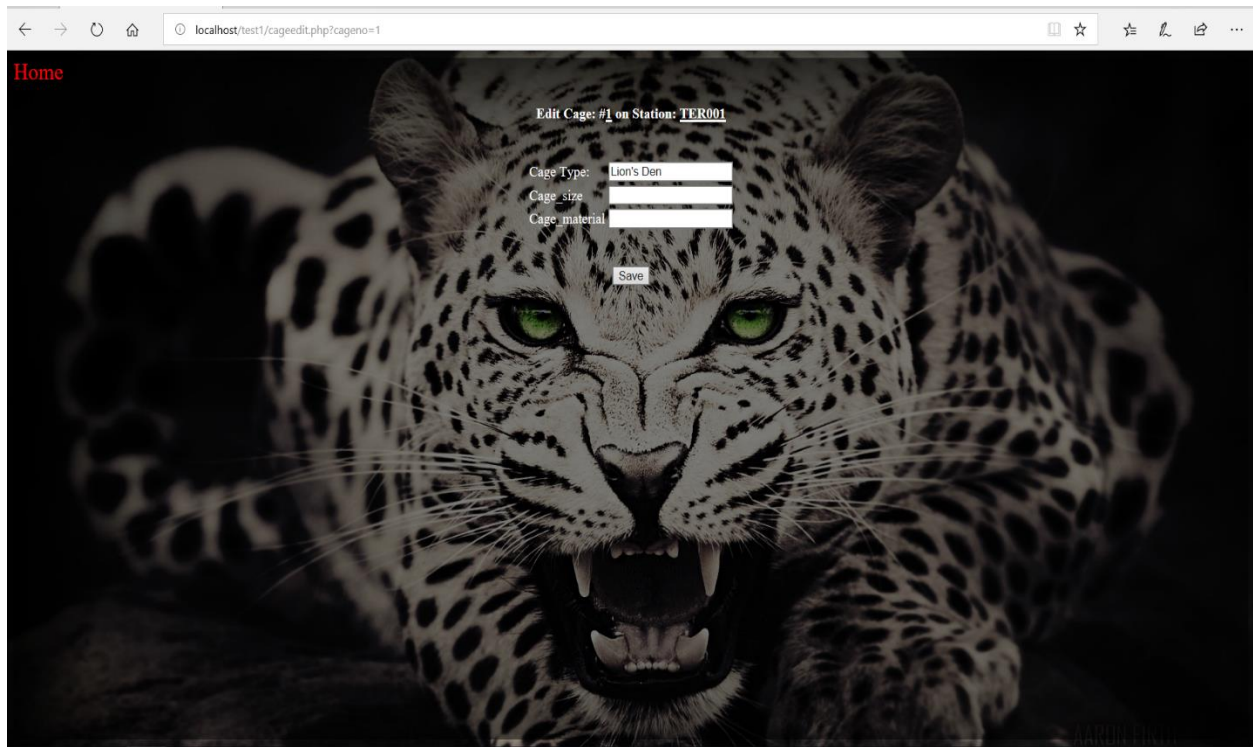
Figure 4.2 shows the cage table in the zoo database, which contain the information about zoo in its attributes. From home page we can get into this cage table by pressing the link in the home page. This page has a link to add, update, and delete a row of a table. An admin can alter ,add or delete the values and user can just see the values in the table.



**Fig. 4.4 Add table page**

Figure 4.3 shows the add table of the page. From every table we have link to add a elements to a table. Only Admin can add elements to a table. It has all the attributes of the table that we are going to inserting. It accepts the inputs from the user and adds that to table. This also has a link to go back to Home page.





**Fig. 4.5 Updating table page.**

Figure 4.4 shows the Updating table page. This page also has access only by admin. He can alter the table. But he can't alter the primary key values. This page also has a link to home page. Every row in a table has a option of updating the vaues. On clicking this Option we can get above window, there we can give the values and we can update the old given values.



Station ID	Cage No : Cage Type	Animal ID	Animal Nick	Gender	Animal Classification
TER001	2 : Dog Cage	001CNN	Browny The Dog	M	CNN001

Staff ID	Staff Name	Staff Type
SCT004	Viktor Lazer	Caretaker
SVT004	Teemo Daquis	Veterinarian

**Fig. 4.6 Search table page.**

Fig 4.5 shows the search table page. Here we can search the animal just by giving the animals nickname ,it made search in the table and give the output to the given valid input. An outsider can easily get the information of animals .

## **CHAPTER 5**

### **CONCLUSION**

Zoo database management helps us to know about the animals in the zoo and about workers in the zoo. The theoretical process involved in database design has been practically implemented. The project provides user friendly interface for the users to interact with the database. All database operations including insertion, deletion, updation and Retrievals are supported along with support for trigger and stored procedure.

## **REFERENCES**

[1] Database systems Models, Languages, Design and Application Programming, RamezElmasri and Shamkant B. Navathe, 7th Edition, 2017, Pearson.

[2] W3Schools: <https://www.w3schools.com/>