VISVESVARAYA TECHNOLOGICAL UNIVERSITY

JNANA SANGAMA, BELAGAVI – 590 018

A Mini Project Report on

Courier Service Management System

Submitted in partial fulfillment of the requirements as a part of the DBMS Lab for the award of degree of

Bachelor of Engineering
in
Information Science and Engineering

Submitted by

Aditya Didwania
1RN16IS005

Ekalavaya Maurya
1RN16IS030

Faculty Incharge
Mrs.Tejashwini P
Assistant Professor
Dept. of ISE, RNSIT

Lab Incharge
Mr. R Rajkumar
Assistant Professor
Dept. of ISE, RNSIT

Department of Information Science and Engineering
RNS Institute of Technology
Channasandra, Dr. Vishnuvardhan Road, RR Nagar Post,
Bengaluru – 560 098

2018 – 2019

RNS Institute of Technology

DEPARTMENT OF INFORMATION SCIENCE & ENGINEERING

CERTIFICATE

This is to certify that the Mini project report entitled ***Courier Service Management System*** has been successfully completed by **Aditya Didwania** bearing USN **1RN16IS005 and**, **Ekalavaya Maurya** bearing USN **1RN16IS030**, presently V semester student of **RNS Institute of Technology** in partial fulfillment of the requirements as a part of the DBMS Laboratory for the award of the degree ***Bachelor of Engineering in Information Science and Engineering*** under **Visvesvaraya Technological University, Belagavi** during academic year 2018 – 2019. It is certified that all corrections/suggestions indicated for Internal Assessment have been incorporated in the report deposited in the departmental library. The mini project report has been approved as it satisfies the academic requirements as a part of DBMS Laboratory for the said degree.

Ms.Tejashwini P Faculty Incharge	**Mr.R Rajkumar** Lab Incharge	**Dr. M V Sudhamani** Professor and HOD

External Viva

Name of the Examiners	**Signature with date**
1. _____	_____
2. _____	_____

ABSTRACT

The purpose of Courier Management System is to automate the existing manual system by the help of computerized equipment and full-fledged computer software, fulfilling their requirements, so that their valuable data/information can be stored for a longer period with easy accessing and manipulation of the same.

The required software and hardware are easily available and easy to work with. Courier Management System, as described above, can lead to error free, secure, reliable and fast management system. It can assist the user to concentrate on their other activities rather to concentrate on the record keeping.

We have used Database Management System to implement this idea. We used HTML and CSS for front-end and JSP for back-end development. DBMS provides an efficient way for storing data in tabular format. It provides simple yet efficient way to insert, update, delete or retrieve the data stored. JSP allows Java code and certain predefined actions to be interleaved with static web markup content, such as HTML, with the resulting page being compiled and executed on the server to deliver a document. HTML and CSS are used to present the data in a presentable way for the end user.

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.

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

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

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

We 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.

We would like to express my profound and cordial gratitude to my Lab Incharge **Mr. R Rajkumar**, Assistant Professor, Department of Information Science and Engineering for their valuable guidance, constructive comments and continuous encouragement throughout the Project work.

We would like to express my profound and cordial gratitude to my Faculty Incharge **Mrs. Tejashwini P** , Assistant Professor, Department of Information Science and Engineering for his/her valuable guidance in preparing Project report.

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

ADITYA DIDWANIA USN: 1RN16IS005

EKALAVAYA MAURYA USN:1RN16IS030

TABLE OF CONTENTS

LIST OF FIGURES

ABBREVIATIONS

HTML	-	Hypertext Markup Language
CSS	-	Cascading Style Sheets
JSP	-	Java Server Pages
SMS	-	Short Message Service
ADMIN	-	Administrator
ER	-	Entity Relationship

Chapter 1

INTRODUCTION

The Database Management System (DBMS) is software that enables the users to define, create, maintain and control the access to the database. It is a software that interact with the user's applications programs and its database. The DBMS have the ability to store, update and retrieve the data. This is the main function of the DBMS because the database can be used if there is any record is being stored into the database. The record need to be retrieve first, then it can be change by the database administrator as it will be the record has been updated. The DBMS will protect the structure of the data structure. One of the main feature of DBMS is that it provides the facility for multiple access simultaneously, i.e. it allows several users to use the database all at the same time. This feature is quite useful for banks and online e-commerce websites.

1.1 Background

The whole database is divided into number of tables and they have relation among them. The diagram showing these tables and relations among them is called Entity-Relationships Diagram (E-R Diagram). Based on this E-R Diagram the schema diagram is prepared which is the basic architecture of the database. According to the architecture or model of the database the tables, their attributes and relations among the tables are created using SQL. Structured Query Language, also known as SQL, is utilized to interact with a database. Per ANSI (American National Standards Institute), it is the standard language for relational database management systems. SQL statements are utilized to perform tasks, for example, upgrade information on a database, or recover information from a database. "Some common relational database management systems that use SQL are: Oracle, Sybase, Microsoft SQL Server, Access, Ingres, etc. Although most database systems use SQL, most of them also have their own additional proprietary extensions that are usually only used on their system. However, the standard SQL commands such as "Select", "Insert", "Update", "Delete", "Create", and "Drop" can be used to accomplish almost everything that one needs to do. Microsoft SQL is the SQL used in this project.

JSP allows Java code and certain predefined actions to be interleaved with static web markup content, such as HTML, with the resulting page being compiled and executed on the server to deliver a document. The compiled pages, as well as any dependent Java libraries, contain Java bytecode rather than machine code. Like any other Java program, they must be executed with in a Java virtual machine (JVM) that interacts with the server's host operating system to provide an abstract, platform-neutral environment.

JSPs are usually used to deliver HTML and XML documents, but through the use of OutputStream, they can deliver other types of data as well. The Web container creates JSP implicit objects like request, response, session, application, config, page, pageContext, out and exception. JSP Engine creates these objects during translation phase.

