VISVESVARAYA TECHNOLOGICAL UNIVERSITY

BELAGAVI-590018

A DBMS Mini-Project Report
On

"Courier Management System"

Submitted in partial fulfillment of the requirements for the 5th semester of
Bachelor of Engineering in Computer Science and Engineering
of Visvesvaraya Technological University, Belagavi

Submitted by:

VARUN VISHNU MURTHY THANTRY	**1RN19CS175**
YESHWANTH P L	**1RN19CS190**

Under the Guidance of:

Mr. Karanam Sunil Kumar
Assistant Professor
Dept. of CSE

Dr. H R Shashidhara
Associate Professor
Dept. of CSE

Department of Computer Science and Engineering
RNS Institute of Technology
Channasandra, Dr.Vishnuvardhan Road, Bengaluru-560 098

2021-2022

RNS Institute of Technology

Channasandra, Dr.Vishnuvardhan Road,
Bengaluru-560 098

DEPARTMENT OF COMPUTER SCIENCE & ENGINEERING

CERTIFICATE

Certified that the DBMS mini-project work entitled **"Courier Management System"** has been successfully carried out by **Varun Vishnu Murthy Thantry** bearing USN **1RN19CS175** and **Yeshwanth P L** bearing USN **1RN19CS190**, bonafide students of **RNS Institute of Technology** in partial fulfillment of the requirements for the **5th semester Bachelor of Engineering** in **Computer Science and Engineering** of **Visvesvaraya Technological University**, Belagavi, during the academic year 2021-2022. It is certified that all corrections/suggestions indicated for Internal Assessment have been incorporated in the report. The project report has been approved as it satisfies the mini-project requirements of the DBMS lab of 5th semester BE in CSE.

Mr. Karanam Sunil Kumar	**Dr. H R Shashidhara**	**Dr. Kiran P**
Assistant Professor	**Associate Professor**	**Prof. and Head**
Dept. of CSE	**Dept. of CSE**	**Dept of CSE**

External Viva:

Name of the Examiners **Signature with Date**

1.

2.

ACKNOWLEDGMENTS

Any achievement, be it scholastic or otherwise does not depend solely on the individual efforts but on the guidance, encouragement and cooperation of intellectuals, elders and friends. A number of personalities, in their own capacities have helped us in carrying out this project work. We would like to take this opportunity to thank them all.

We are grateful to **Dr. M K Venkatesha,** Principal, RNSIT, Bangalore, for his support towards completing this mini project.

We would like to thank **Dr. Kiran P** Prof. & Head, Department of Computer Science & Engineering, RNSIT, Bangalore, for his valuable suggestions and expert advice.

We deeply express our sincere gratitude to our guide **Mr. Karanam Sunil Kumar**, Assistant Professor, **Dr. H R Shashidhara**, Associate Professor, Department of CSE, RNSIT, Bangalore, for their able guidance, regular source of encouragement and assistance throughout this project.

We would like to thank all the teaching and non-teaching staff of department of Computer Science & Engineering, RNSIT, Bengaluru for their constant support and encouragement.

Date :
Place : Bengaluru

Varun Vishnu Murthy Thantry 1RN19CS175
Yeshwanth P L 1RN19CS190

ABSTRACT

This project deals with the ‘Courier Management System’. The system is used for daily cargo office activities such as booking courier; maintain employee details, process pay roll of employees, maintain branch details etc. It is very difficult to do this process manually. Hence it is recommended to computerize this process by developing the relative software as the world is turning into information and technology. The courier management system will speed up the process of booking and delivering the package.

CONTENTS

CHAPTER 1

INTRODUCTION

1.1 DATABASE TECHNOLOGIES

The essential feature of database technology is that it provides an internal representation (model) of the external world of interest. Examples are, the representation of a particular date/time/flight/aircraft in an airline reservation or of the item code/item description/quantity on hand/reorder level/reorder quantity in a stock control system.

The technology involved is concerned primarily with maintaining the internal representation consistent with external reality; this involves the results of extensive R&D over the past 30 years in areas such as user requirements analysis, data modeling, process modeling, data integrity, concurrency, transactions, file organization, indexing, rollback and recovery, persistent programming, object-orientation, logic programming, deductive database systems, active database systems... and in all these (and other) areas there remains much more to be done. The essential point is that database technology is a CORE TECHNOLOGY which has links to:

- Information management / processing
- Data analysis / statistics
- Data visualization / presentation
- Multimedia and hypermedia
- Office and document systems
- Business processes, workflow, CSCW (computer-supported cooperative work)

Relational DBMS is the modern base technology for many business applications. It offers flexibility and easy-to-use tools at the expense of ultimate performance. More recently relational systems have started extending their facilities in directions like information retrieval, object-orientation and deductive/active systems which lead to the so-called 'Extended Relational Systems'. Information Retrieval Systems began with handling library catalogues and then extended to full free-text by utilizing inverted index technology with a lexicon or thesaurus.

Modern systems utilize some KBS (knowledge-based systems) techniques to improve the retrieval.

Object-Oriented DBMS started for engineering applications in which objects are complex, have versions and need to be treated as a complete entity. OODBMSs share many of the OOPL features such as identity, inheritance, late binding, overloading and overriding. OODBMSs have found favours in engineering and office systems but haven't been successful yet in traditional application areas. Deductive / Active DBMS has evolved over the last 20 years and combines logic programming technology with database technology. This allows the database itself to react to the external events and also to maintain its integrity dynamically with respect to the real world.

1.2 CHARACTERISTICS OF DATABASE APPROACH

A modern DBMS has the following characteristics −

- Real-world entity − A modern DBMS is more realistic and uses real-world entities to design its architecture. It uses behaviour and attribute too. For example, a school database may use students as an entity and their age as an attribute.

- Relation-based tables − DBMS allows entities and relations to form tables.
  A user can understand the architecture of a database by just looking at the table names.

- Isolation of data and application − A database system is entirely different than its data. A database is an active entity, whereas data is said to be passive, on which the database works and organizes. DBMS also stores metadata, which is data about data, to ease its own process.

- Less redundancy − DBMS follows the rules of normalization, which splits a relation when any of its attributes has redundancy in its values. Normalization is a mathematically rich and scientific process that will reduces the data redundancy.

- Consistency − Consistency is a state where every relation in a database remains consistent. There exists methods and techniques, that can detect an attempt of leaving database in an inconsistent state. DBMS can provide greater consistency as compared to earlier forms of data storing applications like file-processing systems.

- Query Language − DBMS is equipped with query language, which makes it more efficient to retrieve and manipulate data. A user can apply as many and the filtering options as required to retrieve a set of data. Traditionally it was not possible where file-processing system was used.

- ACID Properties − DBMS follows the concepts of Atomicity, Consistency, Isolation, and Durability (normally shortened as ACID). These concepts are applied on transactions, which manipulate data in a database. ACID properties help the database to stay healthy in multi-transactional environments and also in case of failure.

- Multiuser and Concurrent Access − DBMS supports multi-user environment and allows them to access and manipulate data in parallel. Though there are restrictions on transactions when users attempt to handle the same data item, but users are always unaware of them.

- Multiple views − DBMS offers multiple views for different users. A user in the Sales department will have a different view of the database from the person working in the Production department. This feature enables the users to have a concentrate view of the database according to their requirements.

- Security − Features like multiple views offer security to certain extent when users are unable to access the data of other users and departments. DBMS offers methods to impose constraints while entering data into the database and retrieving the same at a later stage. DBMS offers many different levels of security features, which enables multiple users to have different views with different features. For example, a user in the Sales department cannot see the data that belongs to the Purchase department. It can also be helpful in deciding how much data of the Sales department should be displayed to the user. Since a DBMS is not saved on the disk as traditional file systems, it is very hard for miscreants to break the code.

1.3 APPLICATIONS OF DBMS

Applications of Database Management Systems :

- **Telecom**: There is a database to keeps track of the information regarding the calls made, network usage, customer details etc. Without the database system it is hard to maintain such huge amounts of data which gets updated every millisecond.

