CHAPTER I

INTRODUCTION TO DATABASE MANAGEMENT SYSTEM

1.1: INTRODUCTION

Formally, a "database" refers to a set of related data and the way it is organized. Access to this data is usually provided by a "database management system" (DBMS) consisting of an integrated set of computer software that allows users to interact with one or more databases and provides access to all of the data contained in the database (although restrictions may exist that limit access to particular data). The DBMS provides various functions that allow entry, storage and retrieval of large quantities of information and provides ways to manage how that information is organized.

Because of the close relationship between them, the term "database" is often used casually to refer to both a database and the DBMS used to manipulate it.

Outside the world of professional information technology, the term database is often used to refer to any collection of related data (such as a spreadsheet or a card index) as however size and usage requirements typically necessitate use of a database management system.

Existing DBMSs provide various functions that allow management of a database and its data which can be classified into four main functional groups:

- **Data definition** – Creation, modification and removal of definitions that define the organization of the data.
- **Update** – Insertion, modification, and deletion of the actual data.
- **Retrieval** – Providing information in a form directly usable or for further processing by other applications. The retrieved data may be made available in a form basically the same as it is stored in the database or in a new form obtained by altering or combining existing data from the database.

- **Administration** – Registering and monitoring users, enforcing data security, monitoring performance, maintaining data integrity, dealing with concurrency control, and recovering information that has been corrupted by some event such as an unexpected system failure.

Both a database and its DBMS conform to the principles of a particular database model. "Database system" refers collectively to the database model, database management system, and database.

Physically, database servers are dedicated computers that hold the actual databases and run only the DBMS and related software. Database servers are usually multiprocessor computers, with generous memory and RAID disk arrays used for stable storage. RAID is used for recovery of data if any of the disks fail. Hardware database accelerators, connected to one or more servers via a high-speed channel, are also used in large volume transaction processing environments. DBMSs are found at the heart of most database applications. DBMSs may be built around a custom multitasking kernel with built-in networking support, but modern DBMSs typically rely on a standard operating system to provide these functions. Since DBMSs comprise a significant market, computer and storage vendors often take into account DBMS requirements in their own development plans.

Databases and DBMSs can be categorized according to the database model(s) that they support (such as relational or XML), the type(s) of computer they run on (from a server cluster to a mobile phone), the query language(s) used to access the database (such as SQL or XQuery), and their internal engineering, which affects performance, scalability, resilience, and security.

1.2: HISTORY

The sizes, capabilities, and performance of databases and their respective DBMSs have grown in orders of magnitude. These performance increases were enabled by the technology progress in the areas of processors, computer memory, computer storage, and computer networks. The development of database technology can be divided into three eras based on data model or structure: navigational, SQL/relational, and post-relational.

The two main early navigational data models were the hierarchical model and the CODASYL model (network model).

The relational model, first proposed in 1970 by Edgar F. Codd, departed from this tradition by insisting that applications should search for data by content, rather than by following links. The relational model employs sets of ledger-style tables, each used for a different type of entity. Only in the mid-1980s did computing hardware become powerful enough to allow the wide deployment of relational systems (DBMSs plus applications). By the early 1990s, however, relational systems dominated in all large-scale data processing applications, and as of 2018 they remain dominant: IBM DB2, Oracle, MySQL, and Microsoft SQL Server are the most searched DBMS. The dominant database language, standardized SQL for the relational model, has influenced database languages for other data models. Object databases were developed in the 1980s to overcome the inconvenience of object-relational impedance mismatch, which led to the coining of the term "post-relational" and also the development of hybrid object-relational databases.

The next generation of post-relational databases in the late 2000s became known as NoSQL databases, introducing fast key-value stores and document-oriented databases. A competing "next generation" known as NewSQL databases attempted new implementations that retained the relational/SQL model while aiming to match the high performance of NoSQL compared to commercially available relational DBMSs.

1.3 CHARACTERISTICS OF DATABASE APPROACH

- Represent Some Aspects of real world applications
- Manages Information
- Easy Operation implementation
- Multiple views of database
- Data for specific purpose
- It has Users of Specific interest
- Self-Describing nature
- Logical relationship between records and data

1.4 APPLICATION OF DBMS

A Database management system is a computerized record-keeping system. It is a repository or a container for collection of computerized data files. The overall purpose of DBMS is to allow the users to define, store, retrieve and update the information contained in the database on demand. Information can be anything that is of significance to an individual or organization.

Databases touch all aspects of our lives. Some of the major areas of application are as follows:

- Banking
- Airlines
- Universities
- Manufacturing And Selling
- Human Resources

CHAPTER II

REQUIREMENT SPECIFICATION

2.1 Functional requirements :

The system after careful analysis has been identified to proceed with the following modules:

- User information and branches module.
- Goods booking, loading & Deliveries module.
- Recievers and Reporting modules.
- Routes information modules.

2.2 Non-functional requirements:

- **Safety Requirements:**

New system is safe to use. Its usage will not provide any damage or any type of loss to the systems already in use. In addition, the security model is prepared regarding the safety of Database so that data is not lost in case of any damage to the system.

- **Security Requirements:**

Security of the system shall be definitely be maintained through the password system. Each employee of every branch office will need to be authenticated with a login id and password. Any employee cannot change the system date to make proxy presentation of yourself.

- **Software Quality Attributes:**

  Our proposed system shall provide a automate software product for the different department of the courier company. The system will be highly Adaptable, Available, and Portable.

2.3 HARDWARE REQUIREMENTS:

- **Processor** : AMD 6-core FX6300 3.5ghz
- **RAM** : 4 GB
- **Hard disk** : 2 TB
- **Speed Frequency :** 3.5 ghz to 4.1ghz

2.4 SOFTWARE REQUIREMENTS:

- **Operating System** **:** Ubuntu 20.04 LTS (Focal Fossa**)**
- **Database** **:** Lamp
- **Front End** : Visual Studio code(Html,CSS,JAVASCRIPT)
- **Back End** : PHP