HTML (Hypertext Markup Language) and CSS (Cascading Style Sheet) are the languages used display the data in a presentable manner as a web page in the browser. In this project we have used HTML and CSS for front-end development that displays the data retrieved from the database via SQL.

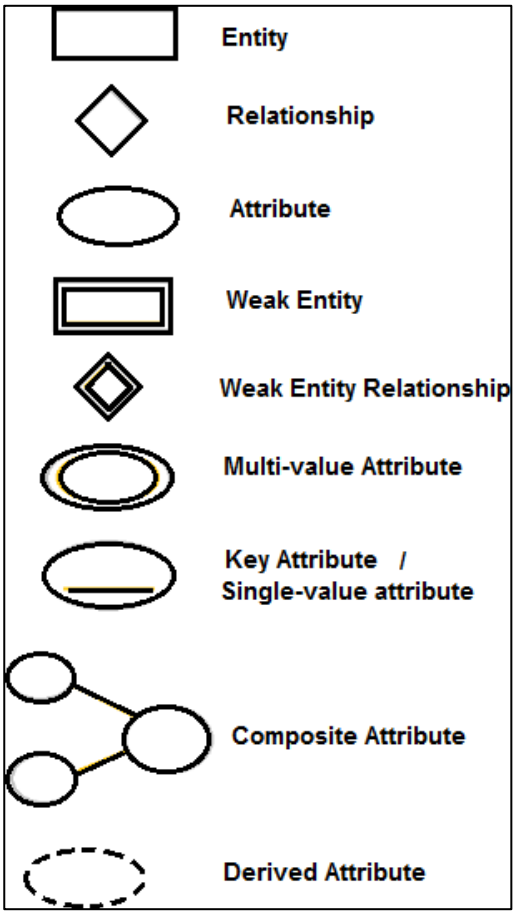
1.2 Introduction to the Courier Service Management System

The system is a web application that can be accessed throughout the organization with proper login provided. This system can be used as a web application for the Admin of the courier service to manage the courier information of the customers. Admin login should be able to assign delivery person, add or remove item, view all the current and previous orders. The key feature of this project is that it is a onetime registration. Our project provides the facility of maintaining the details of the customers. Customer can login to check the status This project is aimed at developing a web application for the courier service. of their current courier delivery, or view their delivered couriers. Delivery man can login to update the status of the courier to be delivered. This project will aid courier service to practice full IT deployment. This will also help in fast access procedures in courier related activities

CHAPTER 2

ER DIAGRAM

An entity relationship model, also called an entity-relationship (ER) diagram, is a graphical representation of entities and their relationships to each other, typically used in computing in regard to the organization of data within databases or information systems. An entity is a piece of data-an object or concept about which data is stored.