- **Industry**: Whether it is a manufacturing unit, a warehouse or a distribution centre, each one needs a database to keep the records of the ins and outs. For example, a distribution centre should keep a track of the product units that were supplied to the centre as well as the products that got delivered from the distribution centre on each day; this is where DBMS comes into picture.
- **Banking System**: For storing information regarding a customer, keeping a track of his/her day to day credit and debit transactions, generating bank statements etc is done with through Database management systems.

- **Education sector**: Database systems are frequently used in schools and colleges to store and retrieve the data regarding the student , staff details, course details, exam details, payroll data, attendance details, fees details etc. There is lots of inter-related data that needs to be stored and retrieved in an efficient manner.

- **Online shopping**: You must be aware of the online shopping websites such as Amazon, Flip kart etc. These sites store the product information, your addresses and preferences, credit details and provide you the relevant list of products based on your query. All this involves a Database management system.

1.4 PROBLEM DESCRIPTION/STATEMENT

The purpose of Courier Management system is to provide a platform for Customers and Employee to manage the booking and delivering process of the courier seamlessly. Courier System helps in transporting so as to decrease the load for manual transportation.

Keeping that in mind, we have built a platform that can be used by both the customer and the Employee. The customer can book a courier from his place to receiver's address and track the courier. Employee can view the courier details of the customer which consists of source, destination and consignment number and looks after the customer's booking and can update the status accordingly.

CHAPTER 2

REQUIREMENT ANALYSIS

2.1 HARDWARE REQUIREMENTS

The Hardware requirements are very minimal and the program can be run on most of the machines.

Processor	:	Pentium4 processor
Processor Speed	:	2.4 GHz
RAM	:	1 GB
Storage Space	:	40 GB
Monitor Resolution	:	1024*768 or 1336*768 or 1280*1024

2.2 SOFTWARE REQUIREMENTS

1. Operating System used: Windows 10
2. Brackets Text Editor: HTML, CSS, PHP
3. WAMPP Server: MySQL, PhpMyAdmin
4. IDE used : NetBeans 8.2
5. Browser that supports HTML

2.3 FUNCTIONAL REQUIREMENTS

2.3.1 MAJOR ENTITIES

Customer: customer is the entity who is going to book a courier from his place to receiver's place. Few attributes are id, name, phone etc.

Branch Manager: Branch Manager(employee) is the entity that looks after the entire process of courier services. Few attributes are id, name, phone ,salary etc

Admin: Admin is the entity who can view courier details, employee details, add new staff etc. Few attributes are id, name, password, mobile.

2.3.2 End User Requirements

The technical requirements for the project are mentioned below.

2.3.3.1 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. Web browsers receive HTML documents from a web server or from a local storage and render them to multimedia web pages. HTML describes the structure of a web page semantically and originally included cues for the appearance of the document.

HTML elements are the building blocks of HTML pages. With HTML constructs, images and other objects like interactive forms can be embedded into the rendered page. It provides a way to create structured documents by denoting structural semantics for the text

like headings, paragraphs, lists, links, quotes and other items. HTML elements are delimited by tags that are written within angle brackets. Tags such as <img /> and <input /> introduce content into the page directly. Other tags such as ... surround and provide information about document text and may include other tags as sub-elements. Browsers do not display the HTML tags, but use them to interpret the content of the page.

HTML can also embed programs written in a scripting language such as JavaScript which affect the behaviour and content of web pages. Inclusion of CSS defines the look and layout of content.

2.3.3.2 CSS

Cascading Style Sheets (CSS) is a style sheet language which is used for describing the presentation of a document written in a markup language. Although most often its used to set the visual style of web pages and user interfaces written in HTML and XHTML, the language can be applied to any XML document, including plain XML, SVG and XUL, and is also applicable to rendering in speech, or on other media. Along with HTML and JavaScript, CSS is a cornerstone technology used by most websites to create visually engaging webpages, user interfaces for web applications, and user interfaces for many mobile applications.

CSS is designed primarily to enable the separation of presentation and content,
including aspects such as the layout, colours, and fonts. This separation can improve content accessibility, provide more flexibility and control in the specification of presentation characteristics, enable multiple HTML pages to share the formatting by specifying the relevant CSS in a separate .css file, and reduce complexity and repetition in the structural content.

2.3.4 PHP

PHP is a server-side scripting language designed primarily for web development but is also used as a general-purpose programming language. Originally created by Rasmus Lerdorf in 1994, the PHP reference implementation is now produced by The PHP Development Team. PHP originally stood for Personal Home Page, but it now stands for the recursive acronym PHP: Hypertext Pre-processor.

PHP code can be embedded into HTML or HTML5 markup, or it can be used in combination with various web template systems, web content management systems and web frameworks. 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 software 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 can also be executed with a command-line interface (CLI) and can be used to implement standalone graphical applications.

The standard PHP interpreter, powered by the Zend Engine, is a free software released under the PHP License. PHP has been widely ported and can be deployed on most web servers, on almost every operating system and platform, free of charge. The PHP language evolved without a written formal specification or standard until 2014, leaving the canonical PHP interpreter as a de facto standard. Since 2014 work has gone into creating a formal PHP specification. HP development began in 1995 when Rasmus Lerdorf wrote several Common Gateway Interface (CGI) programs in C, which he used in order to maintain his personal homepage. He extended them to work with web forms and to communicate with databases, and called this implementation "Personal Home Page/Forms Interpreter" or PHP/FI.

PHP/FI could help to build simple, dynamic web applications. To accelerate bug reporting and to improve the code, Lerdorf initially announced the release of PHP/FI as "Personal

Home Page Tools (PHP Tools) version 1.0" on the Usenet discussion group on June 8, 1995 This release already had the basic functionality that PHP has as of 2013. This included Perl-like variables, form handling, and the ability to embed HTML. The syntax resembled that of Perl but was simpler, more limited and less consistent.

2.3.5 MySQL

MySQL is a Relational Database Management System (RDBMS). MySQL server can manage many databases at the same time. In fact, many people might have different databases managed by a single MySQL server. Each database consists of a structure to hold onto the data itself. A data-base can exist without data, only a structure, be totally empty, twiddling its thumbs and waiting for data to be stored in it.

Data in a database is stored in one or more tables. You must create the data-base and the tables before you can add any data to the database. First you create the empty database. Then you add empty tables to the database. Database tables are organized in rows and columns. Each row represents an entity in the database, such as a customer, a book, or a project. Each column contains an item of information about the entity, such as a customer name, a book name, or a project start date. The place where a particular row and column intersect, the individual cell of the table, is called a field. Tables in databases can be related. Often a row in one table is related to several rows in another table. For instance, you might have a database containing data about books you own. You would have a book table and an author table. One row in the author table might contain information about the author of several books in the book table. When tables are related, you include a column in one table to hold data that matches data in the column of another table.

MySQL, the most popular Open Source SQL database management system, is developed, distributed, and supported by MySQL AB. MySQL AB is a commercial company, founded by the MySQL developers. It is a second generation Open Source company that unites Open Source values and methodology with a successful business model.

- MySQL is a database management system. A database is a structured collection of data. It can be anything from a simple shopping list to a picture gallery or the vast amount of information in a corporate network. To add, access, and process data stored in a computer database, you need a database management system such as MySQL Server. Since computers are very good at handling

large amounts of data, database management systems play a central role in computing, as standalone utilities, or as parts of other applications.

- MySQL is a relational database management system. A relational database stores data in separate tables rather than putting all the data in one big storeroom. This adds speed and flexibility. The SQL part of "MySQL" stands for "Structured Query Language." SQL is the most common standardized language used to access databases and is defined by the ANSI/ISO SQL Standard. The SQL standard has been evolving since 1986 and several versions exist. "SQL-92" refers to the standard released in 1992, "SQL:1999" refers to the standard released in 1999, and "SQL:2003" refers to the current version of the standard. We use the phrase "the SQLstandard" to refer to the current version of the SQL
  Standard.

  MySQL software is Open Source. Open Source means that it is possible for anyone to use and modify the software. Anybody can download the MySQL software from the Internet and use it without paying anything. If you wish, you may study the source code and change it to suit your needs. The MySQL software uses the GPL (GNU General Public License), to define what you may and may not do with the software in different situations.

The MySQL Database Server is very fast, reliable, and easy to use.