CHAPTER III

DESIGN

3.1 ER diagram:

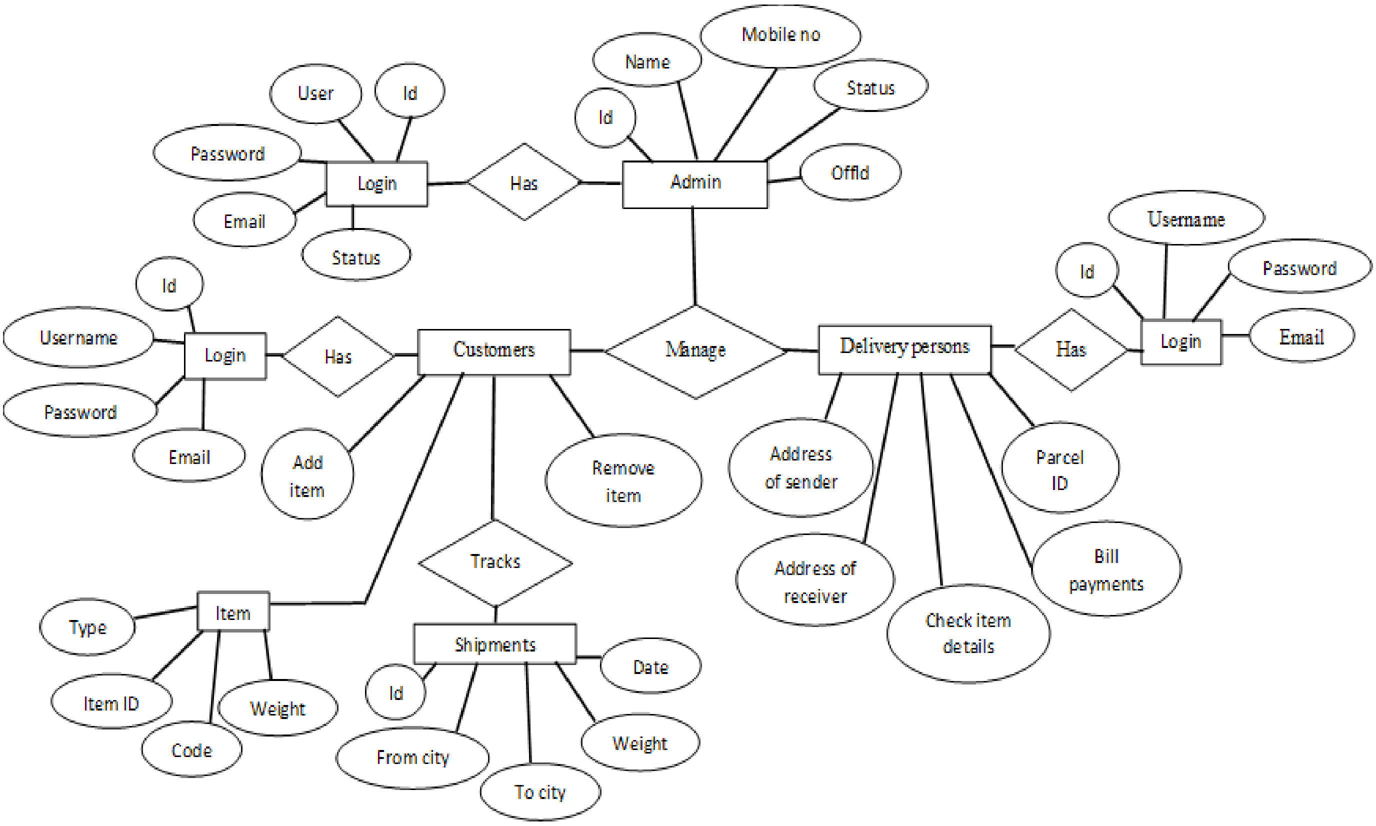


Fig 3.1 ER diagram

3.2 Schema diagram:

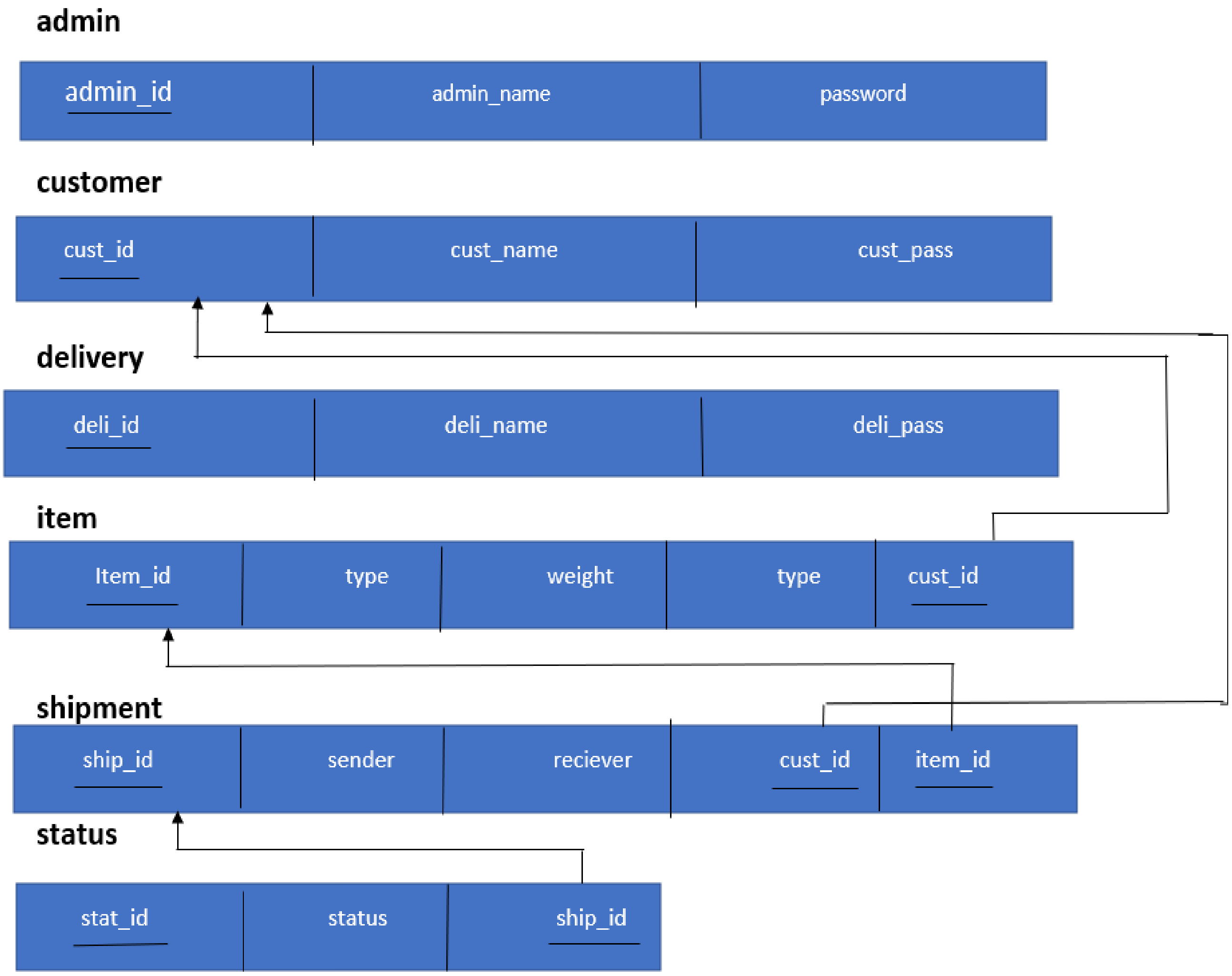


Fig 3.2 Schema diagram

3.3 DATA-FLOW DIAGRAM:

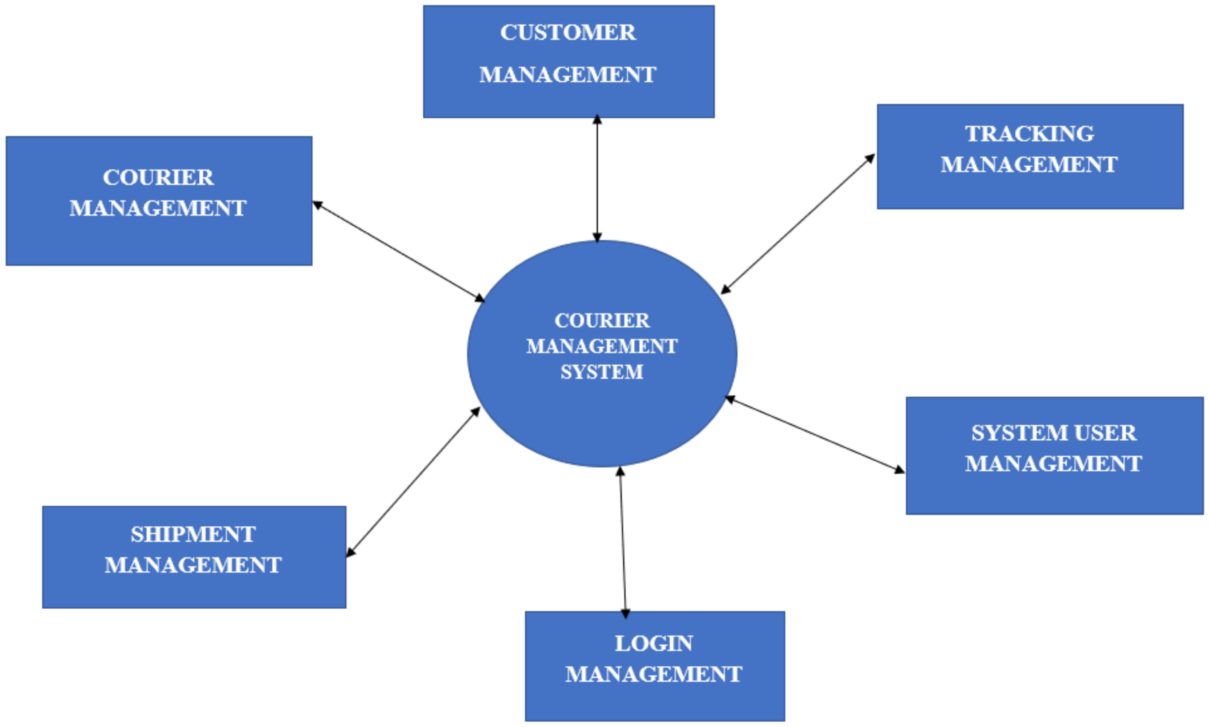


Fig 3.3 Data-flow digram

CHAPTER IV

IMPLEMENTATION

User login.html

```
<!DOCTYPE html>
<html>
<link rel="stylesheet" href="userlogin.css">
<head>
<meta charset="UTF-8">
<title>User Login </title>
<script type="text/javascript">
function fn2()
{
window.location ="welcome user.php";
}
</script>
</head>
<form action="connection.php" method= "POST" onsubmit="return fn1()">
<body class="class">
<center>
<h1>
<label class="dp"> User Login</label>
</h1>
</center>
<center class="login">
<lable class="log">Login here</lable>
```

```
<br>
<br>
<label for="username"class="usname"> Enter username </label>
<br>
<br>
<input type="text" id="uname" name="username" >
<br>
<br>
<label for="password" class="pswd"> Enter password </label>
<br>
<br>
<input type="text" id="pwd" name="password">
<br>
<br>
<button type="submit" class="submit" on click="fn2()" >submit</button>
</center>
            </body>
         </form>
    <footer class="create">
        <h5>Created by Mani Bharathi.R and Kevin.V</h5>
    </footer>
</html>
```

User login.php

```
<?php
include_once 'data.php';
```

```
{
$username = $_POST['username'];