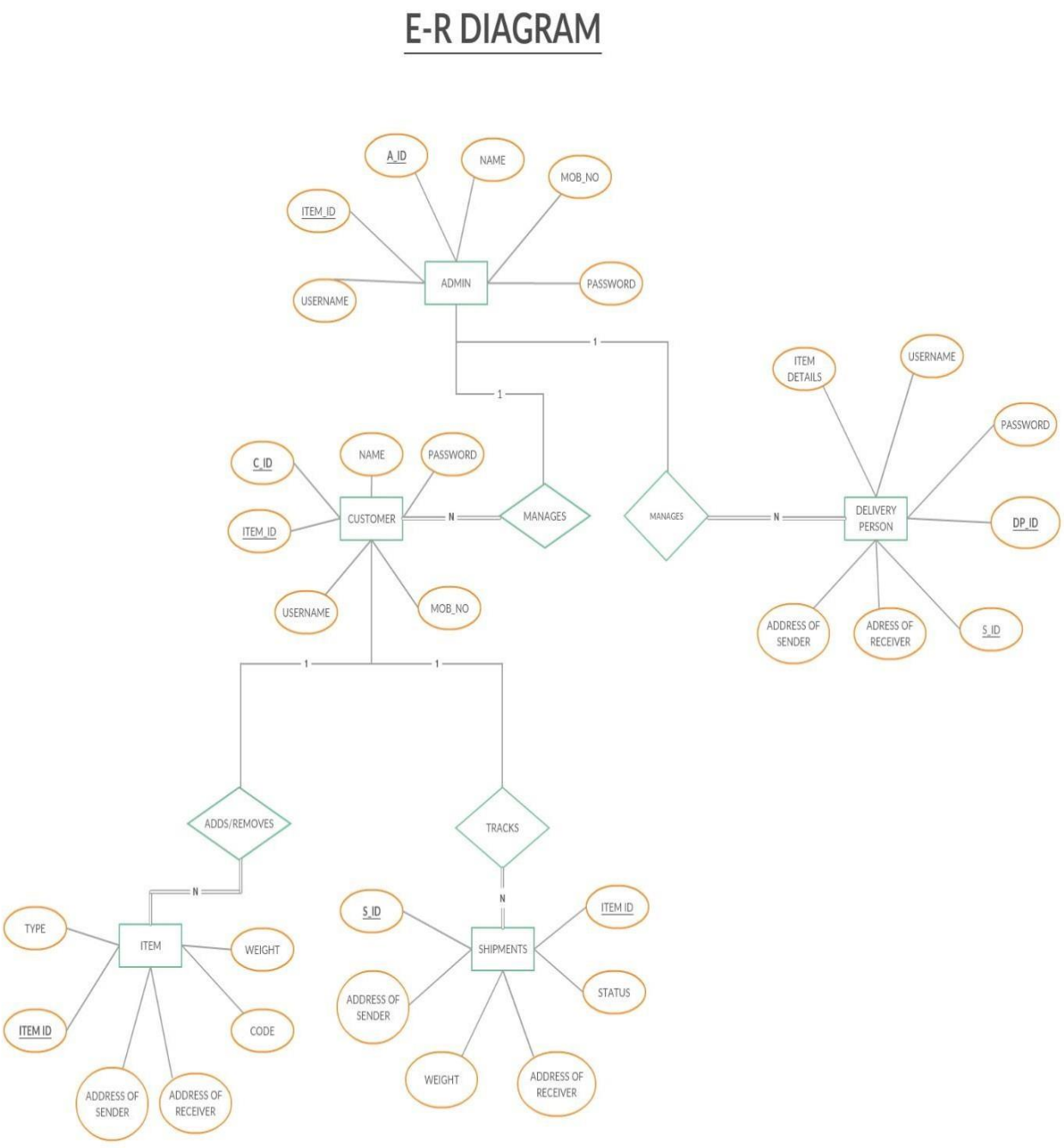


Fig.2 ER Diagram

As shown in the Fig 2, we have 5 entities in our ER Diagram namely Admin, Customer, Delivery Person, Item, Shipments

CHAPTER 3

RELATIONAL SCHEMA

A database schema is the skeleton structure that represents the logical view of the entire database. It defines how the data is organized and how the relations among them are associated. It formulates all the constraints that are to be applied on the data. A database schema defines its entities and the relationship among them. It contains a descriptive detail of the database, which can be depicted by means of schema diagrams. It's the database designers who design the schema to help programmers understand the database and make it useful.

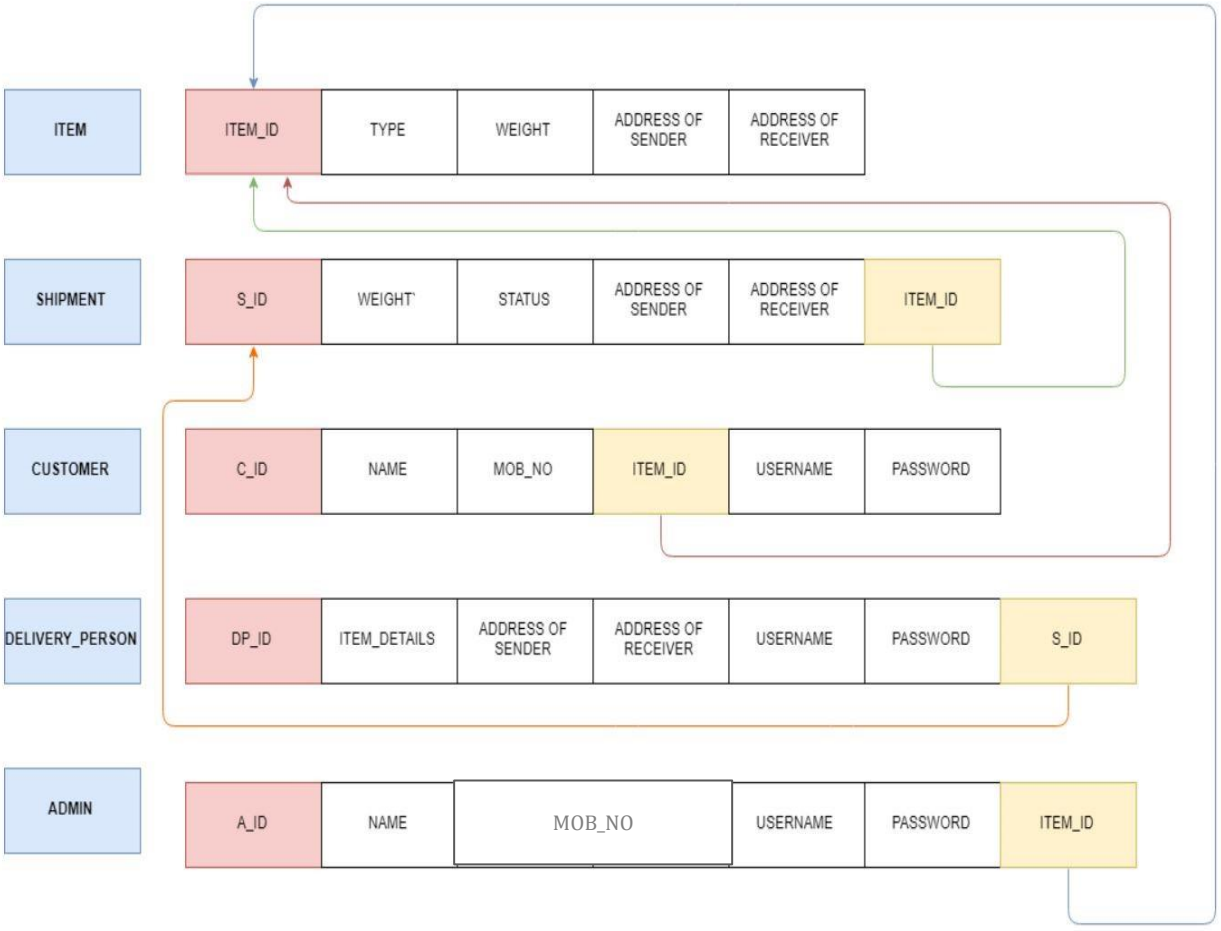


Fig.3 Relational Schema Diagram

As shown in the Fig 3, we have 5 tables in our Relation Schema Diagram Namely Item, Shipment, Customer, Delivery Person, Admin. The red Attributes represent the Primary key and the yellow Attributes represent the Foreign Key

A relational database schema is the tables, columns and relationships that make up a relational database. There are two steps to creating a relational database schema:

Creating the logical schema and Creating the physical schema. The logical schema depicts the structure of the database, showing the tables, columns and relationships with other tables in the database and can be created with modeling tools or spreadsheet and drawing software. The physical schema is created by actually generating the tables, columns and relationships in the relational database management software (RDBMS). Most modeling tools can automate the creation of the physical schema from the logical schema, but it can also be done by manually.