MySQL Server was originally developed to handle large databases and has been successfully used in highly demanding production environments for several years. MySQL Server today offers a rich and useful set of functions. Its connectivity, speed, and security make MySQL Server highly suited for accessing databases on the Internet.

- MySQL Server works in a client/server or embedded systems. The MySQL Database Software is a client/server system which consists of a multi-threaded SQL server that supports different back ends, several different client programs and libraries, administrative tools, and a wide range of application programming interfaces (APIs).

2.3.6 XAMPP SERVER

Xampp server installs a complete, ready-to-use development environment. Xampp server allows you to fit your needs and allows you to setup a local server with the same characteristics as your production.

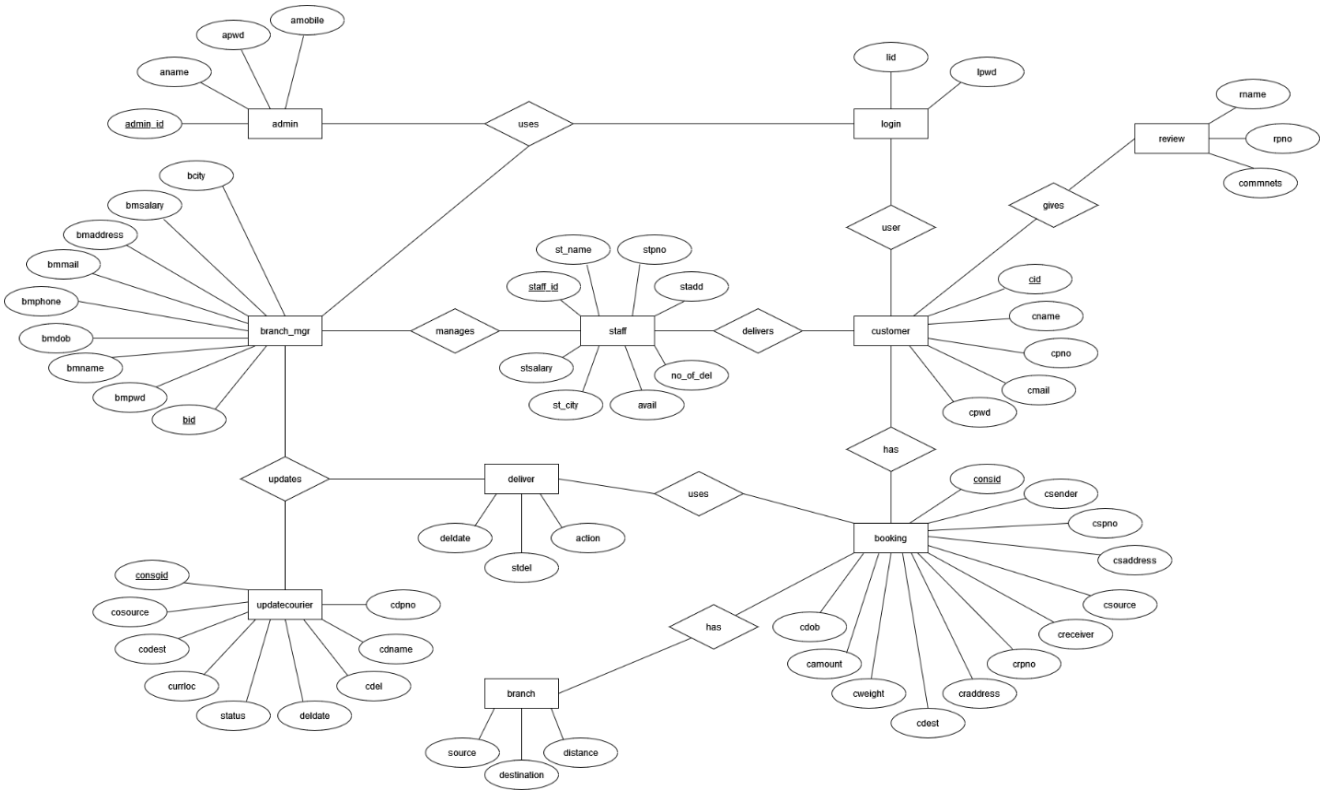
While setting up the server and PHP on your own, you have two choices for the method of connecting PHP to the server. For many servers, PHP has a direct module interface (also called SAPI). These servers include Apache, Microsoft Internet Information Server, Netscape and iPlanet servers. Many other servers support ISAPI, the Microsoft module interface (OmniHTTPd for example). If PHP has no module support for your web server, you can always use it as a CGI or FastCGI processor. This means you set up your server to use the CGI executable of PHP to process all PHP file requests on the server.

CHAPTER 3

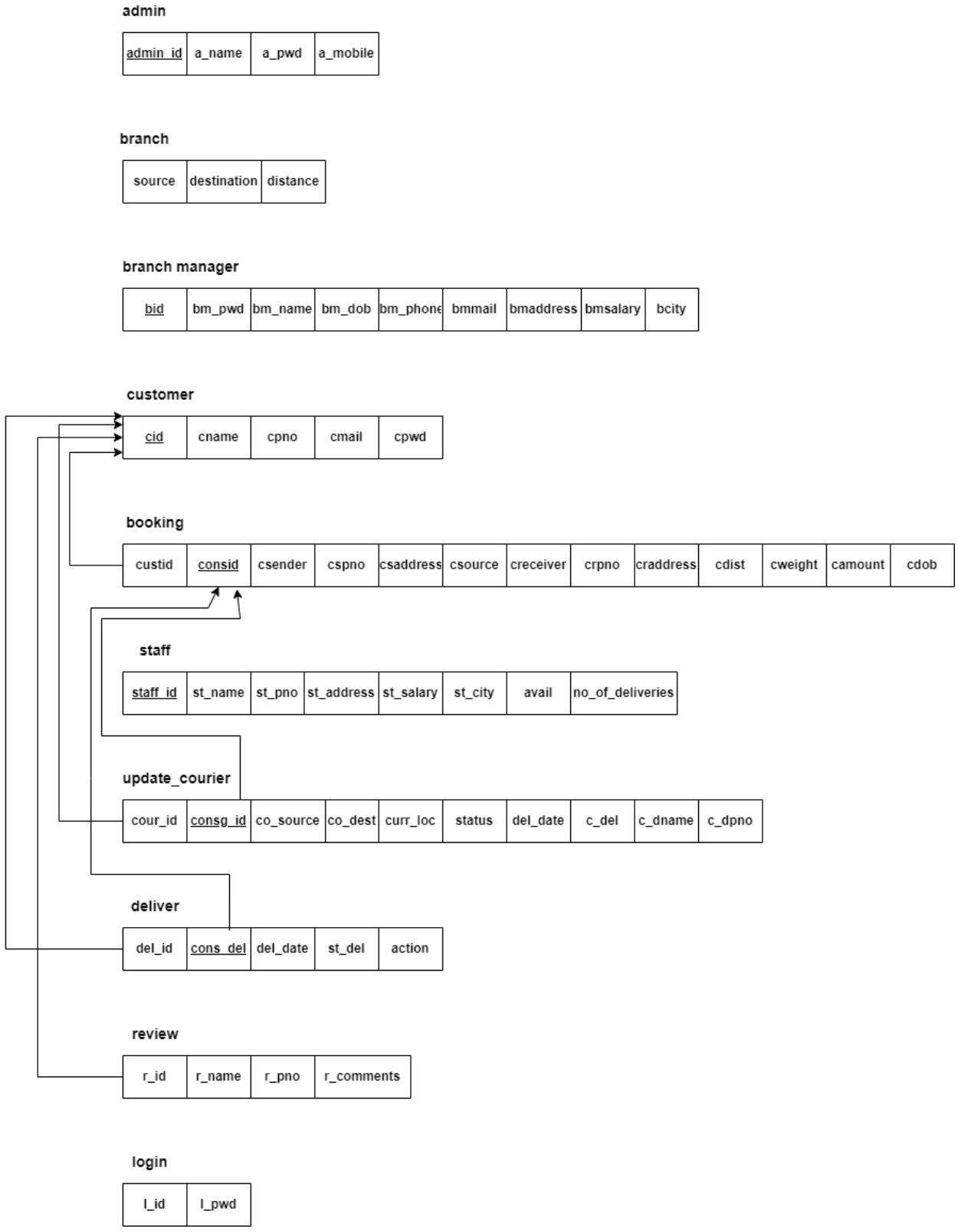
DATABASE DESIGN

3.1 Entities, attributes and relationships

Along with the above mentioned major entities, Courier Management System contains many other entities too. Their attributes are as follows:



