# **Courier Management System**

Submitted in partial fulfilment of the requirements

For the award of the degree

of

### BACHELOR OF COMPUTER APPLICATIONS

Of

Bengaluru City University

Ву

Mohammed-Junaid

U18GO21S0088

Under the Guidance of

Smt. V Sudha



**July 2024** 

DEPARTMENT OF COMPUTER APPLICATIONS

SSMRV COLLEGE

BENGALURU - 560041

#### R.V. Educational Institutions

### Rashtreeya Sikshana Samithi Trust

S.S.M.R.V. College

4thT Block Jayanagar, Bengaluru 560041.



### **Department of Computer Applications**

#### CERTIFICATE

This is to certify that this project entitled "COURIER MANAGEMENT SYSTEM" is a Bonafide work done by Mohammed Junaid Reg no: U18GO21S0088 submitted to Bengaluru City University, Bengaluru for the award of Bachelor of Computer Applications during the academic year 2023-2024.

**Internal Project Guide** 

Head of the Dept.

Head of the Department

Submitted You the Viva-Voce Examination held on 8

SSMRV BANGALORE - 560 041

2

#### **ACKNOWLEGEMENT**

We consider it is our privilege to express through the paper of this report to all those who guided and inspired us in the completion of this project.

We would like to express our sincere gratitude to Dr. Geetha R, Principal who gave us permission and support to go ahead with this project.

We are grateful to our Head of the Department of Computer Applications, Smt. R. Shanthi Krishna and our Internal Guide, Smt. V Sudha for her valuable suggestions and guidance throughout the project for its successful completion and for remaining as the source of inspiration by giving her full support for completion of the project.

Last but not the least We thank our friends, well-wishers and parents for giving us their constant support and encouragement, without whom this project has not been a reality.

Mohammed Junaid [U18GO21S0088]

# **ABSTRACT**

The "Courier Management System" project aims to revolutionize the courier service industry byautomating and streamlining the processes involved in managing courier operations. In an erawhere efficiency and reliability are paramount, traditional manual systems often fall short, leading to errors, delays, and customer dissatisfaction. This project addresses these challenges by developing an integrated web-based platform designed to enhance the accuracy, efficiency, and overall user experience in courier services.

The system is built using a robust client-server architecture, leveraging technologies such as HTML, CSS, JavaScript for the client-side, PHP for the server-side, and MySQL for databasemanagement. Apache serves as the web server, ensuring reliable and secure communication between the client and server. The core functionalities of the system include user registration, parcel booking, real-time tracking, and comprehensive administrative control, all accessible through a user-friendly interface.

Key features of the Courier Management System include:

- User Module: Facilitates easy registration, login, and management of user profiles. Userscan book parcels, view their booking history, and track the status of their parcels in real-time.
- Parcel Module: Automates the booking and tracking process. Users can input parcel details, choose delivery options, and receive updates on their parcel's status. This moduleensures that all necessary information is captured accurately, minimizing errors.

- Admin Module: Provides administrative capabilities for managing users and parcels. Administrators can oversee all operations, manage user accounts, and handle exceptions. This module ensures that the system runs smoothly and efficiently.
- **Reporting Module:** Utilizes Power BI to generate insightful reports on courier operations. This module provides administrators with valuable analytics, helping themmake informed decisions and improve service quality.

# **INDEX**

ACKNOWLEGMENT	3	
ABSTRACT	4	
Objective	8	
Existing System	8	
Proposed System	8	
Study of the System	9	
<ul><li>TECHNICAL FEA</li><li>OPERATIONAL</li></ul>	ASIBILITY FEASIBILI	
<ul> <li>Functional Requirement</li> <li>User Module</li> <li>Admin Module</li> <li>Parcel Module</li> <li>Reporting Mod</li> </ul>		11 12 12 13 14
<ul><li> User Module</li><li> Admin Module</li><li> Parcel Module</li></ul>		12 12 13
<ul><li>User Module</li><li>Admin Module</li><li>Parcel Module</li><li>Reporting Mod</li></ul>	ule	12 12 13
<ul> <li>User Module</li> <li>Admin Module</li> <li>Parcel Module</li> <li>Reporting Mod</li> </ul> System Requirement	ule 15	12 12 13
<ul> <li>User Module</li> <li>Admin Module</li> <li>Parcel Module</li> <li>Reporting Mod</li> </ul> System Requirement Hardware Configuration	ule 15 15	12 12 13
	ABSTRACT Objective Existing System Proposed System Study of the System  • FEASIBILITY ST • TECHNICAL FEA • OPERATIONAL	ABSTRACT 4 Objective 8 Existing System 8 Proposed System 8

Features

Usage

• Sp	eed Optimization.	18	
	curity	18	
• Sy	ntax	19	
> MYSQL	ı	20	
• M	Y SQL Functions	21	
> PHP MY	YADMIN	22	
• M	Y ADMIN Functions.	22	
> APACH	E WEB Server	23	
• Co	onfiguration Files	24	
> INPUT	DESIGNS	25	
> OUTPU	T DESIGNS	27	
➤ ER Diag	grams	29	
• DA • UN	TRAL Model ATA FLOW DIAGRA ML DIAGRAMS NTITY RELATIONS		29 30 34 36
> Architec	cture Flow		39
> CODE			42
> DATAB	ASE DESIGN		57
Output S	Screen of the Project (	Screenshots).	60
> Conclus	ion		67
Bibliogr	aphy		68