The structure of a relational database will inevitably change over time as data needs change. It's just as important to document changes to your relational database schema as it was to document the original design, otherwise it becomes increasingly difficult to update or integrate the database. Be sure to save copies of previous configurations, so that changes can be removed if problems occur.

They provide a high-level view as well as a detailed view of the database design. On a high level, they show the SQL developer which tables and columns are present, as well as their proper spelling. With regard to relationships, they illustrate the requirements for joining tables.

CHAPTER 4

SYSTEM DESIGN

4.1 Tables

```
mysql> desc item;
+--------------------+-------------+------+-----+---------+-------+
| Field              | Type        | Null | Key | Default | Extra |
+--------------------+-------------+------+-----+---------+-------+
| item_id            | varchar(6)  | NO   | PRI | NULL    |       |
| type               | varchar(20) | YES  |     | NULL    |       |
| weight             | varchar(4)  | YES  |     | NULL    |       |
| address_of_sender  | varchar(20) | YES  |     | NULL    |       |
| address_of_receiver| varchar(20) | YES  |     | NULL    |       |
+--------------------+-------------+------+-----+---------+-------+
```

Fig 4.1 Item table

This is the description of Item table.

```
mysql> desc customer;
+----------+-------------+------+-----+---------+-------+
| Field    | Type        | Null | Key | Default | Extra |
+----------+-------------+------+-----+---------+-------+
| c_id     | varchar(20) | NO   | PRI | NULL    |       |
| name     | varchar(20) | YES  |     | NULL    |       |
| mob_no   | varchar(10) | YES  |     | NULL    |       |
| item_id  | varchar(6)  | YES  | MUL | NULL    |       |
| username | varchar(20) | NO   |     | NULL    |       |
| password | varchar(20) | NO   |     | NULL    |       |
+----------+-------------+------+-----+---------+-------+
```

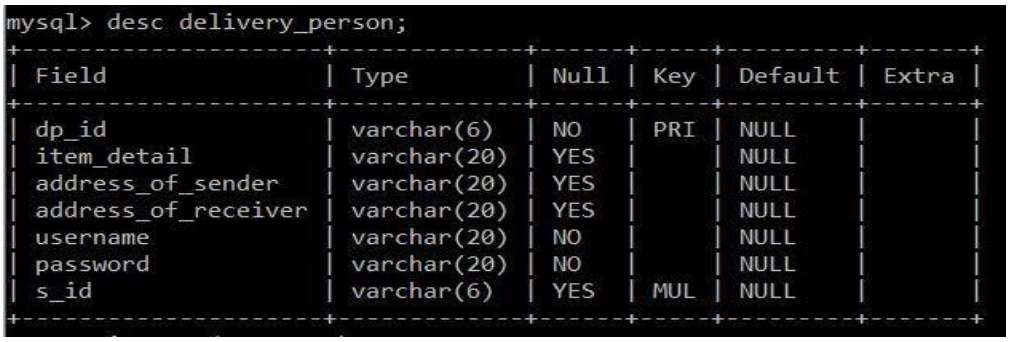
Fig 4.2 Customer Table

This is the description of customer table.

```
mysql> desc shipment;
+--------------------+-------------+------+-----+---------+-------+
| Field              | Type        | Null | Key | Default | Extra |
+--------------------+-------------+------+-----+---------+-------+
| s_id               | varchar(6)  | NO   | PRI | NULL    |       |
| weight             | varchar(4)  | YES  |     | NULL    |       |
| status             | varchar(20) | YES  |     | NULL    |       |
| address_of_sender  | varchar(20) | YES  |     | NULL    |       |
| address_of_receiver| varchar(20) | YES  |     | NULL    |       |
| item_id            | varchar(6)  | YES  | MUL | NULL    |       |
+--------------------+-------------+------+-----+---------+-------+
```

Fig 4.3 Shipment Table

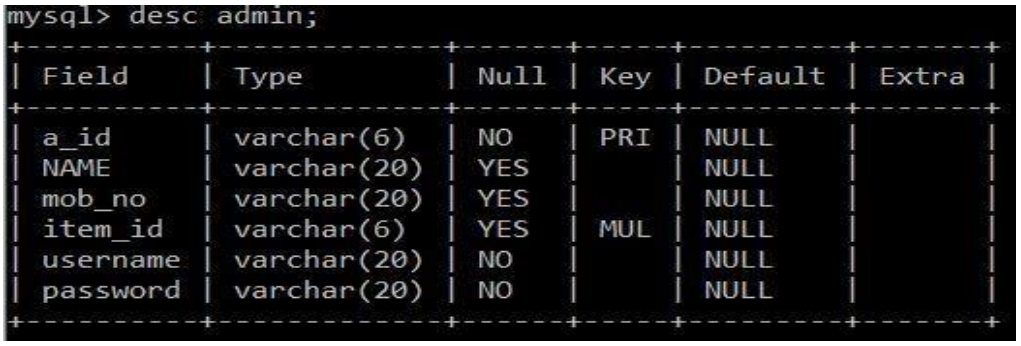
This is the description of Shipment table.



```
mysql> desc delivery_person;
+--------------------+-------------+------+-----+---------+-------+
| Field              | Type        | Null | Key | Default | Extra |
+--------------------+-------------+------+-----+---------+-------+
| dp_id              | varchar(6)  | NO   | PRI | NULL    |       |
| item_detail        | varchar(20) | YES  |     | NULL    |       |
| address_of_sender  | varchar(20) | YES  |     | NULL    |       |
| address_of_receiver| varchar(20) | YES  |     | NULL    |       |
| username           | varchar(20) | NO   |     | NULL    |       |
| password           | varchar(20) | NO   |     | NULL    |       |
| s_id               | varchar(6)  | YES  | MUL | NULL    |       |
+--------------------+-------------+------+-----+---------+-------+
```

Fig 4.4 Delivery Person Table

This is the description of Delivery Person table.



```
mysql> desc admin;
+----------+-------------+------+-----+---------+-------+
| Field    | Type        | Null | Key | Default | Extra |
+----------+-------------+------+-----+---------+-------+
| a_id     | varchar(6)  | NO   | PRI | NULL    |       |
| NAME     | varchar(20) | YES  |     | NULL    |       |
| mob_no   | varchar(20) | YES  |     | NULL    |       |
| item_id  | varchar(6)  | YES  | MUL | NULL    |       |
| username | varchar(20) | NO   |     | NULL    |       |
| password | varchar(20) | NO   |     | NULL    |       |
+----------+-------------+------+-----+---------+-------+
```

Fig 4.5 Admin Table

This is the description of admin table.