$password = $_POST['password'];
$sql = "INSERT INTO customer (name, password) VALUES ('$username','$password')"
;
mysqli_query($conn, $sql);
header("Location: ../userlogin.php?signup=success");
}
?>
```

Admin login.html

```
<!DOCTYPE html>
<html>
<link rel="stylesheet" href="adminlogin.css">
<head>
<meta charset="UTF-8">
<title>Admin Login </title>
<script type="text/javascript">
function fn1()
{
var Username = document.getElementById("uname");
var Password = document.getElementById("pwd");
if(Username.value.trim() ==""||Password.value.trim() =="")
{
alert("please provide Username and Password");
return false;
}
else
{
true;
}
}
</script>
</head>
<form action="connection.php" method= "POST" onsubmit="return fn1()">
<body class="class">
<form>
<center>
<h1>
<label class="dp"> Admin Login</label>
</h1>
</center>
<center class="login">
<lable class="log">Login here</lable>
<br>
<br>
<label for="username"class="usname"> Enter username </label>
<br>
<br>
<input type="text" id="uname" name ="username" >
<br>
<br>
<label for="password" class="pswd"> Enter password </label>
```

```
<br>

<br>

<input type="text" id="pwd" name ="password" >

<br>

<br>

<button type="submit" class="submit" onclick="">submit</button>

</center>

</form>

</body>

<footer class="create">

<h5>Created by Mani Bharathi.R and Kevin.V</h5>

</footer>

</html>
```