# **OBJECTIVE:**

	<b>Develop a User-Friendly Interface</b> : Create an intuitive and easy-to-navigate web-based
	interface for customers and administrators to manage courier-related tasks efficiently.
	Automate Courier Operations: Implement an automated system to handle the complete
	lifecycle of a courier service, from parcel booking and tracking to delivery and reporting.
	Enhance Tracking Capabilities: Provide real-time tracking of parcels to ensure
	transparency and improve customer satisfaction.
	<b>EXISTING SYSTEM:</b>
	Manual Data Entry and Processing: Many traditional courier services rely on manual
_	data entry for recording parcel information, which leads to errors, inconsistencies, and
	data loss. This process is time-consuming and requires significant human resources.
	Lack of Real-Time Tracking: In conventional systems, customers often do not have
	access to real-time tracking information for their parcels. This lack of transparency leads
	to customer dissatisfaction and increased inquiries to customer support.
	Inefficient Resource Management: Manual systems struggle with efficient allocation
	and management of resources, such as delivery personnel and vehicles. This can result in
	delayed deliveries and higher operational costs.
	PROPOSED SYSTEM:
	Automated Data Entry and Processing: Automate the recording of parcel information
	through user-friendly forms and database integration. Use validation checks to ensure
	data accuracy.
	Real-Time Parcel Tracking: Integrate GPS tracking and status updates, allowing
	customers to monitor their parcels in real-time through a web-based interface.
	Efficient Resource Management: Develop algorithms for dynamic scheduling and routing to
	ensure timely deliveries and efficient resource utilization

### STUDY OF THE SYSTEM:

To provide flexibility to the users, the interfaces have been developed that are accessible through abrowser. The GUI'S at the top level have been categorized as

- 1. Administrative user interface
- 2. The Operational or Generic User Interface
- The 'Administrative user interface' concentrates on the consistent information that is practically, part of the organizational activities and which needs proper authentication forthe data collection. These interfaces help the administrators with all the transactional states like Data insertion, Data deletion and Date updation along with the extensive data search capabilities.
- The 'Operational or Generic User Interface' helps the end users of the system in transactions throughthe existing data and required services. The operational user interfacealso helps the ordinary users in managing their own information in a customized manner as per the included flexibilities.

### **FEASIBILITY STUDY:**

### **FEASIBILITY REPORT:**

Preliminary investigation examines project feasibility; the likelihood the system will be useful to the organization. The main objective of the feasibility study is to test the Technical, Operational and Economical feasibility for adding new modules and debugging old running system. All systems are feasible if they are given unlimited resources and infinite time. There are aspects in the feasibility study portion of the preliminary investigation:

- Technical Feasibility
- Operation Feasibility
- Economical Feasibility.

### **TECHNICAL FEASIBILITY:**

The technical issue usually raised during the feasibility stage of the investigation includes the following:

- Does the necessary technology exist to do what is suggested?
- Do the proposed equipments have the technical capacity to hold the data required to use thenew system?
- Will the proposed system provide adequate response to inquiries, regardless of the number or location of users?
- Can the system be upgraded if developed?
- Are there technical guarantees of accuracy, reliability, ease of access and data security?

### **OPERATIONAL FEASIBILITY:**

Proposed projects are beneficial only if they can be turned out into information systems, whichwill meet the organization's operating requirements. Operational feasibility aspects of the project are to be taken as an important part of the project implementation. Some of the important issues raised are to test the operational feasibility of a project includes the following: -

- Is there sufficient support for the management from the users?
- Will the system be used and work properly if it is being developed and implemented?
- Will there be any resistance from the user that will undermine the possible application benefits? This system is targeted to be in accordance with the above-mentioned issues.

Beforehand, the management issues and user requirements have been taken into consideration. So there is no question of resistance from the users that can undermine the possible application benefits.

The well-planned design would ensure the optimal utilization of the computer resources and would help in the improvement of performance status.

### **ECONOMIC FEASIBILITY:**

A system can be developed technically and that will be used if installed must still be a good investment for the organization. In the economical feasibility, the development cost in creating the system is evaluated against the ultimate benefit derived from the new systems. Financial benefits must equal or exceed the costs. The system is economically feasible. It does not require any additional hardware or software.

# **FUNCTIONAL REQUIREMENTS:**

The functional requirements outline the specific behaviors, features, and functions that the proposed Courier Management System must provide. These requirements ensure the system meets the needs of all users, including customers, administrators, and delivery personnel. The following are the primary functional requirements for the system:

User Module
Admin Module
Parcel Module
Reporting Module

### □ USER MODULE:

#### • User Registration and Authentication

- Users must be able to register by providing their personal details.
- The system should provide secure login functionality with username and password.
- Password reset functionality must be available for users who forget their passwords.

#### • User Profile Management

- Users should be able to view and update their profile information.
- The system must validate and store updated profile data securely.

### Parcel Booking

- Users must be able to book parcels by entering details such as sender and recipient information, parcel description, and delivery preferences.
- The system should generate a unique tracking number for each booked parcel.

#### • Real-Time Parcel Tracking

- Users should be able to track their parcels in real-time using the unique tracking number.
- The system must provide status updates, including the current location and estimateddelivery time.

#### □ ADMIN MODULE:

#### • User Management

- Administrators must be able to view, edit, and delete user accounts.
- The system should provide a search functionality to locate specific users.