3.4 Relational Schema



CHAPTER 4

IMPLEMENTATION

4.1 Creating Database Connection

1. PHP provide built-in database connectivity for a wide range of databases – MySQL, PostgreSQL, Berkeley DB, Informix, Lotus Notes and more

2. Use either mysql connect or mysql pconnect to create database connection

3. mysql connect: connection is closed at end of script(end of page)

4. mysql pconnect: creates persistent connection – connection remains ever after end of the page

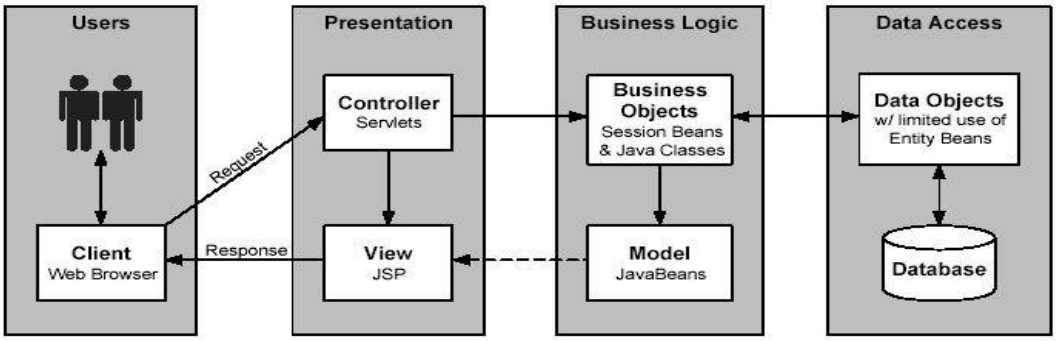
5. Connect to the MySQL server
   $connection=mysql_connect("localhost", $username,$password);

6. Access the database
   mysql_select_db("databasename",$connection);

7. Perform SQL operations
   Example: $result = mysql_query($query,$connection)

8. Disconnect from the server
   mysql_close($connection);

Architecture used (4-Tier Architecture)



Four Tier architecture is a client–server architecture in which presentation, application processing, and data management functions are physically separated. Four-tier application architecture provides a model by which developers can create flexible and reusable applications. By segregating an application into tiers, developers acquire the option of modifying or adding a specific layer, instead of reworking the entire application.

Presentation Layer

This is the topmost level of the application. The presentation tier displays information related to services such as browsing merchandise, purchasing and shopping cart contents. It also communicates with other tiers and puts out the results to the browser/client tier and to all other tiers in the network. In simple terms, it is a layer which users can access directly (such as a web page, or an operating system's GUI).

Business Layer

Business layer or domain logic is the part of the program that encodes the real-world business rules which determine how data can be created, stored, and changed. It is contrasted with the remainder of the software that might be concerned with lower-level details of managing a database or displaying the user interface, system infrastructure, or generally connecting various parts of the program.

Data access layer

A Data Access Layer (DAL) in computer software, is a layer of computer program which provides simplified access to data stored in persistent storage.

For example, the DAL might return a reference to an object (in terms of object-oriented programming) with its attributes instead of a row of fields from a database table. This allows the client (or user) modules to be created with a higher level of abstraction. This kind of model could be implemented by creating a class of data access methods that directly reference a corresponding set of database stored procedures. Another implementation could potentially retrieve or write records to or from a file system. The DAL hides the complexity of the underlying data store from the external world.

Control layer

The control layer is responsible for the communication between business and presentation layer. It connects logic and data with each other and provides a better connectivity and separation between layers.

4.2 Pseudo Code for Major Functionalities

Login page: It is used for login purposes. When we enter the correct email and password it will go to the next page. We can use signup to create an account.

```
<?php
session_start();
$db=mysqli_connect("localhost","root","","courier");
if ($db->connect_error) {
    die("Connection failed: " . $db->connect_error);
}
if(isset($_POST['login_btn'])){
    if(!empty($_POST["cid"] && $_POST["cpwd"])){
    $username=$_POST['cid'];
    $password=$_POST['cpwd'];
        $sql="SELECT * FROM customer WHERE cid='$username' AND
cpwd='$password'";
        $result=mysqli_query($db,$sql);
            if(mysqli_num_rows($result)==1){
                $_SESSION["cid"]=$username;
                header('location:cust2.php');

            }

            else{
            echo "<script type='text/javascript'>alert('you are not logged
in');</script>";
            header('location:register.html');
            }
    }
    else{
        echo "<script type='text/javascript'>alert('please fill in the
credential or register');</script>";
        header('location:register.html');
    }

}
?>
```

Home page:

```
</head>
<body>
<div class="jumbotron">
  <div class="container text-center">
    <h1 id="home"><img src="logo.jpg" width="200" height="80">DELIVERY
CORNER</h1>
  </div>
</div>

<br>
<nav class="navbar navbar-inverse">
  <div class="container-fluid">

    <div class="collapse navbar-collapse" id="myNavbar">
      <ul class="nav navbar-nav">
        <li class="active"><a href="home.html">Home</a></li>
        <li><a href="admin.html">Admin</a></li>
        <li><a href="emp.php">Employee</a></li>
        <li><a href="cust.php">Customer</a></li>
        <li><a href="hub.html">Hubs</a></li>
        <li><a href="cont.html">Contact</a></li>
        <li><a href="review.html">Review</a></li>
        <li><a href="track.php">Track</a></li>
      </ul>
    </div>
  </div>
</nav>

<div class="container">
  <div id="myCarousel" class="carousel slide" data-ride="carousel">
    <!-- Indicators -->
    <ol class="carousel-indicators">
      <li data-target="#myCarousel" data-slide-to="0" class="active"></li>
      <li data-target="#myCarousel" data-slide-to="1"></li>
      <li data-target="#myCarousel" data-slide-to="2"></li>
    </ol>

    <!-- Wrapper for slides -->
    <div class="carousel-inner">
      <div class="item active">
        <img src="co1.jpg" alt="no1" style="width:100%; height: 300px;">
        <div class="bottom-left"style="font-size: 15px"><b>DELIVER ON
TIME</b></div>
      </div>
```

Continued……

codet.php: This is for viewing booked courier details

```php
<?php
include 'db.php';

//for total count data
$countSql = "SELECT COUNT(consid) FROM booking";
$tot_result = mysqli_query($conn, $countSql);
$row = mysqli_fetch_row($tot_result);
$total_records = $row[0];
$total_pages = ceil($total_records / $limit);

//for first time load data
if (isset($_GET["page"])) { $page  = $_GET["page"]; } else { $page=1; };
$start_from = ($page-1) * $limit;
$sql = "SELECT * FROM booking ORDER BY custid ASC LIMIT $start_from, $limit";

$rs_result = mysqli_query($conn, $sql);
?>
```