Admin login.php

```
<?php
    include_once 'datacon.php';
{
    $username = $_POST['username'];
    $password = $_POST['password'];

    $sql = "INSERT INTO admin (name, password) VALUES ('$username','$password')";
    mysqli_query($conn, $sql);



    header("Location: ../admin.php?signup=success");

}
?>
```

Delivery person login.html

```
<!DOCTYPE html>
<html>
<link rel="stylesheet" href="dplogin.css">
<head>
<meta charset="UTF-8">
<title>Delivery Login </title>
<script type="text/javascript">
function fn1()
{
var Username = document.getElementById("uname");
var Password = document.getElementById("pwd");
if(Username.value.trim() ==""||Password.value.trim() =="")
{
alert("please provide Username and Password");
return false;
}
else
{
true;
window.location ="Welcome-dp/Welcome-dp.html";
}
}


</script>
</head>
<form action="test.php" method= "POST" onsubmit="return fn1()">
<body class="class">
<center>
<h1>
<label class="dp"> Delivery Login</label>
</h1>
</center>
```

```
<center class="login">
<lable class="log">Login here</lable>
<br>
<br>
<label for="username"class="usname"> Enter username </label>
<br>
<br>
<input type="text" id="uname" name="username" >
<br>
<br>
<label for="password" class="pswd"> Enter password </label>
<br>
<br>
<input type="text" id="pwd" name="password" >
<br>
<br>
<button type="submit" class="submit" onclick="return fn1()">submit</button>
</center>
</form>
</body>
<footer class="create">
<h5>Created by Mani Bharathi.R and Kevin.V</h5>
</footer>
</html>
```

Delivery person login.php

```
<?php
$servername='localhost';
$username='root';
$password=";
$dbname='courier';
$conn=mysqli_connect($servername, $username, $password, $dbname);
if(!$conn){
    die('Could not Connect MySql Server:' .mysql_connect_error());
  }
?>
```

All orders.html

```
<!DOCTYPE html>
<html>
<link rel="stylesheet" href="all_order.css">
<head>
<title> All order</title>
</head>
<body class="body">
<center>
<form>
<h1><label class="all">All Order</label></h1>
<br>
<table class="tab">
<tr>
<th id="cuid">CUSTOMER ID</th>
<th><label class="name">NAME</label></th>
<th><label class="cumob">CUSTOMER MOBILE </label><br> NUMBER</th>
<th><label class="itemid">ITEM ID</label></th>
<th><label class="type">TYPE</label> TYPE</th>
<th><label class="weight">WEIGHT(IN KG)</label> </th>
<th><label class="addofsen">ADDRESS OF SENDER</label> </th>
<th><label class="assofrec">ADDRESS OF RECIVER</label></th>
<th><label class="status">STATUS</label></th>
</tr>
```

```
</table>
<footer>
</footer>
</form>
</center>
</body>
</html>
```

All-orders data.php

```
<?php
$servername='localhost';
$username='root';
$password=";
$dbname="courier";
$conn=mysqli_connect($servername,$username,$password,"$dbname");
if(!$conn){
    die('Could not Connect MySql Server:' .mysql_error());
}
?>
<?php
$sql = "SELECT * FROM details ";
$result = mysqli_query($conn, $sql);
?>
<?php
if (mysqli_num_rows($result) > 0) {
?>
  while($row = mysqli_fetch_assoc($result)) {
   <tr>
  <td><?php echo $row["item"]; ?></td>
  <td><?php echo $row["weight"]; ?></td>
  <td><?php echo $row["pickup"]; ?></td>
  <td><?php echo $row["dropa"]; ?></td>
 }
} else {
 echo "0 results";
}

mysqli_close($conn);
?>
```

All-orders connection.php

```
<?
$sql = "SELECT id, item,weight,pickup,dropa,phoneon FROM details ";
$result = mysqli_query($conn, $sql);

if (mysqli_num_rows($result) > 0) {
  // output data of each row
  while($row = mysqli_fetch_assoc($result)) {
    echo "id: " . $row["id"]. " - Name: " . $row["firstname"]. " " . $row["lastname"]. "<br>";
  }
} else {
  echo "0 results";
}

mysqli_close($conn);
?>
```

Customer order connection.php

```
<?php
$servername='localhost';
$username='root';
$password='';
$dbname="courier";
$conn=mysqli_connect($servername,$username,$password,$dbname);
if(!$conn){
    die('Could not Connect MySql Server:' .mysql_error());
   }

$sql = "SELECT id, item, weight, pickup, dropa  FROM details";
$result = mysqli_query($conn, $sql);

if (mysqli_num_rows($result) > 0) {
  // output data of each row
  while($row = mysqli_fetch_assoc($result)) {
    echo "<tr>
    <td>" .$row["id"]. "</td>
    <td>" .$row["item"]. "</td>
    <td>" .$row["weight"]. "</td>
    <td>" .$row["pickup"]. "</td>
    <td>" .$row["dropa"]. "</td>
    </tr>"

  }
} else {
  echo "0 results";
}
mysqli_close($conn);
?>
```

Customer order.html

```
<!DOCTYPE html>
<html>
<link rel="stylesheet" href="customerorder.css">
<head>
<title> All order</title>
</head>
<body class="body">
<center>
<form action="connection.php" method="POST">
<h1><label class="all">Customer order</label></h1>
<br>
<table class="tab">
<tr>
<th><label class="id">ITEM ID</label></th>
<th><label class="item">TYPE</label></th>
<th><label class="weight">WEIGHT(IN KG)</label></th>
<th><label class="pickup">ADDRESS OF SENDER</label></th>
<th><label class="dropa">ADDRESS OF RECIVER</label></th>
<th><label class="status">STATUS</label></th>
</tr>
</table>
<footer>

</footer>
</form>
</center>
</body>
</html>
```

Status-check.html

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

```
<link rel="stylesheet" href="checkstatus.css">
<head>
<title>check status</title>
</head>
<script type="text/javascript">
function fn1()
{
var itemid = document.getElementById("itemid");
if(itemid.value.trim() =="")
{
alert("Enter Item Id");
return false;
}
else
{
true;
}
}
</script>
<body class="body">
<form onsubmit="return fn1()" action="" method="POST">
<center>
<h1>
<label class="check"> Check status</label>
</h1>
</center>
<h3>
<i>
<label for="itemid" class="id">Enter The Item Id </label>
<input type="text" id="itemid">
</i>
</h3>
<br>
<br>
```

```
<button type="submit" class="submit" value="submit">submit</button>
</form>
</body>
</html>
```