#### • Parcel Management

- Administrators should be able to view all parcels, update their status, and managedelivery schedules.
- The system must allow admins to assign parcels to delivery personnel and update parcelstatuses.

#### • Resource Management

- The system should help in scheduling and routing deliveries efficiently.
- Administrators must be able to allocate resources such as delivery personnel and vehicles.

#### □ PARCEL MODULE:

### • Parcel Information Management

- The system should store detailed information about each parcel, including sender, recipient, parcel type, and delivery instructions.
- It must provide an interface for viewing and updating parcel information.

### • Status Updates

- The system must allow administrators and delivery personnel to update the status ofparcels at various stages (e.g., picked up, in transit, delivered).
- Users should receive notifications about status changes via email or SMS.

#### • Delivery Confirmation

- The system should allow delivery personnel to confirm the delivery of parcels, including capturing recipient signatures.
- Confirmation details must be stored and accessible for future reference.

### **□** REPORTING MODULE:

### • Performance Analysis

- The system should provide tools for analyzing delivery performance, including on-timedelivery rates and average delivery times.
- It must allow filtering and sorting of performance data based on various criteria.

### • Customer Satisfaction

- The system should include features for collecting customer feedback and satisfaction ratings.
- Reports on customer satisfaction should be generated for management review.

### • Operational Insights

- The system must provide insights into operational efficiency, resource utilization, andbottlenecks.
- Administrators should be able to generate custom reports based on specific operationalmetrics.

# **SYSTEM ENVIRONMENT**

# > Hardware Configuration

- 1. Pentium IV Processor
- **2.** 512 MB RAM
- 3.40GB HDD
- **4.** 1024 \* 768 Resolution Color Monitor

# > Software Configuration

- 1. OS: Windows XP
- 2. PHP Triad (PHP5.6, MySQL, Apache, and PHPMyAdmin)

# > Software Features / Database

### PHP TRIAD:

PHPTriad installs a complete working PHP/MySQL server environment on Windows platforms (9x/NT). Installs PHP, MySQL, Apache, and PHPMyAdmin.

### > PHP

PHP (Hypertext Preprocessor) is a widely-used open-source server-side scripting language designed primarily for web development. It was originally created by Danish-Canadian programmer RasmusLerdorf in 1994 and has since been continuously developed by a large community of developers.

PHP is particularly well-suited for creating dynamic web pages and web applications. It can generate dynamic content, interact with databases, handle forms, manage sessions, create cookies, and perform many other tasks necessary for web development. PHP code is usually embedded directly into HTML documents, allowing developers to mix PHP.

### > Some key features of PHP include:

- **1. Open Source:** PHP is free to use and has a large community of developers contributing to its development and support.
- 2. Cross-Platform: PHP runs on various platforms, including Windows, Linux, macOS and many others.
- **3. Server-Side Scripting:** PHP scripts are executed on the server, generating HTML which is thensent to the client's browser. This allows for dynamic content generation.
  - **4. Database Integration:** PHP has built-in support for working with databases like MySQL,PostgreSQL, SQLite, and others, making it easy to create database-driven web applications.
- **5. Extensive Library Support:** PHP has a vast ecosystem of libraries and frameworks that provide pre-built functions and modules for common tasks, speeding up development.

**6. Simple Syntax:** PHP syntax is relatively easy to learn and understand, especially for those with abackground in C-style languages like C, C++, and Java.

Overall, PHP remains one of the most popular choices for web development due to its versatility, ease of use, and extensive community support.

## > Usage

PHP is a general-purpose scripting language that is especially suited for web development. PHP generally runs on a web server, taking PHP code as its input and creating web pages as output. It can be used for command-line scripting and client-side GUI applications. PHP can be deployed on most web servers, many operating systems and platforms, and can be used with many relational database management systems. It is available free of charge, and the PHP Group provides the completesource code for users to build, customize and extend for their own use.

PHP primarily acts as a filter, taking input from a file or stream containing text and/or PHP instructions

and outputs another stream of data; most commonly the output will be HTML. It can automatically detect the language of the user. From PHP 4, the PHP parser compiles input to produce

bytecode for processing by the Zend Engine, giving improved performance over its interpreter predecessor. Originally designed to create dynamic web pages, PHP's principal focus is server-side scripting, and it is similar to other server-side scripting.

Languages that provide dynamic content from a web server to a client, such as Microsoft's ActiveServer Pages, Sun Microsystems' JavaServer Pages, and mod\_perl. PHP has also attracted the development of many frameworks that provide building blocks and a design structure to promoterapid application develo-pment (RAD). Some of these include CakePHP, Symfony, CodeIgniter, and Zend Framework, offering features similar to other web application frameworks.

### **Speed optimization**

As with many scripting languages, PHP scripts are normally kept as human-readable source code, even on production web servers. In this case, PHP scripts will be compiled at runtime by the PHP engine, which increases their execution time. PHP scripts are able to be compiled before runtime using PHP compilers as with other programming languages such as C (the language PHP and its extensions are written in). Code optimizers aim to reduce the computational complexity of the compiled code by reducing its size and making other changes that can reduce the execution time with the overall goal ofimproving performance. The nature of the PHP compiler is such that there are often opportunities for code optimization, and an example of a code optimizer is the Zend Optimizer PHP extension.

Another approach for reducing overhead for high load PHP servers is using PHP accelerators. Thesecan offer significant performance gains by caching the compiled form of a PHP script in shared memory to avoid the overhead of parsing and compiling the code every time the script runs.