CHAPTER 5

RESULTS, SNAPSHOTS AND DISCUSSIONS

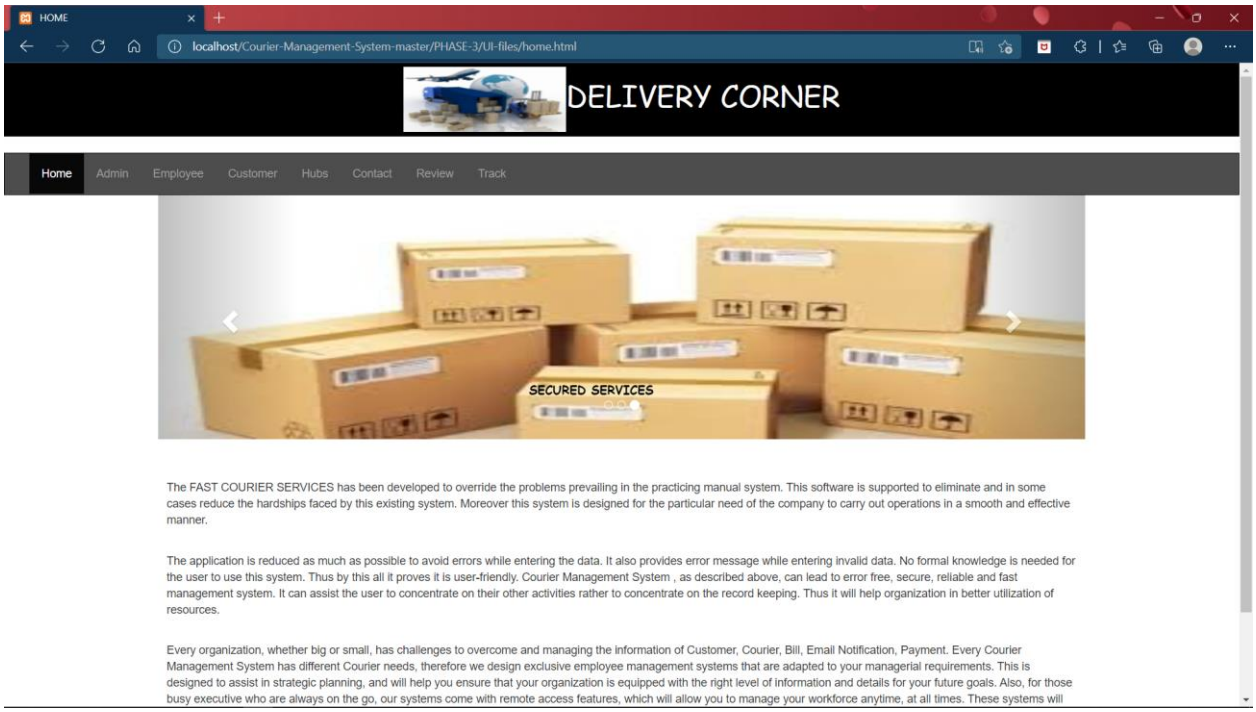


Fig.5.1 – Home Page

This is the homepage of the courier management system. It contains various sections-admin, employee, customer, hubs, contact, review and track. From this page user can navigate to other sections of the website.

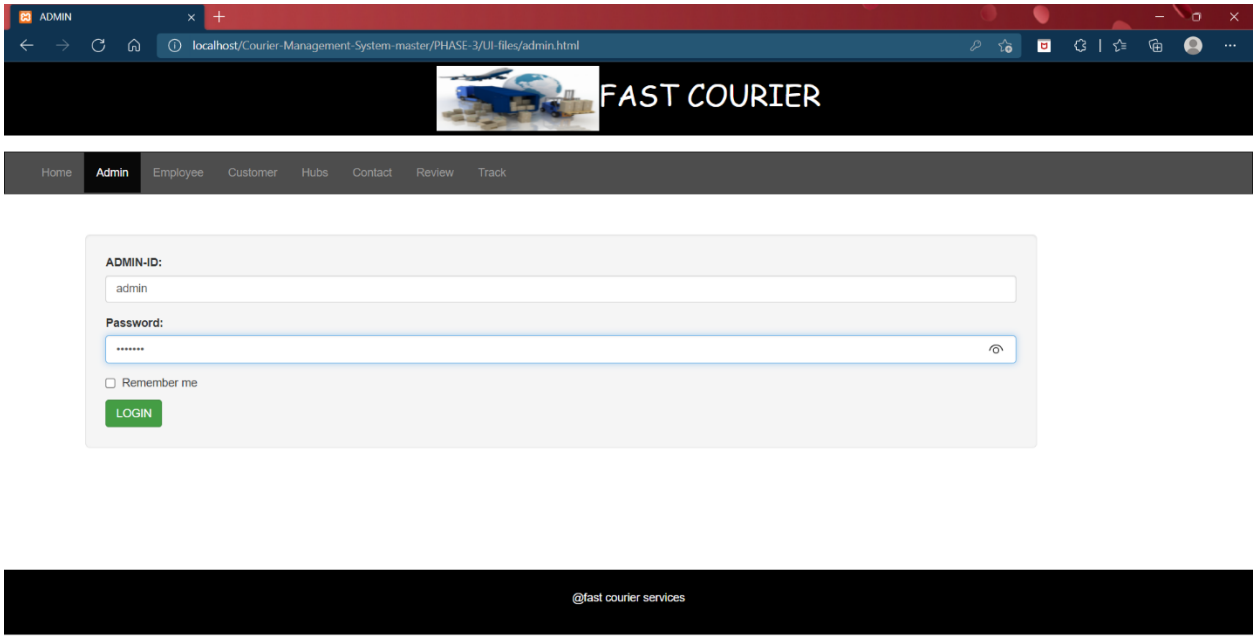


Fig.5.2 – Admin Login

This is admin login section. Admin can login using his id and password.

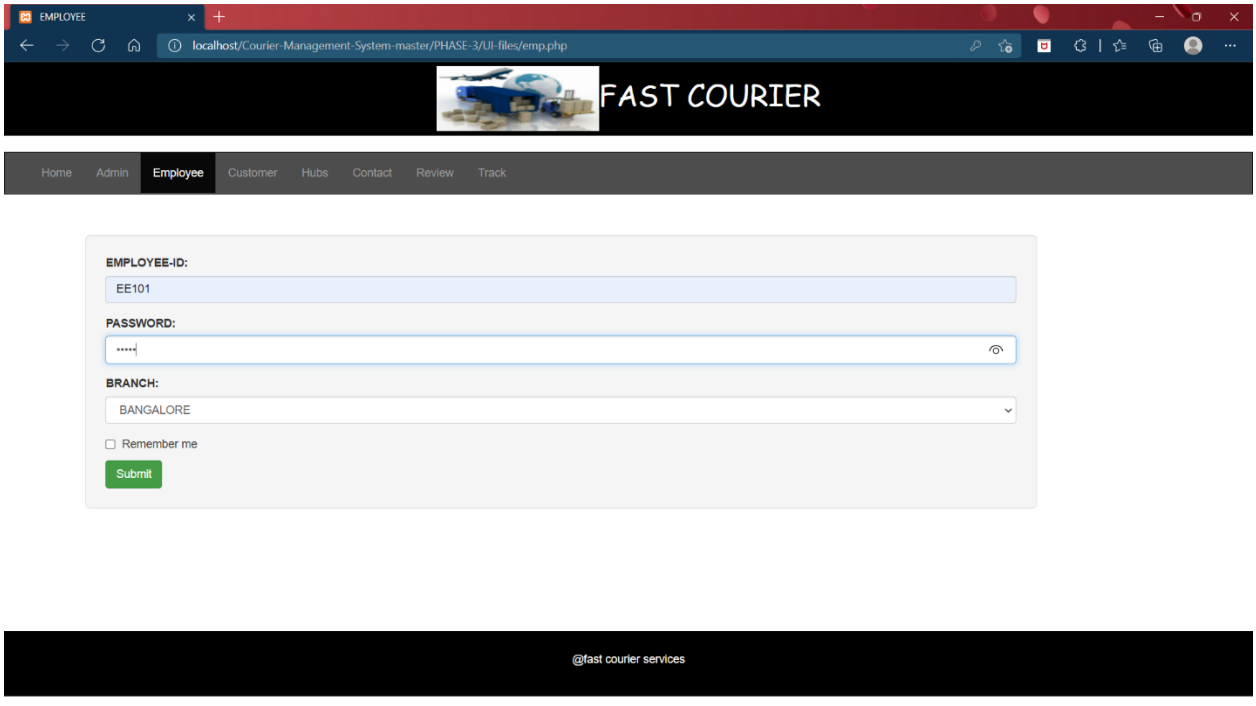


Fig.5.3 – Employee Login

Employees of the courier company can login by entering their id and password and their city.