Welcome-user.html

```
!DOCTYPE html>
<html>
<script>

function additem()
{

}
function itemprice()
{

}
function checkstatus()
{

}
</script>
    <link rel="stylesheet" href="welcomeuser.css">
    <head>
        <title> welcome user </title>
    </head>
    <body class="body">
        <form>
        <center>
        <h1><label class="welcome">welcome User</label></h1>
        </center>
        <div class="add">
        <a onclick="additem()">ADD ITEM</a>
        </div>
        <div class="Check" >
        <a onclick="checkstatus()">CHECK STATUS</a>
        </div>

    </form>
</body>
</html>
```

Courier database.sql

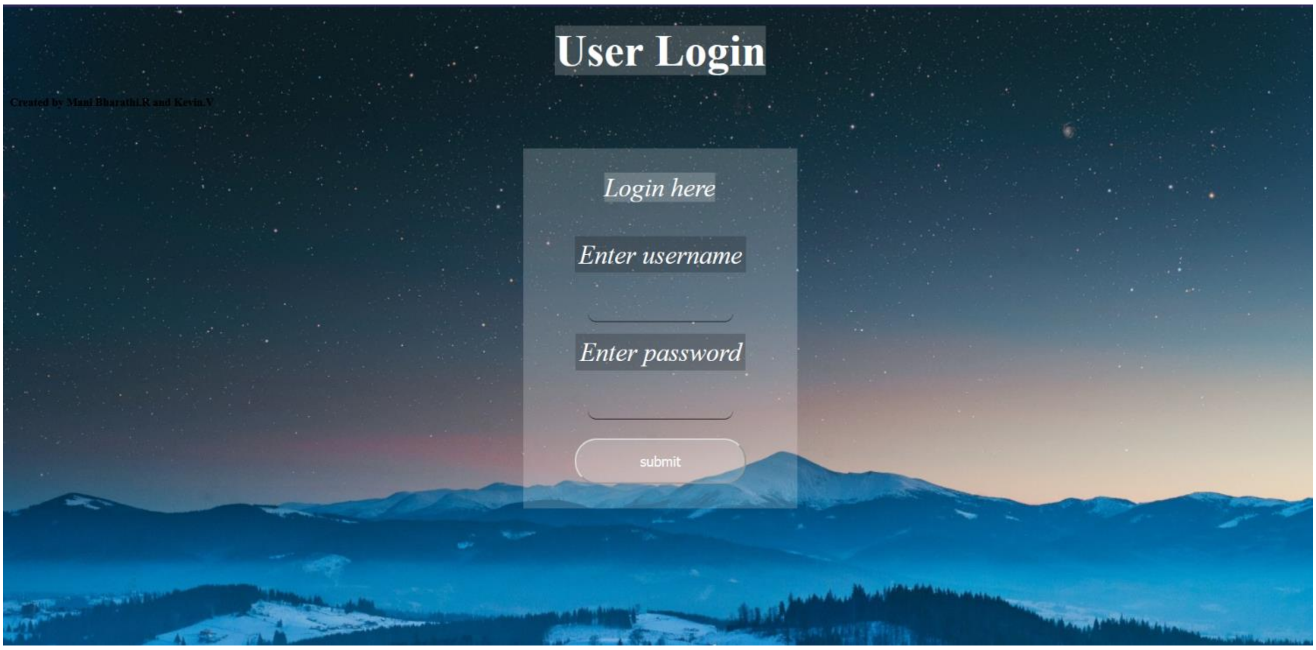
```
CREATE DATABASE COURIER;
USE COURIER;
CREATE TABLE customer(
    id int(11),
    name varchar(30),
    password varchar(30),
    PRIMARY KEY (id)
);
CREATE TABLE delivery(
    id int(10),
    name varchar(30),
    password varchar(30),
    PRIMARY KEY (id)
);
CREATE TABLE admin(
    id int(10),
    name varchar(30),
    password varchar(30),
    PRIMARY KEY (id)
);
CREATE TABLE item(
    id int(10),
    type varchar(30),
    weight varchar(10),
    phoneno varchar(20)
);

CREATE TABLE shipment(
  id int(10),
  address_of_sender(30),
  address_of_reciver(30),
  status varchar(30)
  item_id --as foriegn key
  PRIMARY KEY (id)
);

if (mysqli_query($conn, $sql)) {
        echo "New record created successfully";
      } else {
        echo "Error: " . $sql . "<br>" . mysqli_error($conn);
      }
     mysqli_close($conn);
```