### **Security**

The National Vulnerability Database stores all vulnerabities found in computer software. The overallproportion of PHP-related vulnerabilities on the database amounted to: 12% in 2003, 20% in 2004, 28% in 2005, 43% in 2006, 36% in 2007, and 35% in 2008. Most of these PHP-related vulnerabilities can be exploited remotely: they allow hackers to steal or destroy data from data sources linked to thewebserver (such as an SQL database), send spam or contribute to DOS attacks using malware, whichitself can be installed on the vulnerable servers.

These vulnerabilities are caused mostly by not following best practice programming rules: technical security flaws of the language itself or of its core libraries are not frequent. Recognizing that programmers cannot be trusted, some languages include taint checking.

Hosting PHP applications on a server requires a careful and constant attention to deal with these security risks. There are advanced protection patches such as Suhosin and Hardening-Patch, especially designed for web hosting environments. Installing PHP as a CGI binary rather than as an Apache module is the preferred method for added security. With respect to securing the code itself, PHP code can be obfuscated to make it difficult to read while remaining functional.

### > Syntax

Note: - Code in bold letters shows the PHP code embedded within HTML

PHP only parses code within its delimiters. Anything outside its delimiters is sent directly to the output and is not parsed by PHP. The most common delimiters are <?php and ?>, which are open and close delimiters respectively. <script language="php"> and </script> delimiters are also available. Short tags can be used to start PHP code, <? or<?= (which is used to echo back a string or variable) and the tag toend PHP code, ?>. These tags are commonly used, but like ASP-style tags (<% or <%= and %>), they are less portable as they can be disabled in the PHP configuration. For this reason, the use of short tags and ASP-style tags is discouraged. The purpose of these delimiters is to separate PHP code from non- PHP code, including HTML.

Variables are prefixed with a dollar symbol and a type does not need to be specified in advance. Unlike function and class names, variable names are case sensitive. Both double-quoted ("") and heredoc strings allow the ability to embed a variable's value into the string. PHP treats newlines as whitespace in the manner of a free-form language (except when inside string quotes), and statements are terminated by a semicolon. PHP has three types of comment syntax: /\* \*/ serves as block comments, and // as well as # are used for inline comments. The echo statement is one of several facilities PHP provides to outputtext (e.g. to a web browser).

In terms of keywords and language syntax, PHP is similar to most high level languages that follow the C style syntax. *If* conditions, *for* and *while* loops, and function returns are similar in syntax to languages such as C, C++, Java and Perl.

# **MY SQL**

What is a database?

Quite simply, it's an organized collection of data. A database management system (DBMS) such as Access, FileMaker Pro,Oracle or SQL Server provides you with the software tools you need to organize that data in a flexible manner. It includes facilities to add, modify or delete data from the database, ask questions (or queries) about the data stored in the database and produce reports summarizing selected contents.

MySQL is a multithreaded, multi-user SQL database management system(DBMS). The basic program runs as a server providing multi-user access to a number of databases. Originally financed in a similar fashion to the JBoss model, MySQL was owned and sponsored by a single for-profit firm, the Swedish company MySQLAB now a subsidiary of Sun Microsystem, which holds the copyright to most of the codebase. The project's source code is available under terms of the GNU General Public Licence, as well as under a variety of proprietory agreements.

MySQL is a database. The data in MySQL is stored in database objects called tables. A table is a collections of related data entries and it consists of columns and rows. Databases are useful when storing information categorically. A company may have a database with the following tables: "Employees", "Products", "Customers" and "Orders".

# 2.3.1.2.1 MySQL Functions

```
mysql_affected_rows — Get number of affected rows in previous
   MySQL operation mysql_change_user — Change logged in user of the
   active connection mysql_client_encoding — Returns the name of the
   character set
  mysql_close — Close MySQL connection
  mysql_connect — Open a connection to a
  MySQL Server mysql_create_db — Create a
  MySQL databasemysql_data_seek — Move
  internal result pointer mysql_db_name — Get
  result data
     mysql_db_query — Send a MySQL
     query mysql_drop_db — Drop
     (delete) a MySQL
mysql_errno — Returns the numerical value of the error message from previous MySQL
operation mysql_error — Returns the text of the error message from previous
                            MySQL operation
    mysql_escape_string — Escapes a string for use in a mysql_query
   mysql fetch array — Fetch a result row as an associative array, a numeric
   array,or both mysql_fetch_assoc — Fetch a result row as an associative
                                    array
  mysql_fetch_field — Get column information from a result and return
  as anobject mysql_fetch_lengths — Get the length of each output in a
      result mysql_fetch_object — Fetch a result row as an object
      wsmysql_num_rows — Get number of rows in result
      mysql_pconnect — Open a persistent connection to a
                        MySQLserver
   mysql_ping — Ping a server connection or reconnect if there is no
          connection mysql_query — Send a MySQL query
      mysql_result — Get result data
```

## 2.3.1.4 phpMyAdmin

phpMyAdmin is an opensource tool written in PHP intended to handle the administration of MySQLover the World Wide Web. phpMyAdmin supports a wide range of operations with MySQL. Currently it can create and drop databases, create/drop/alter tables, delete/edit/add fields, execute any SQL statement, manage users and permissions, and manage keys on fields. while you still have the ability to directly execute any SQL statement. phpMyAdmin can manage a whole MySQL server (needs a super-user) as well as a single database. To accomplish the later you'll need a properly set up MySQL user who can read/write only the desired database. It's up to you to look up the appropriate part in the MySQL manual.