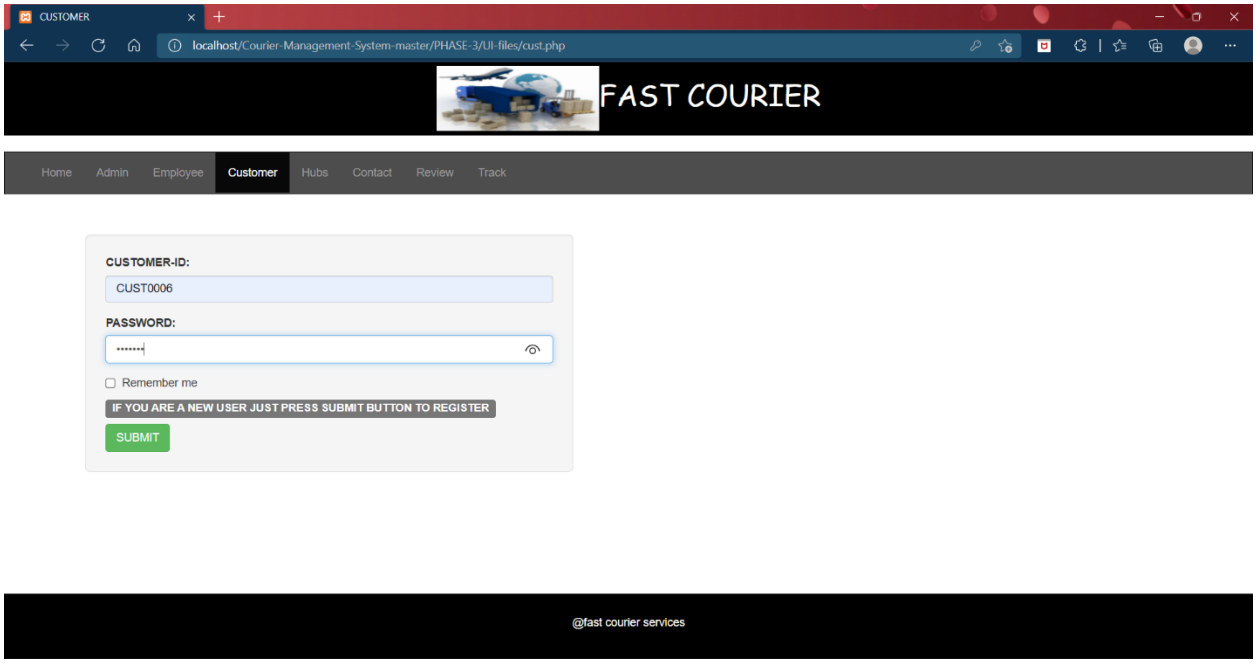


Fig.5.4 –Customer Login

An old customer can login to his account by entering his id and password. A new customer can register himself and then login.

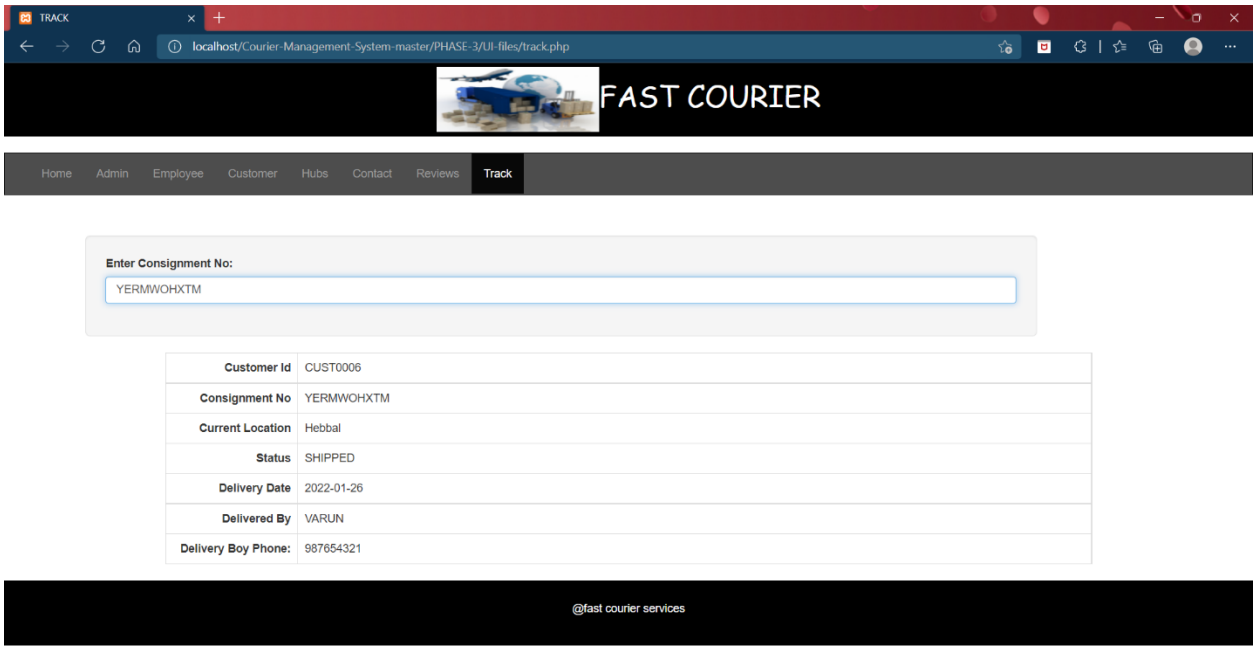


Fig.5.5 – Track

A user can track a particular courier using the consignment number.

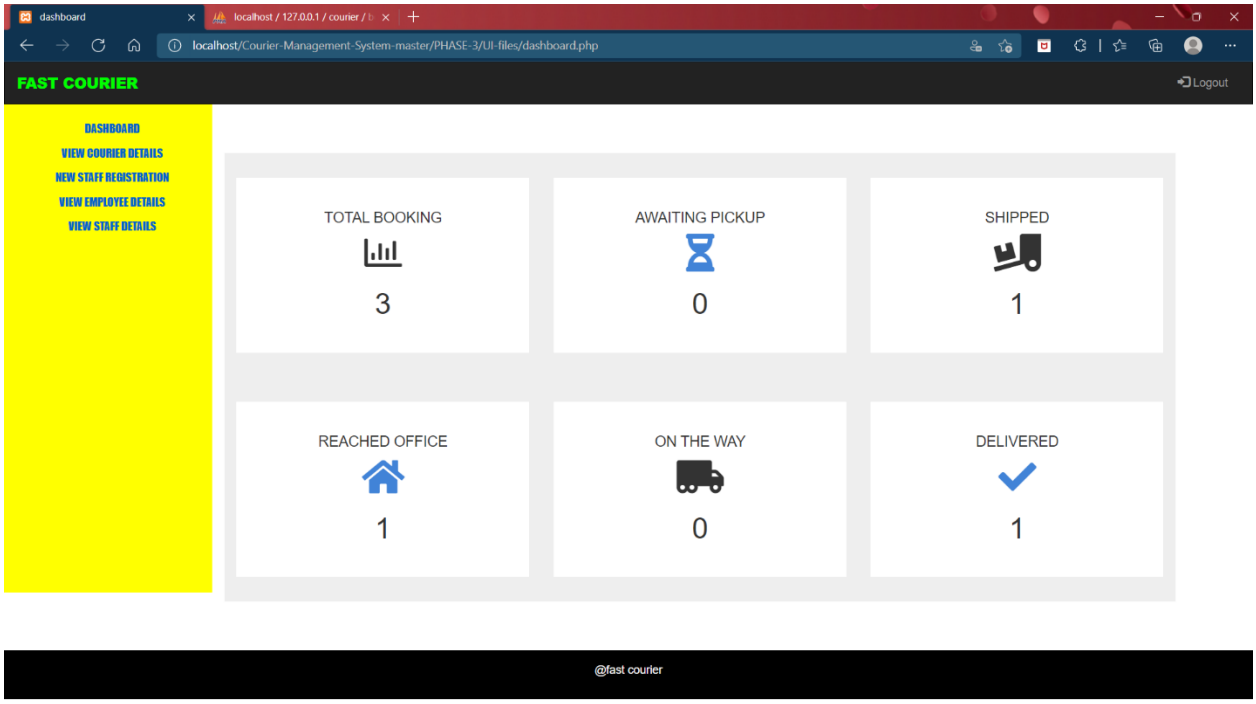


Fig.5.6 – Admin Dashboard

In the admin section, admin can overlook the total no.of bookings and their status among other things like view courier details, new staff registration, view employee details and view staff details.