CHAPTER 5

IMPLEMENTATION

HTML - Hypertext Markup Language (HTML) is the standard markup language for creating web pages and web applications. With Cascading Style Sheets (CSS) and JavaScript, it forms a triad of cornerstone technologies for the World Wide Web. images and other objects such as interactive forms may be embedded into the rendered page. HTML provides a means to create structured documents by denoting structural semantics for text such as headings, paragraphs, lists, links, quotes and other items.HTML elements are delineated by tags, written using angle brackets. Tags such as <img/>and<input/> directly introduce content into the page. Other tags such as <p> surround and provide information about document text and may include other tags as sub-elements.

CSS - Cascading Style Sheets (**CSS**) is a style sheet language used for describing the presentation of a document written in a markup language like HTML. CSS is a cornerstone technology of the World Wide Web, alongside HTML and JavaScript CSS is designed to enable the separation of presentation and content, including layout, colors, and fonts. This separation can improve content accessibility, provide more flexibility and control in the specification of presentation characteristics, enable multiple web pages to share formatting by specifying the relevant CSS in a separate .css file, and reduce complexity and repetition in the structural content.

MY SQL - It is a relational database management system. As a database it's a software product whose primary function is to store & retrieve data as requested by other software applications, be it those on the same computer or those running on another computer across a network (including the internet).

JSP -It is used to create web application just like Servlet technology. It can be thought of as an extension to Servlet because it provides more functionality than servlet such as expression language, JSTL, etc.

5.1 HARDWARE AND SOFTWARE REQUIREMENTS

Software requirement: -

Operating System	:	Linux Windows
IDE	:	Eclipse
Database	:	MySQL
Front-End	:	HTML,CSS
Server Side Language	:	JSP
Server	:	Apache Tomcat 9

Hardware requirement:-

RAM: 2 GB

HDD: 40 GB

CPU: Pentium processor and above

5.2 CODE SEGMENT

Login Page:

```
<%@ page import ="java.sql.*" %>
<%@ page import =" java.sql.Connection" %>
<%@ page import ="java.sql.DriverManager" %>
<%@ page import ="java.sql.ResultSet" %>
<%@ page import ="java.sql.SQLException" %>
<%@ page import ="java.sql.Statement" %>
<%@ page import ="java.lang.*" %>
<%
try{
String username = request.getParameter("username"); String password =
request.getParameter("password"); Class.forName("com.mysql.jdbc.Driver");
Connectionconn DriverManager.getConnection("jdbc:mysql://localhost:3306/dbms" , "root" ,
"Adity@05");
PreparedStatement pst = conn.prepareStatement("select username,password from admin where
username=? and password=?");
pst.setString(1, username); pst.setString(2, password); ResultSet rs = pst.executeQuery();

if(rs.next()){ RequestDispatcher rd =
request.getRequestDispatcher("welcome-admin.html"); rd.forward(request, response);
}
else{
%>
<h1>You have entered incorrect password</h1>
<% RequestDispatcher rd = request.getRequestDispatcher("adminlogin.html");
rd.include(request, response);
}
pst.close();
conn.close();
}
catch(Exception e){
out.println("Something went wrong !! Please try again ");
%>
<br>
<a href="index.html"><button>Back</button></a>
<%
}
%>
```

Adding user item:

```
<style>
h3{
text-align:center;
font-style:italic;
background-color:black;
opacity:0.9;
color:white;
}
div{
text-align:center;
font-size:30px;
}
.button
{
width: 150px;
margin-left:45%;

}
#submission {
text-align: center;
}
</style>
<!DOCTYPE html>
<html>
<head>
<meta charset="ISO-8859-1">
<title>Removed Dp</title>
</head>
<body style="background-image:url(user.jpg);">
<h3>Notification</h3>
<%@ page import ="java.sql.*" %>
<%@ page import =" java.sql.Connection" %>
<%@ page import ="java.sql.DriverManager" %>
<%@ page import ="java.sql.ResultSet" %>
```

```
<%@ page import ="java.sql.SQLException" %>
<%@ page import ="java.sql.Statement" %>
<%@ page import ="java.lang.*" %>
<%
try{
String itemid=request.getParameter("itemid");
String custid=request.getParameter("custid");
String itemtype=request.getParameter("itemtype");
String itemweight=request.getParameter("itemweight");
String itempickupadd=request.getParameter("itempickupadd");
String itemdropadd=request.getParameter("itemdropadd");
Class.forName("com.mysql.jdbc.Driver");
Connection conn = DriverManager.getConnection("jdbc:mysql://localhost:3306/dbms" ,
"root" , "Adity@05");
Statement st=conn.createStatement();
int i=st.executeUpdate("insert into
item(item_id,type,weight,address_of_sender,address_of_receiver)values('"+itemid+"','"
+itemtype+"','"+itemweight+"','"+itempickupadd+"','"+itemdropadd+"')");
out.println("<span style=\"font-weight:bold; color:black;font-size:30px \"> Data
Inserted successfully </span>");
%>
<a href="welcome-user.html" id="submission"><button
class="button">Back</button></a>
<%
}
catch(Exception e)
{
out.println("<span style=\"font-weight:bold; color:black;font-size:30px \">Something
went wrong.Please Try Agin.. </span>" + e);
%>
<a href="welcome-user.html" id="submission">
<button class="button">Back</button></a>
<%
}
%>
```