# > phpMyAdmin Features:

- · browse and drop databases, tables, views, fields and indexes
- · create, copy, drop, rename and alter databases, tables, fields and indexes
- · maintenance server, databases and tables, with proposals on server configuration
- execute, edit and bookmark any SQL-statement, even batch-queries
- · load text files into tables
- · create and read dumps of tables
- export data to various formats: CSV, XML, PDF, ISO/IEC 26300 OpenDocument
   Text andSpreadsheet, Word, Excel and L<sup>A</sup>T<sub>E</sub>X formats
- · administer multiple servers
- · manage MySQL users and privileges
- · check referential integrity in MyISAM tables
- · using Query-by-example (QBE), create complex queries automatically connecting required tables
- · create PDF graphics of your Database layout
- search globally in a database or a subset of it
- transform stored data into any format using a set of predefined functions, like displayingBLOB-data as image or download-link
- · support InnoDB tables and foreign keys
- · support mysqli, the improved MySQL extension

# 2.3.1.4 Apache Web server

Often referred to as simply *Apache*, a public-domain open source Web server developed by a loosely-knit group of programmers. The first version of Apache, based on the NCSA httpd Web server, was developed in 1995.

Core development of the Apache Web server is performed by a group of about 20 volunteer programmers, called the *Apache Group*. However, because the source code is freely available, anyone can adapt the server for specific needs, and there is a large public library of Apache add-ons. In many respects, development of Apache is similar to development of the Linux operating system.

The original version of Apache was written for UNIX, but there are now versions that run under OS/2, Windows and other platforms. The name is a tribute to the Native American Apache Indian tribe, a tribe well known for its endurance and skill in warfare. A common misunderstanding is that it was called Apache because it was developed from existing NCSA code plus various patches, hence the name *a patchy server*, or Apache server.

Apache consistently rates as the world's most popular Web server according to analyst surveys. Apache has attracted so much interest because it is full-featured, reliable, and free. Originally developed for UNIX<sup>TM</sup> operating systems, Apache has been updated to run on Windows, OS/2, and other platforms. One aspect of Apache that some site administrators find confusing — especially those unfamiliar with UNIX-style software — is its configuration scheme. Instead of using a point-and-click graphic user interface (GUI) or Windows Registry keys as most other

### **Configuration Files**

Apache uses a system of three text files for managing its configuration data. All three of these files (almost always) appear in Apache's ./conf directory and are designed to be edited by system administrators:

- 1. httpd.conf for general settings
- 2. srm.conf for resource settings
- 3. access.conf for security settings

When Apache first starts, these files are processed in the order shown above. Originally, the initial installation of Apache included default entries within each of the three files. In the most recent versions of Apache, however, the default installation has changed. Now httpd.conf is treated as the "master" configuration file and it contains all of the settings. Both srm.conf and access.conf still exist in the installation, but they contain no settings and are empty except for some comments.

#### **Access and Security Settings**

It is recommended practice now for Apache administrators to manage their resource and security settings from httpd.conf. Administrators of older versions of Apache can simply cut their entries from srm.conf and access.conf and paste them into the master file. If an administrator wants to go one step further and delete the two empty files, they should also place the following entries in httpd.conf to prevent Apache from attempting to access them.

# **INPUT DESIGN:**

Input design is a part of overall system design. The main objective during the input design is as given below:

- To produce a cost-effective method of input.
- To achieve the highest possible level of accuracy.
- To ensure that the input is acceptable and understood by the user.

# **INPUT STAGES:**

The main input stages can be listed as below:

- Data recording
- Data transcription
- Data conversion
- Data verification
- Data control
- Data transmission
- Data validation
- Data correction

### **INPUT TYPES:**

It is necessary to determine the various types of inputs. Inputs can be categorized as follows:

- External inputs, which are prime inputs for the system.
- Internal inputs, which are user communications with the system.
- Operational, which are computer department's communications to the system?
- Interactive, which are inputs entered during a dialogue.

### **INPUT MEDIA:**

At this stage choice has to be made about the input media. To conclude about the input media consideration has to be given to;

- Type of input
- Flexibility of format
- Speed
- Accuracy
- Verification methods
- Rejection rates
- Ease of correction
- Storage and handling requirements
- Security
- Easy to use
- Portability

Keeping in view the above description of the input types and input media, it can be said that most of the inputs are of the form of internal and interactive.

As Input data is to be the directly keyed in by the user, the keyboard can be considered to be the most suitable input device.

# **OUTPUT DESIGN:**

Outputs from computer systems are required primarily to communicate the results of processing to users. They are also used to provide a permanent copyof the results for later consultation. The various types of outputs in general are:

- External Outputs whose destination is outside the organization.
- Internal Outputs whose destination is with in organization and they are the User's main interface with the computer.
- Operational outputs whose use is purely with in the computer department.
- Interface outputs, which involve the user in communicating directly with the system.

# **OUTPUT DEFINITION**

The outputs should be defined in terms of the following points:

- Type of the output
- Content of the output
- Format of the output
- Location of the output
- Frequency of the output
- Volume of the output
- Sequence of the output

It is not always desirable to print or display data as it is held on a computer. It should be decided aswhich form of the output is the most suitable.

### For Example

- Will decimal points need to be inserted
- Should leading zeros be suppressed.