CHAPTER V

SNAPSHOTS



User Login

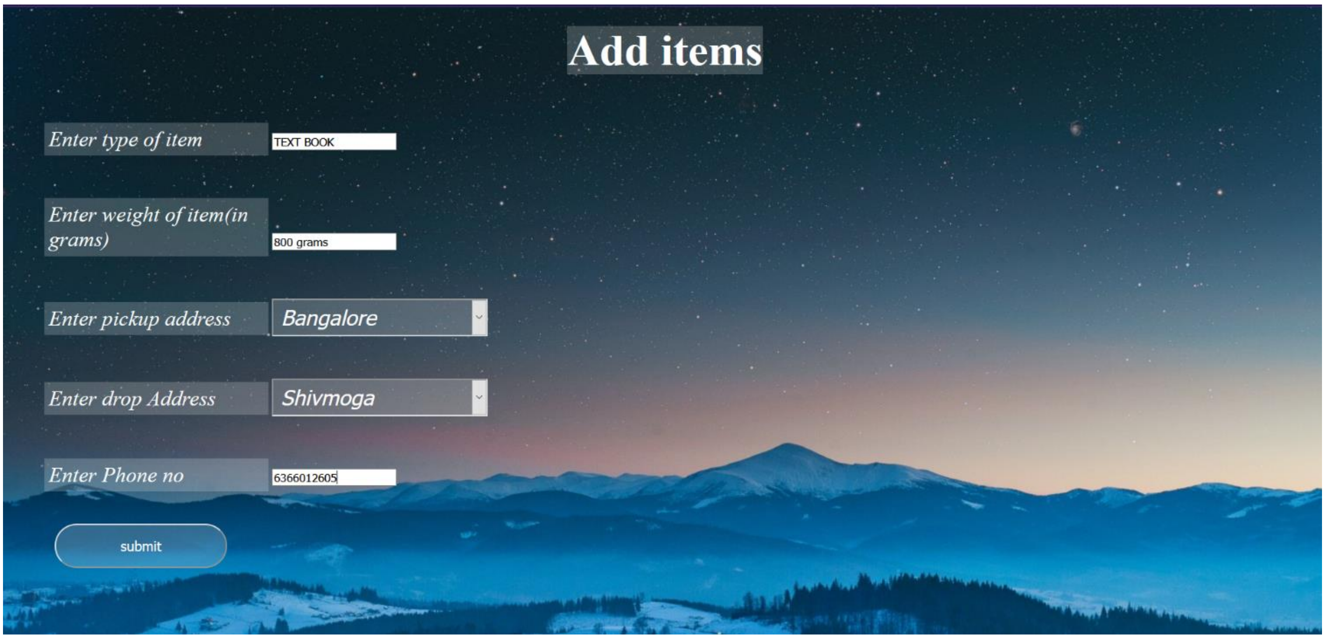
Created by Mani Bharathi.R and Kevin.V

Login here

Enter username

Enter password

Fig 6.1 Customer login



Add items

Enter type of item: TEXT BOOK

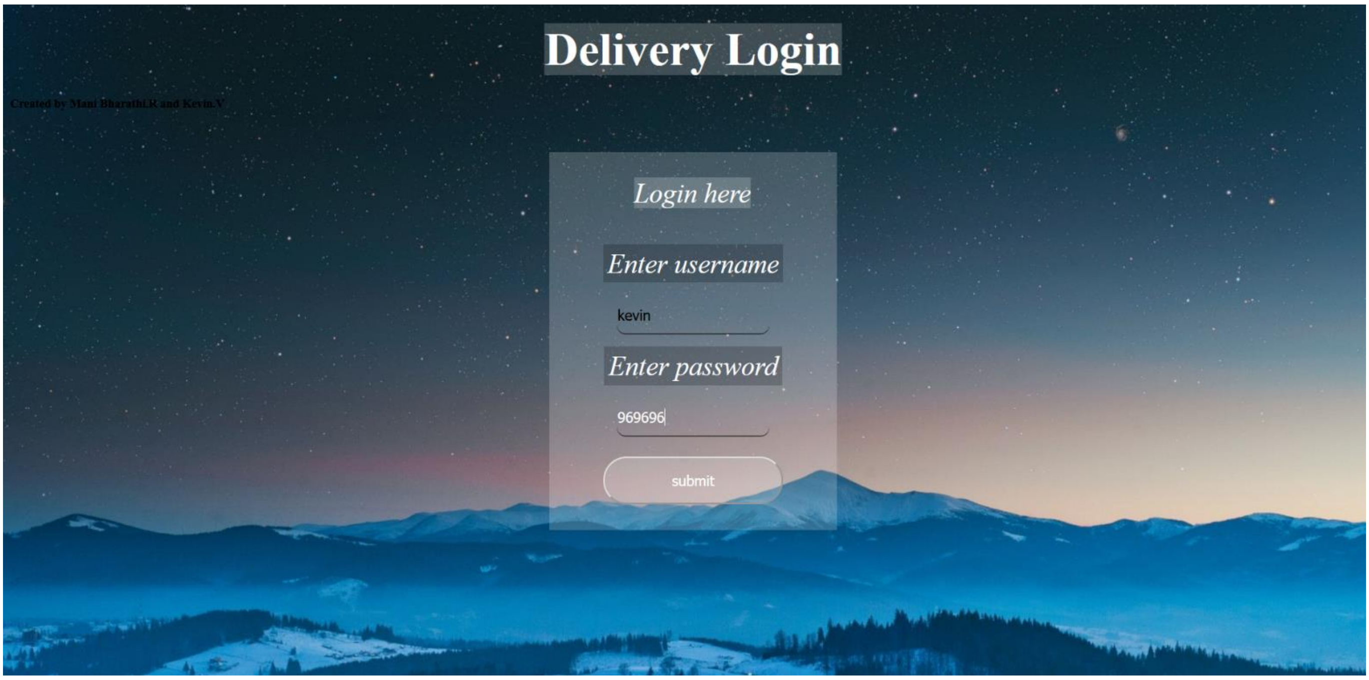
Enter weight of item(in grams): 800 grams