Stored Procedure Code:

```
delimiter $$
create procedure showall()
select c.c_id,c.name,c.mob_no,i.item_id,i.type,i.weight,i.address_of_sender,
i.address_of_receiver,s.status
from shipment s,customer c,item i
where
i.item_id=c.item_id
and
s.item_id=i.item_id$$
```

Calling Stored Procedure From JSP:

```
Class.forName("com.mysql.jdbc.Driver");
Connection conn = DriverManager.getConnection("jdbc:mysql://localhost:3306/dbms" ,
"root" , "Adity@05");
 CallableStatement pst = conn.prepareCall("{call showall()}");
 ResultSet rs = pst.executeQuery();
```

The Stored Procedure is used to display all the orders placed.

Trigger Code:

```
delimiter //
create trigger weightcheck
before
 insert on item
for each row
if new.weight <0 then
set new.weight =0;
end if;//
delimiter;
```

The Trigger is used to set negative weights to zero.

5.3 SNAPSHOTS

Fig 5.1 Welcome Page

This page provides the option of login and sign up page for both user and publisher

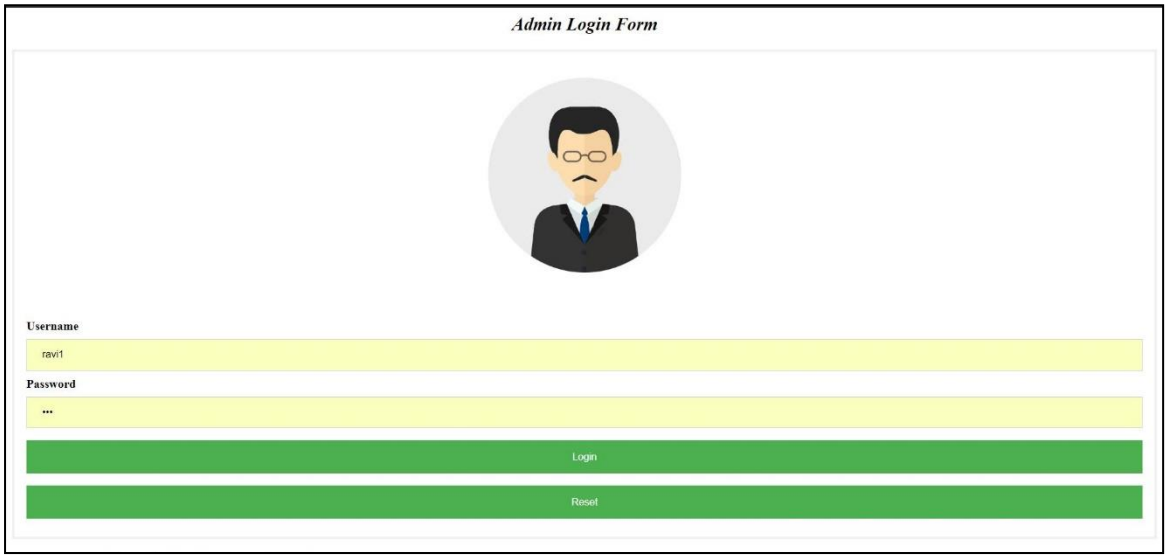


Fig 5.2 Admin Login

This page is the login page for Admin where the admin enters their username and password.

Fig 5.3 Customer Login

This page is the login page for user where the customer enters their username and password

Fig 5.2 Delivery Person

This page is the login page for Delivery Man where they enter their username and password**.**

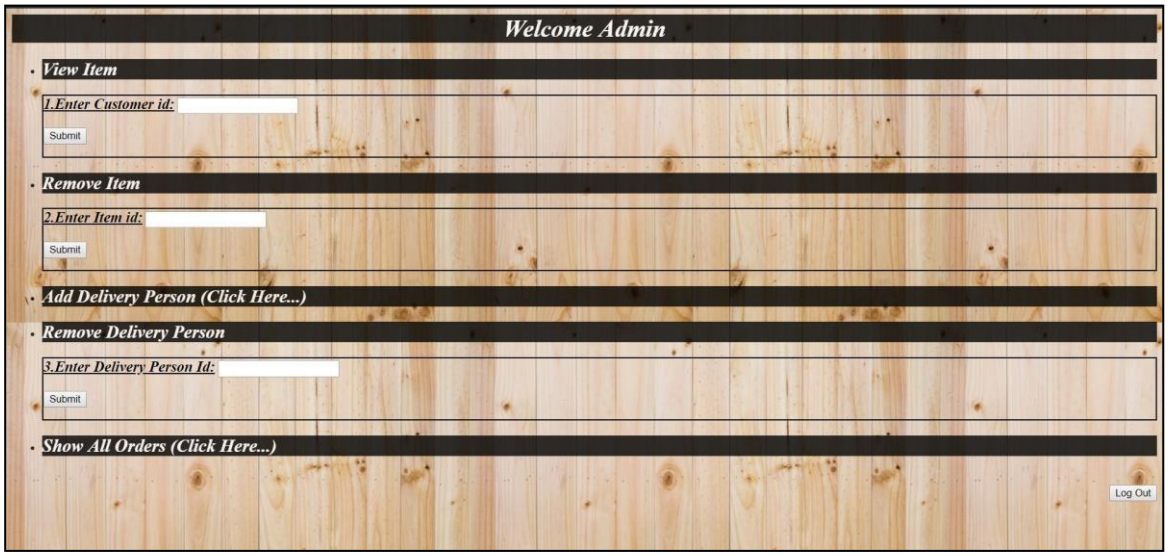


Fig 5.5 Admin Dashboard

This page gives admin the options to view a customer's order, delete an item, add delivery person, remove delivery person or check all orders

Fig 5.6 Order Details

It fetches the details of specific customer's order.