# **OUTPUT MEDIA:**

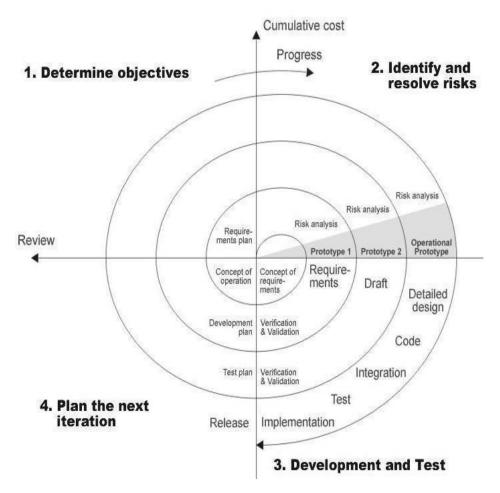
In the next stage it is to be decided that which medium is the most appropriate for the output. The main considerations when deciding about the output media are:

- The suitability for the device to the particular application.
- The need for a hard copy.
- The response time required.

### **ER DIAGRAMS**;

### **SPIRAL MODEL:**

The following diagram shows how a spiral model acts like:

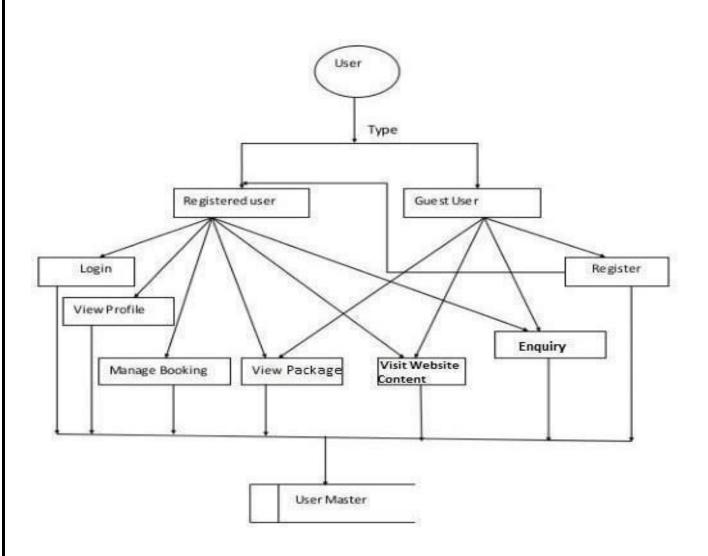


# **Performance Requirements:**

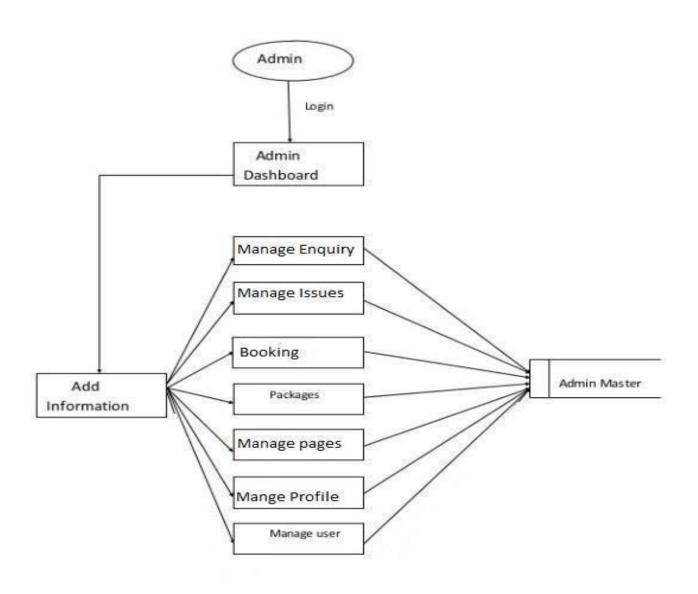
Performance is measured in terms of the output provided by the application. Requirement specification plays an important part in the analysis of a system. Only when the requirement specifications are properly given, it is possible to design a system, which will fit into required environment. It rests largely with the users of the existing system to give the requirement specifications because they are the people who finally use the system.