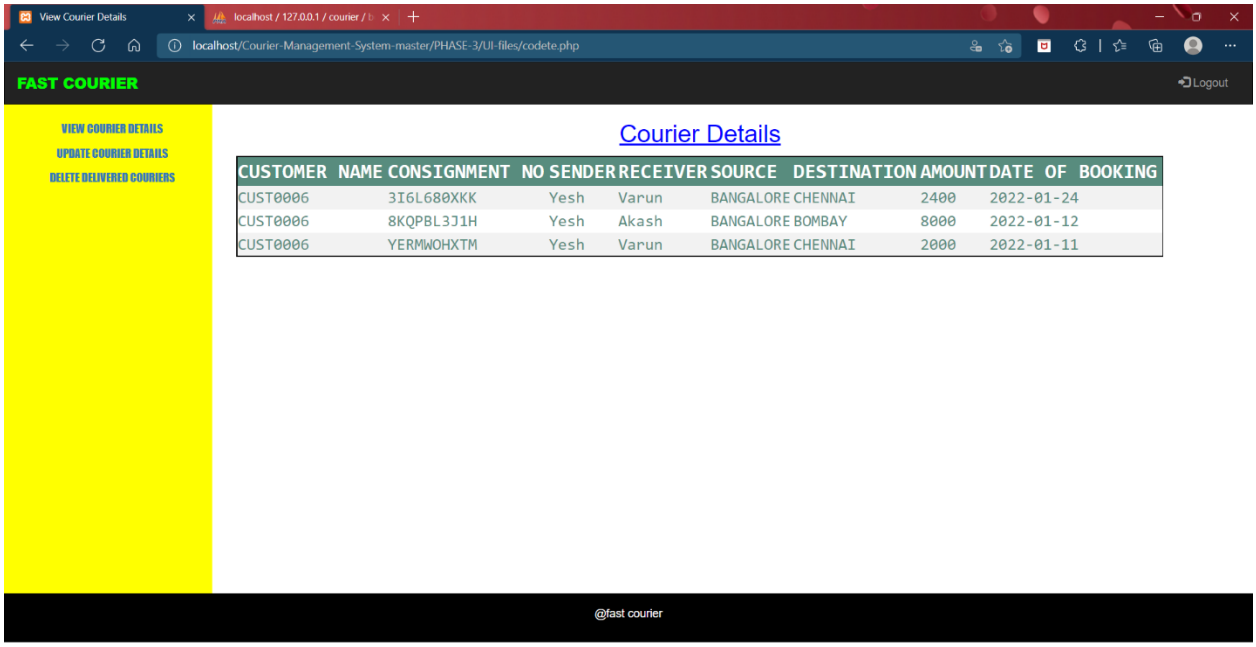
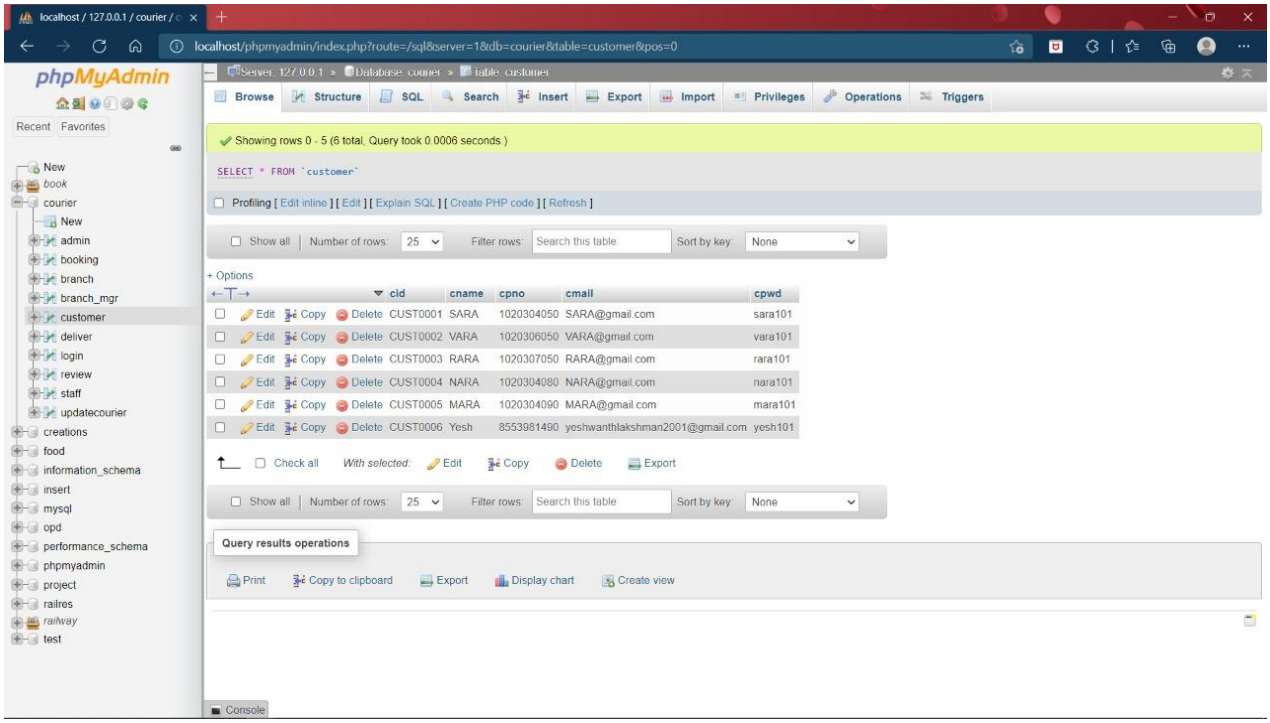


Fig.5.7 – Courier Details

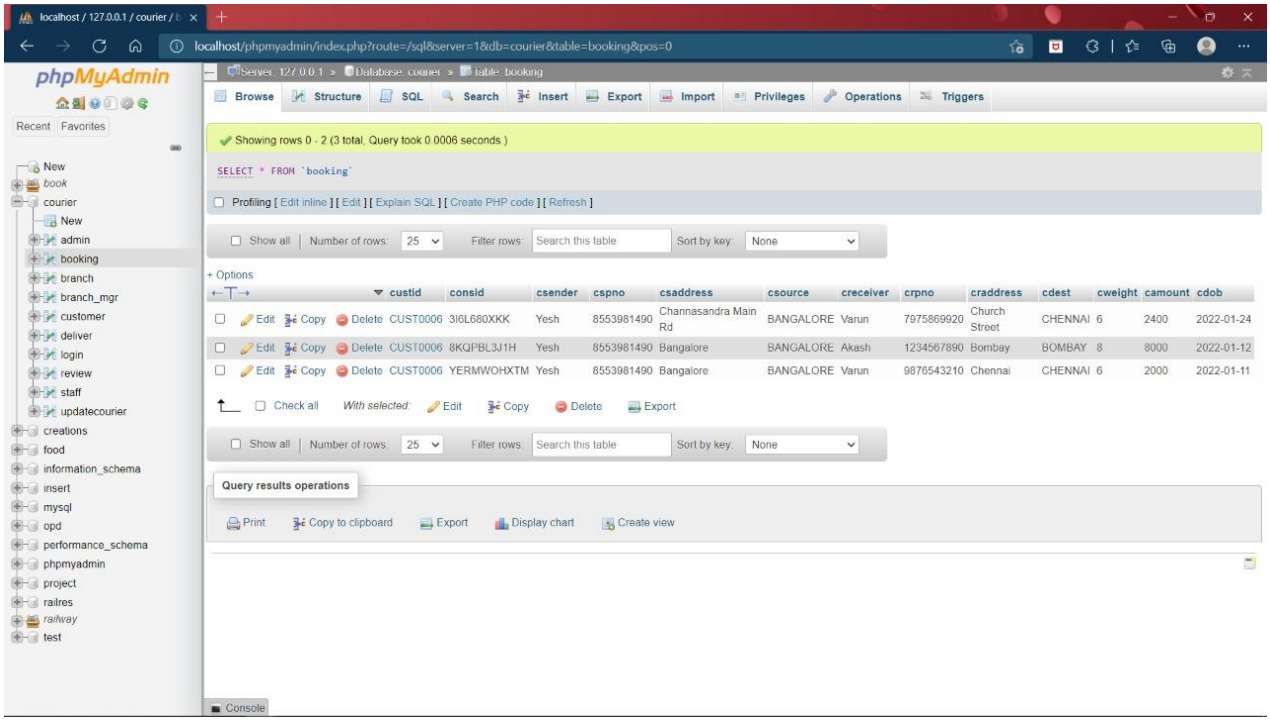
In the employee section, the employee can view the courier details and manage the couriers by updating their status.