Fig 5.7 Adding Delivery Person

This page allows the admin to add new delivery person. On Successful addition of delivery person, it generates a notification page.

All Orders Are:

CUSTOMER ID	CUSTOMER NAME	CUSTOMER MOBILE NUMBER	ITEM ID	TYPE	WEIGHT(IN KG)	ADDRESS OF SENDER	ADDRESS OF RECEIVER	STATUS
ct11	sohan	962021345	it11	accessories	40	patna	bangalore	In Transit
ct12	ramesh	967544322	it12	accessories	80	ranchi	kolkata	In Transit
ct13	vivek	636134256	it13	furniture	100	mysore	kolkata	delivered
ct14	kovid	234569620	it14	furniture	300	patna	mysore	Unpicked
ct15	ajay	196202176	it15	electronics	120	patna	ranchi	delivered
ct16	karthik	196202179	it16	cloths	100	hyderabad	newdelhi	Unpicked
ct3	sanjana	988907656	it3	electronics	30	hyderabad	bangalore	Unpicked
ct6	sonam	775612023	it6	electronics	30	kanpur	assam	Delivered
ct9	satyam	998677653	it9	stationary	150	indore	patna	Unpicked

Back

Fig 5.8 All Orders

Details of all placed orders using select query.

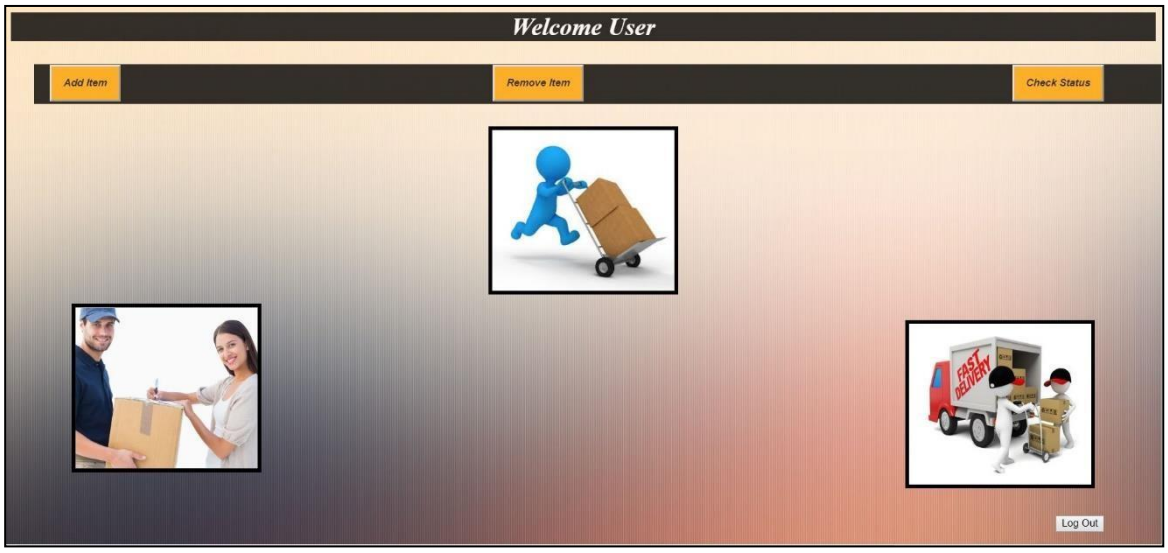


Fig 5.9 Customer Dashboard

This page gives user the options to add item, remove item or check status of the order.

Fig 5.10 Placing the Order.

Customer needs to enter these details to place the courier: type of item, weight of item, pickup address, drop address.

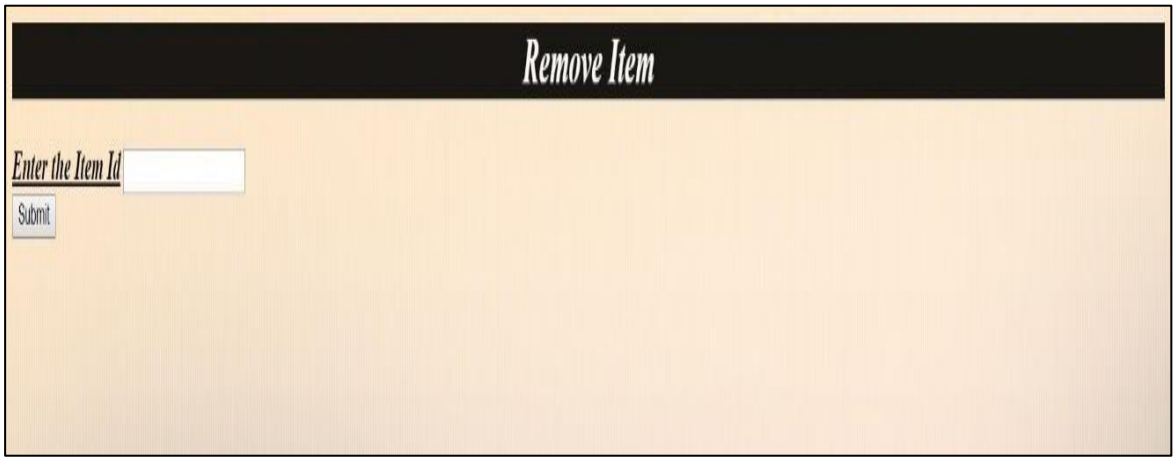


Fig 5.11 Removing the Order

For removing the order customer only need to specify item id.