# **DFD (DATA FLOW DIAGRAM):**

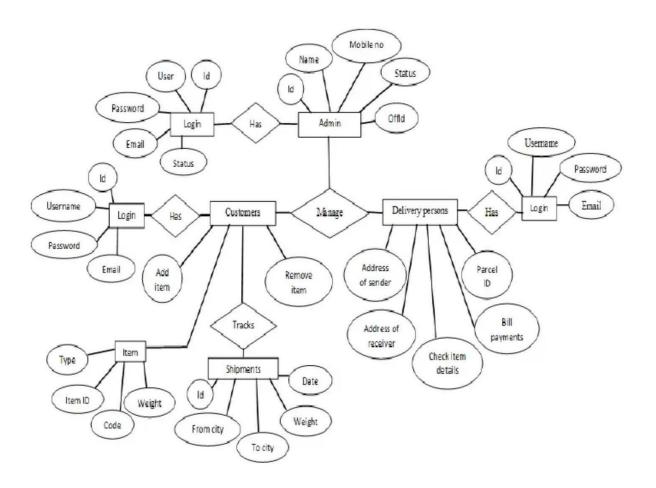
a) <u>Users:</u>

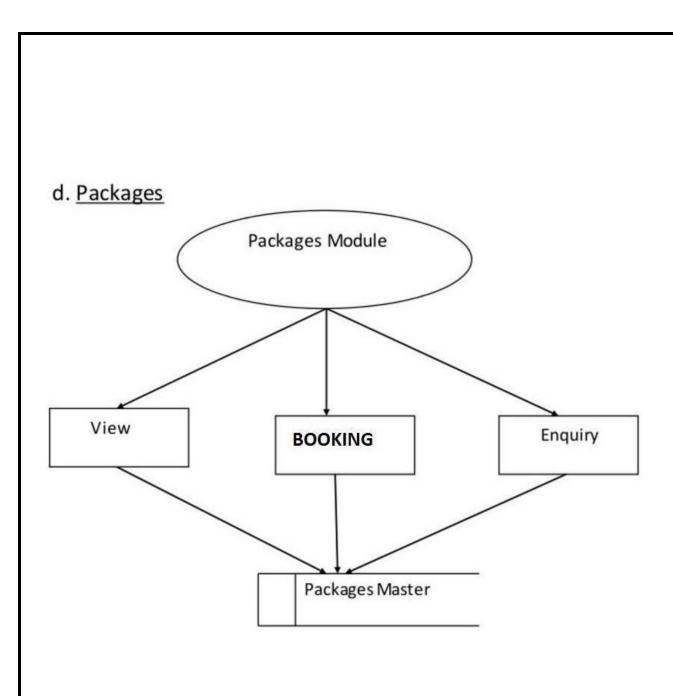


# b) Admin:

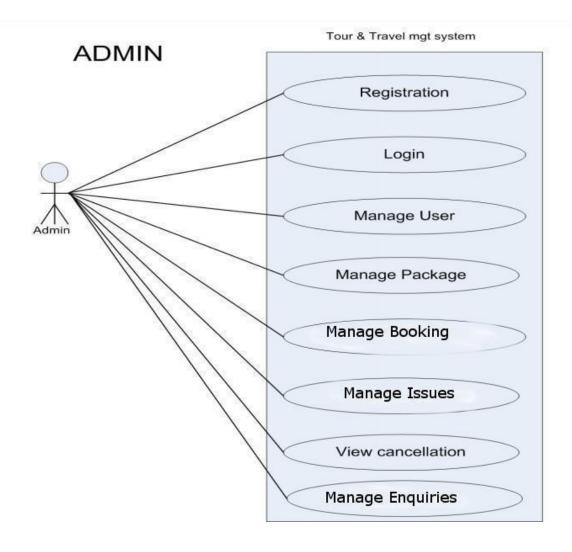


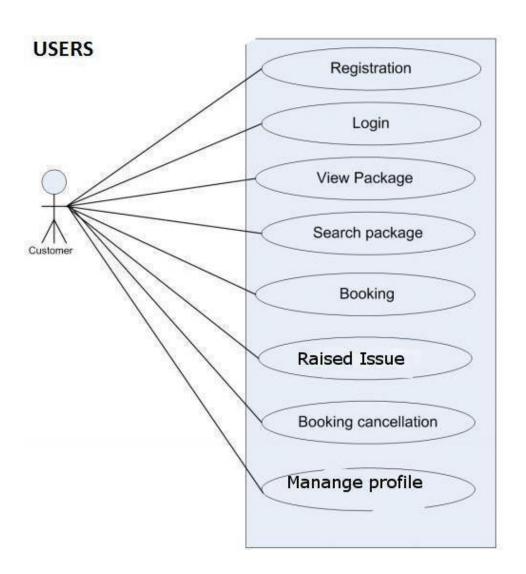
c) The following entity relationship diagram shows the graphical view of required components of the system.





# $\ \square\$ UML (Unified Modeling Language) Diagram:





# **ENTITY-RELATIONSHIP Diagrams**

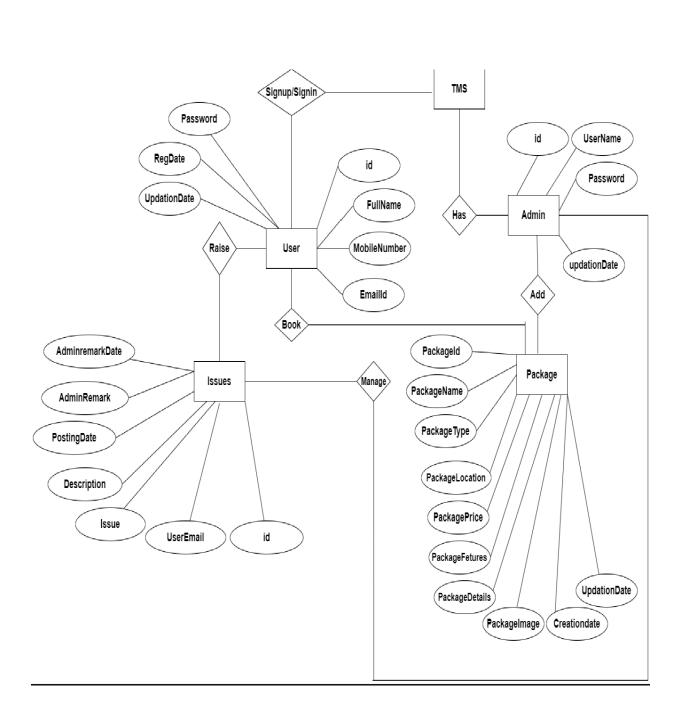
E-R (Entity-Relationship) Diagram is used to represents the relationship between entities in the table.