Enter pickup address: Bangalore

Enter drop Address: Shivmoga

Enter Phone no: 6366012605

Fig 6.2 Adding items



Delivery Login

Created by Mani Bharathi.R and Kevin.V

Login here

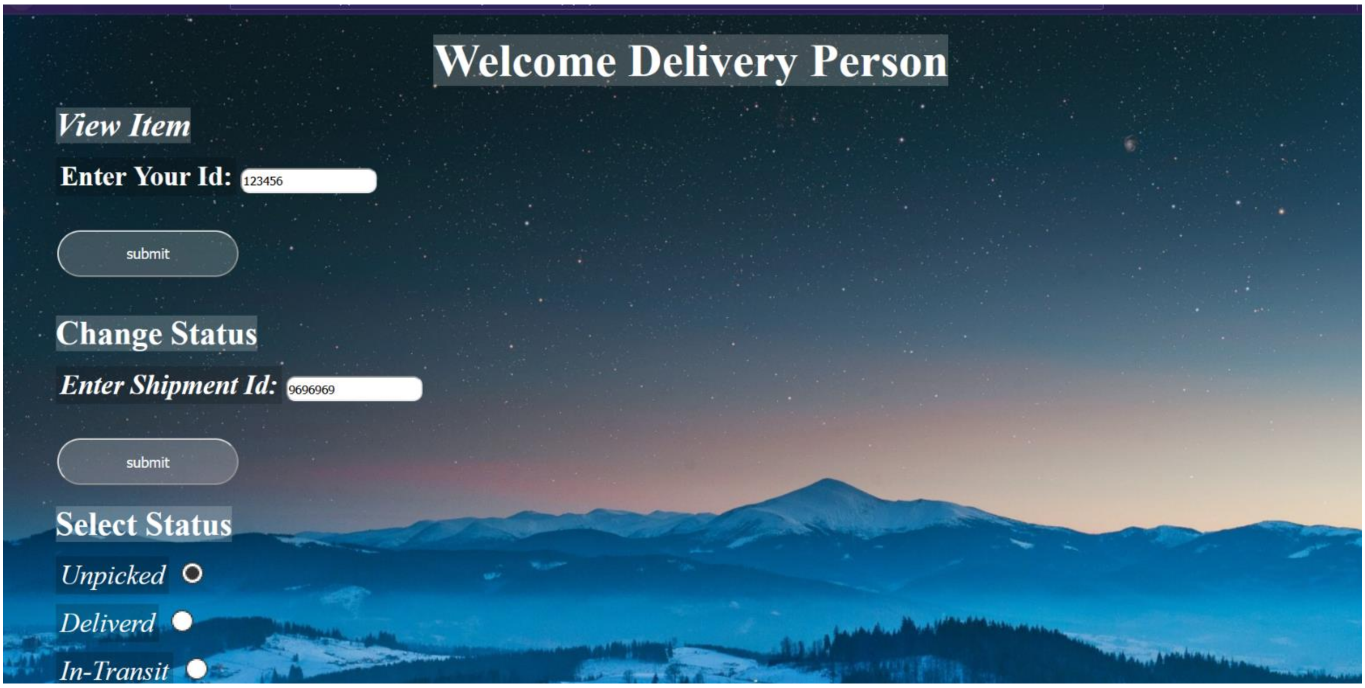
Enter username

kevin

Enter password

969696

Fig 6.3 Delivery person login



Welcome Delivery Person

View Item

Enter Your Id: 123456

Change Status

Enter Shipment Id: 9696969

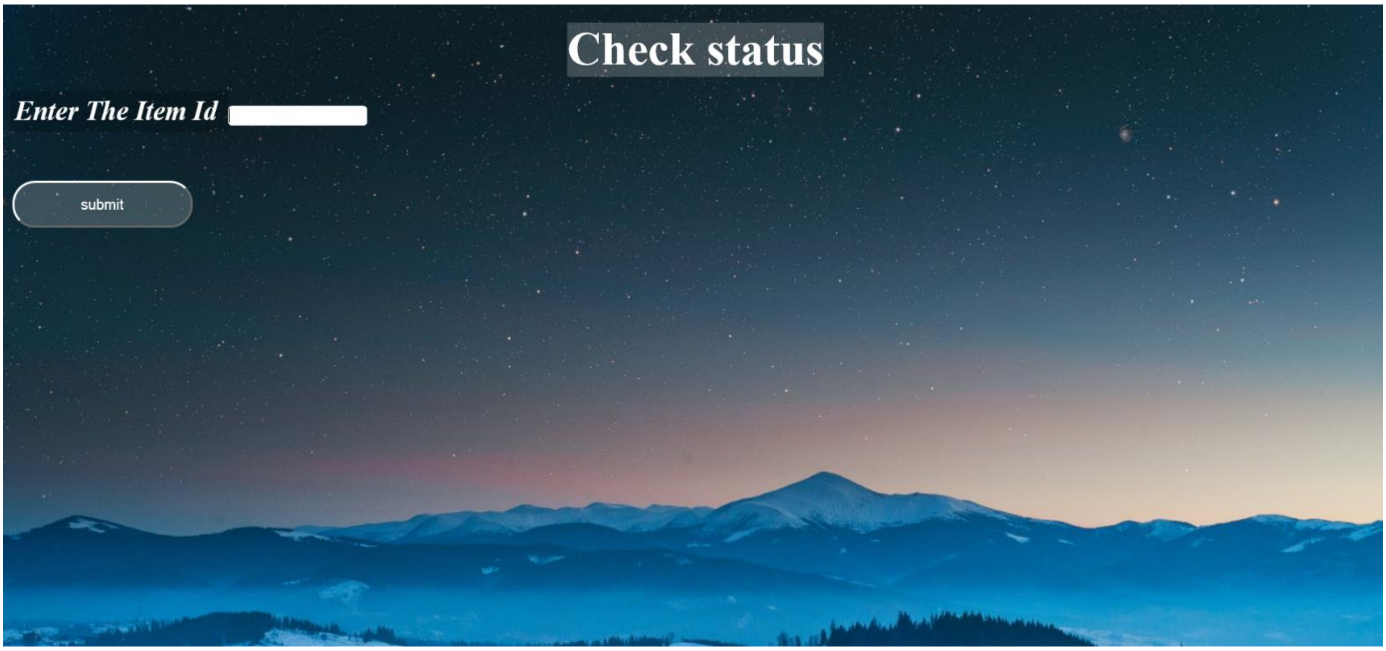
Select Status

Unpicked ●

Deliverd ○

In-Transit ○

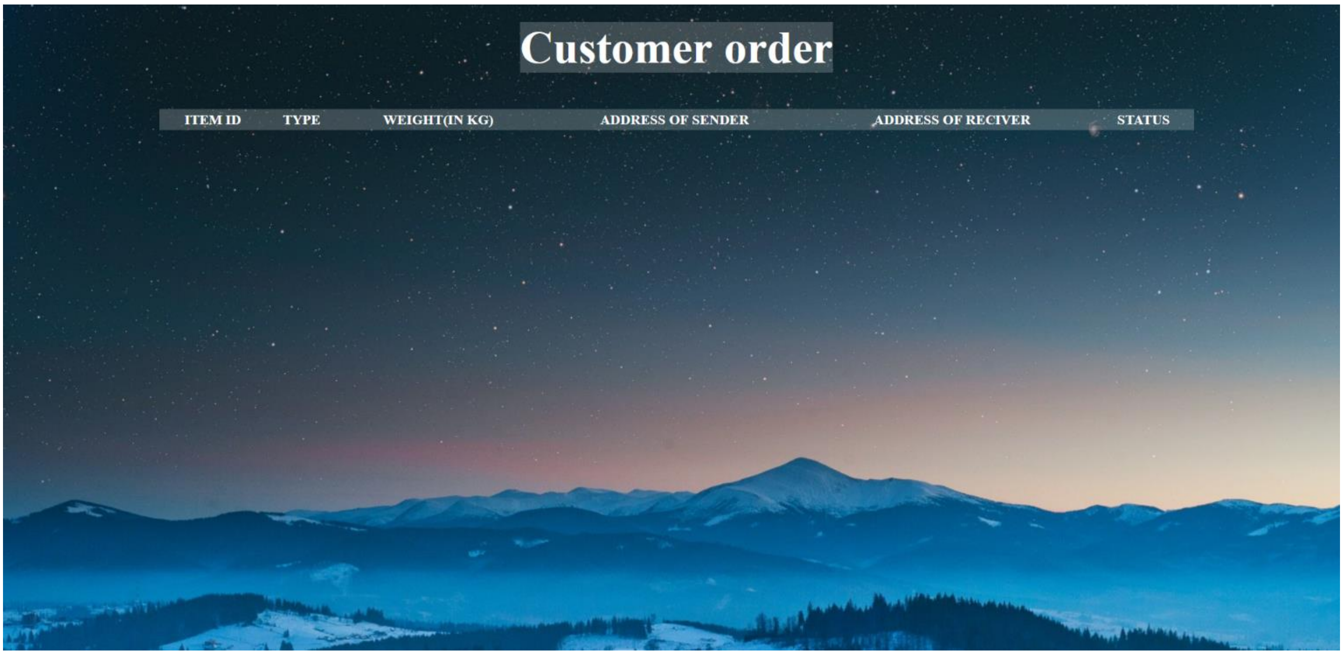
Fig 6.3 Welcome Delivery person



Check status

Enter The Item Id

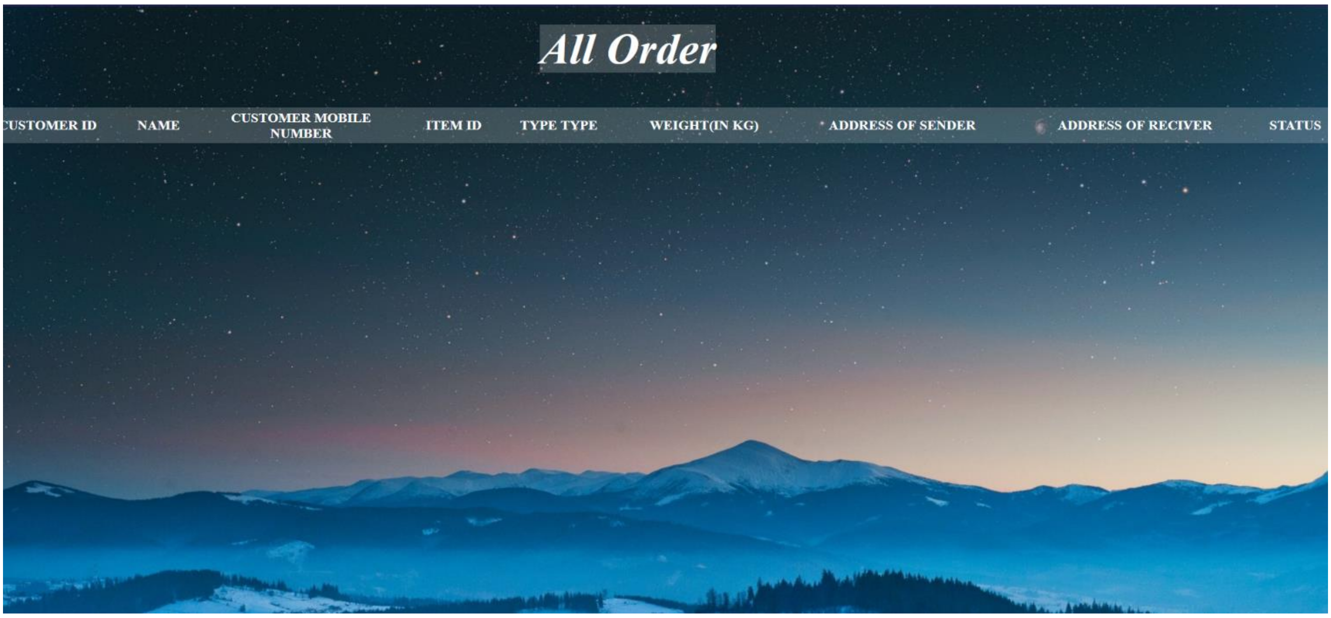
Fig 6.4 Check item status



Customer order

ITEM ID	TYPE	WEIGHT(IN KG)	ADDRESS OF SENDER	ADDRESS OF RECIVER	STATUS

Fig 6.5 Customer Orders



All Order

CUSTOMER ID	NAME	CUSTOMER MOBILE NUMBER	ITEM ID	TYPE TYPE	WEIGHT(IN KG)	ADDRESS OF SENDER	ADDRESS OF RECIVER	STATUS

Fig 6.6 All orders

CONCLUSION

After this project, User can register themselves and book for courier service using thesystem. And the branch manager can view and make deliveries for the couriers. Only theregistered user can book for the courier and view status of the booked courier. The branchmanager can add new staff members or delete them. Thestatus of the courier can bechanged by the branch manager and also specifying the staff member who delivered it.

The project titled ‘Courier Management Service’ was developed to the courierservices anddirection and with their help. The system was tested and the performance of the system wasfound to be acceptable.All the necessary output was created.The system was found to beuser-friendlywith help message for the customer. The menu Driven Architecture of thesystem provide an easy to use environment for the user.