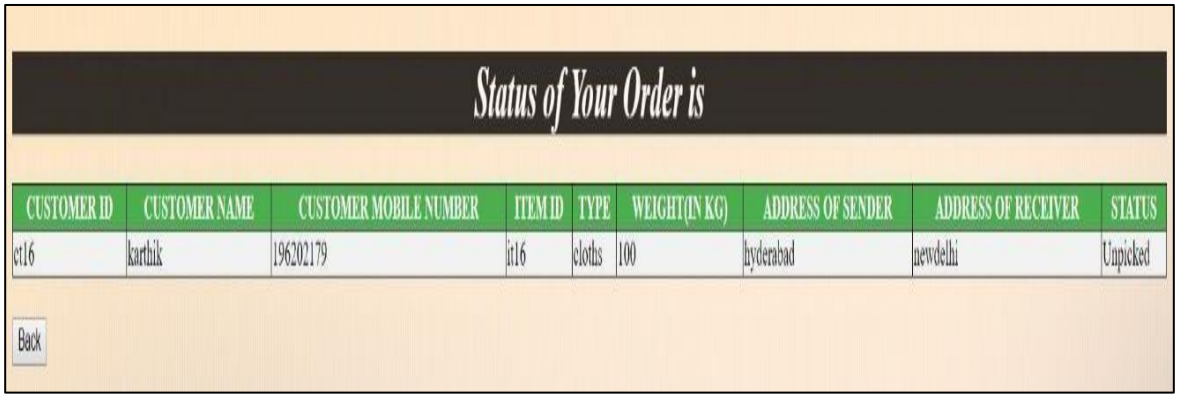


Fig 5.12 Getting the status of order.

Checking the status of item, customer only need to enter item id.

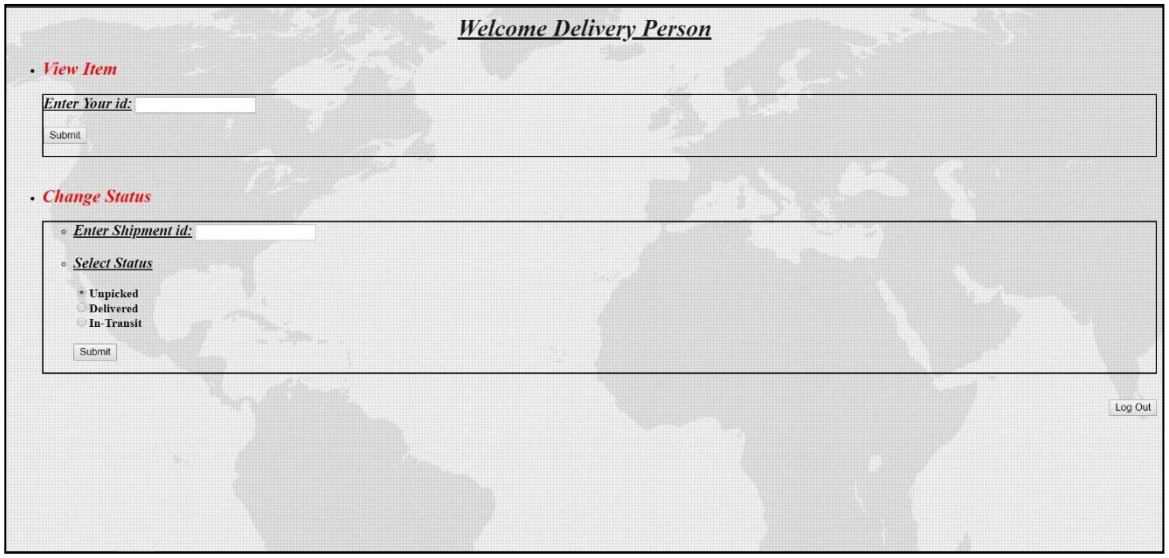


Fig 5.13 Delivery Person Dashboard

This page gives delivery person the options to view the order assigned and change the status of order.

Fig 5.14 Checking Received Order

Delivery Person can see the couriers that they have to deliver.

Fig 5.15 Changing the status of order

Delivery Person can update the status of their order by specifying the shipment id.

Chapter 6

CONCLUSION AND FUTURE ENHANCEMENT

CONCLUSION

This project will replace the traditional way of courier distribution. The couriers were distributed in a manual fashion and that can be replaced by faster and more effective system. This platform eliminates this problem as it saves the data online and it reduces the redundancy. This project has very high potential to grow in future.

While implementing this project we came to know the real-world problems and found a solution to the problem. During this project we learned about the basic Database Management System and how to implement it for real world problems.

To implement the idea behind the project we also learned basic JSP for back- end development and HTML and CSS for front-end development. we now feel quite confident in the mentioned languages and can implement other projects using these languages.

Courier Management System which supports the high accessibility of courier services to the corporate and to the customer. The system is being used for day to day activities such as booking a courier, maintain hub details, maintain company details, process data of businesses and many other things.

Courier management computerization is “the incorporation of appropriate technology to help manager manage information. Technology is considered suitable when it utilizes the most abundant domestic possessions and conserves investment and skilled personnel”

FUTURE ENHANCEMENT

In a nutshell, it can be summarized that the future scope of the project circles around maintaining information regarding:

- Integrate the system into now evolving e-commerce industries.
- Develop advance software for Courier Management System.
- Host the platform on online servers to make it accessible worldwide.
- Integrate Multiple load balancers to distribute the loads of the system.
- Implement a notification system via SMS to notify customer about the order.
- Implement the backup mechanism for taking backup of codebase and database on regular basis on different servers.

REFERENCES

- Fundamentals of Database Systems (6th Edition) by Ramez Elmasri and Shamkant Navathe
- Database management systems (3rd edition) by Raghu Ramakrishna
- www.w3schools.com
- www.youtube.com/telusko
- www.stackoverflow.com