Table entries in database:

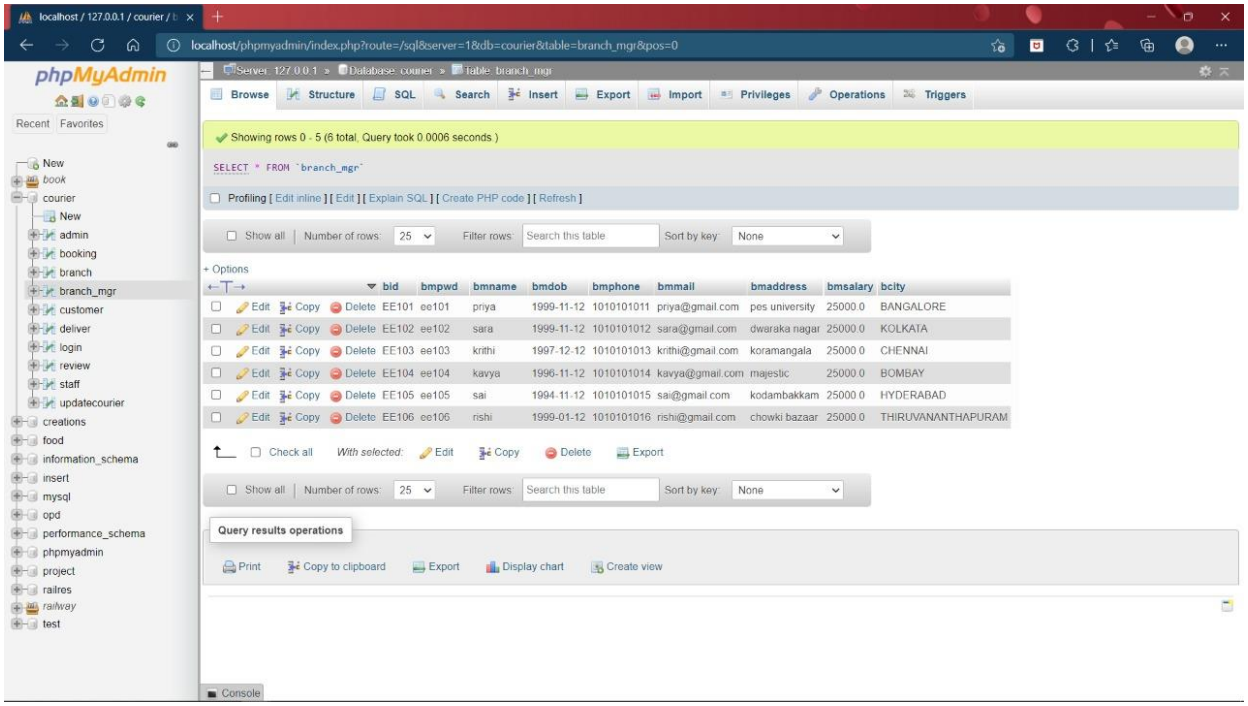
Customer Database:



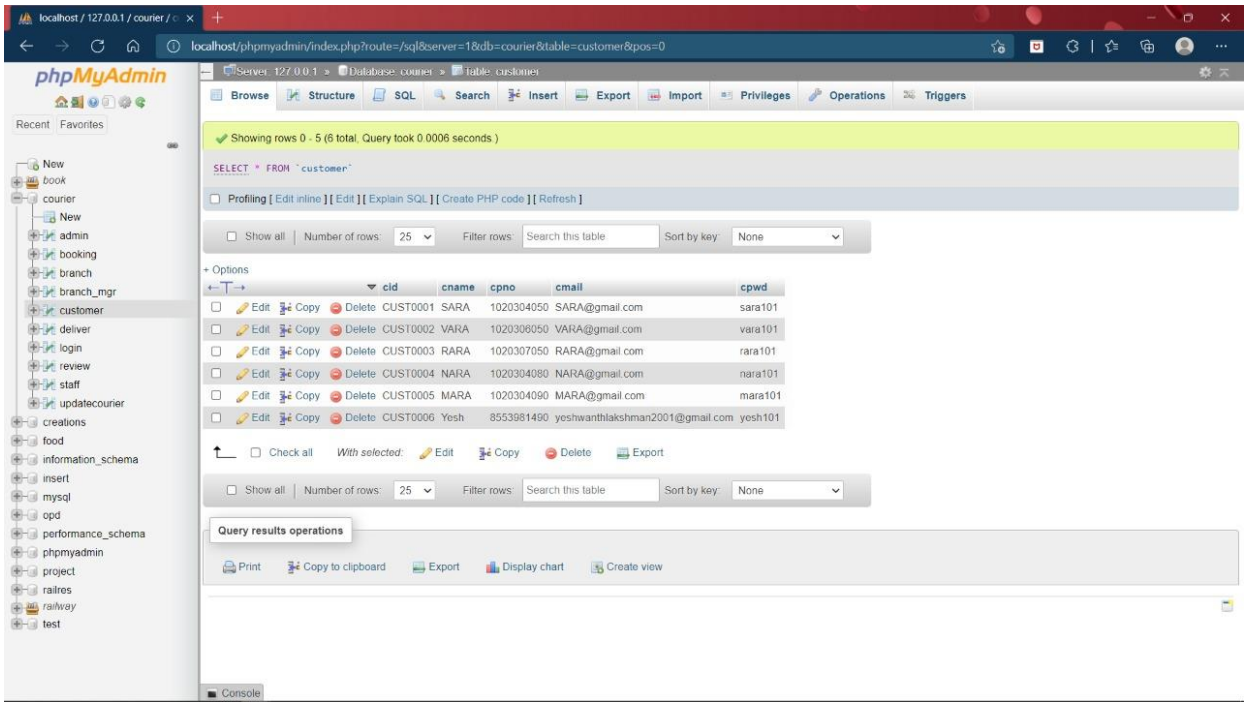
Booking Database:



Branch Manager Database:



Customer Database:



CHAPTER 6

CONCLUSION AND FUTURE ENHANCEMENTS

6.1 Conclusion

Courier management system is used for daily activities like booking a courier, maintain employee details, process payroll of employees, maintain hub details, maintain company details etc.

Keeping that in mind, we have built a platform that can be used by both the customer and the Employee. The customer can book a courier from his place to receiver's address and track the courier. Employee can view the courier details of the customer which consists of source, destination and consignment number and also looks after the customer's booking and can update the status accordingly.

We have made our website as user friendly as possible

Our back-end tech is reliable and efficient enough to not create any hurdles from the time the user logs into the website to the time when he/she has found the information needed. We hope that the users have a seamless experience when they visit our site.

6.2 Future Enhancements

User input should be cleaned before accepting it to minimize changes of SQL injections. There should be a log of transactions /backup to deal with unpredicted data loss. Security can be made stronger whenever necessary. Techniques like multiple server architecture can be implemented to cope with heavier traffic. Using frame works like django authentication will become easier. Also react/view web frameworks can be used to make to improve look and feel. Data validation also can be improved.

BIBILIOGRAPHY

[1] Ramez Elmarsi and Shamkant B. Navathe, Fundamentals of Database Systems, Pearson, 7th Edition.

[2] https://stackoverflow.com/

[3] https://www.w3schools.com/

[4] www.php.net

[5] https://trainings.internshala.com