- A courier delivers messages, packages, and mail.
- Couriers are distinguished from ordinary mail servicesby features such as speed, security, tracking,signature, specialization and individualization ofexpress services, and swift delivery times, which areoptional for most everyday mail services.
- The aim of this research project is to design andimplement a Courier Service Packaging and Courier Management System that will automate the process of delivery tracking and monitoring for the recipients of the deliveries.

FUTURE ENHANCEMENT

The use of this system is for the better services and faster processing. The application is very useful and easy to use. More time will be saved. During billing process, system generates a consignment number for their products. Through this consignment number, customer or its recipients will be able to track their products from any location using Internet.

In courier services sector more company is providing services but in India my courier system is leading company in courier sector.

- The use of this System for the better services and faster processing.
- The application is very useful and easy to use.
- Courier facility in India is very cheaper and maximum charges but we can send the courier to destination in minimum charges.
- More time will be save.

REFERENCES

- **REFERENTIAL URL's:**

[1] www.w3schools.com

[2] www.tutorialspoint.com/mysql/

✓ **REFERENCE BOOKS:**

[1] RamezElmarsi and ShamkantNavathe, "Fundamentals of Database System", 2017 [7th Edition]

[2] Ramakrishnan and Gehrke, "Database Management System", 2014 [3rd Edition]

[3] Bipin Desai, "An Introduction to Database System"

[4] HTML and XHTML: The Complete Reference - Thomas A. Powell

[5] PHP & MySQL Novice to Ninja – Kevin Yank