The symbols used in E-R diagrams are:

<u>SYMBOL</u>	<u>PURPOSE</u>
	Represents Entity sets.
	Represent attributes.
	Represent Relationship Sets.
	Line represents flow

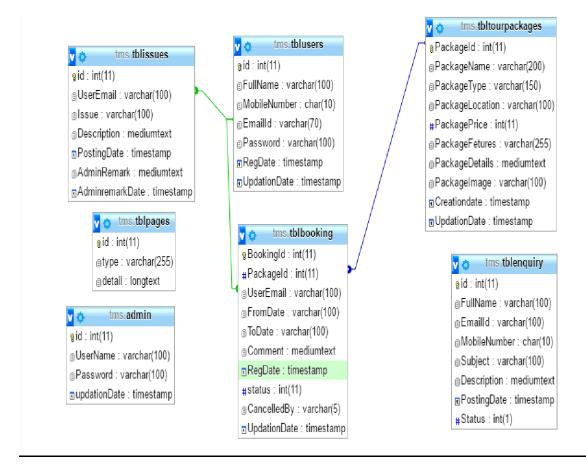
Structured analysis is a set of tools and techniques that the analyst. To develop a new kind of a system:

The traditional approach focuses on the cost benefit and feasibility analysis, Project management, and hardware and software selection a personal considerations.



## **Class Diagram**

The class diagram shows a set of classes, interfaces, collaborations and their relationships.



## **Architecture flow : (N-Tier)**

N-Tier Applications can easily implement the concepts of Distributed Application Design and Architecture. The N-Tier Applications provide strategic benefits to Enterprise Solutions. While 2-tier, client-server can help us create quick and easy solutions and may be used for Rapid Prototyping, they can easily become maintenance and security night mare The N-tier Applications provide specific advantages that are vital to the business continuity of the enterprise. Typical features of a real life n-tier may include the following:

- Security
- Availability and Scalability
- Manageability
- Easy Maintenance
- Data Abstraction

•

The above mentioned points are some of the key design goals of a successful n-tier application that intends to provide a good Business Solution.

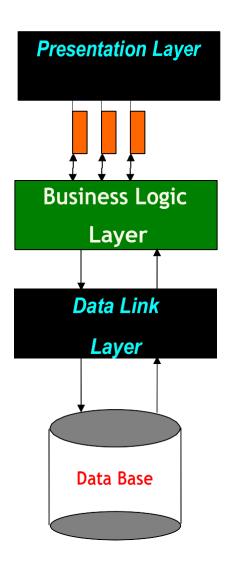
#### **Definition:**

Simply stated, an n-tier application helps us distribute the overall functionality into various tiers or layers:

- Presentation Layer
- Business Logic layer
- Data Link Layer
- Database/Data Store

Each layer can be developed independently of the other provided that it adheres to the standards and communicates with the other layers as per the specifications. This is the one of the biggest advantages of the n-tier application. Each layer can potentially treat the other layer as a 'Block-Box'. In other

words, each layer does not care how other layer processes the data as long as it sends the right data in a correct format.



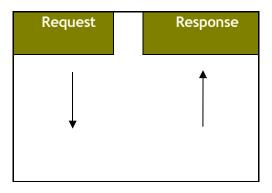


Fig: N-Tier Architecture

## 1. Presentation Layer:

Also called as client layer, comprises of components that are dedicated to presenting the data to theuser. For example: Windows/Web Forms and buttons, edit boxes, Text boxes, labels, grids, etc.

## 2. Business Logic Layer:

This layer encapsulates the Business rules or the business logic of the encapsulations. To have a separate layer for business logic is of a great advantage. This is because any changes in Business Rules can be easily handled in this layer. As long as the interface between the layers remains the same, any changes to the functionality/processing logic in this layer can be made without impacting the others. A lot of client-server apps failed to implement successfully as changing the business logicwas a painful process.

## 3. Data Link Layer:

This layer comprises of components that help in accessing the Database. If used in the right way, this layer provides a level of abstraction for the database structures. Simply put changes made to the database, tables, etc do not affect the rest of the application because of the Data Access layer. The different application layers send the data requests to this layer and receive the response from this layer.

#### 4. <u>Database Layer</u>:

This layer comprises of the Database Components such as DB Files, Tables, Views, etc. The Actual database could be created using SQL Server, Oracle, Flat files, etc. In an n-tier application, the entireapplication can be implemented in such a way that it is independent of the actual Database. For instance, you could change the Database Location with minimal changes to Data Access Layer. Therest of the Application should remain unaffected.

## Code:

This contains the information of our company

# Aboutus.php

```
<style>
.who {
 border-bottom:2px solid red;
}
p,h3 {
color: #555555;
  font-size: 16px;
  font-family: 'Open Sans', sans-serif;
  line-height: 1.6;
}
h3 {
 border-bottom:2px solid black;
 padding:10px;
.choose p {
border-left:2px solid black;
border-right:2px solid black;
}
</style>
<body><br/>style="background-color:white !important;"></body></br>
<?php include 'service-master.php';?>
<div class="container">
<h1 style="text-align:center">About us</h1>
<div class="row">
```

```
<div class="col-sm-4">
    <img class="img-responsive thumbnail" src="images/serv-6.jpg">
    </div>
    <div class="col-sm-8">
    <h2 class="who">WHO WE ARE</h2>
```

<p>We strongly believe that pursuing all of these goals is in our interest and in the interest of all of our stakeholders are us customers, employees, investors and the planet as a whole. We add value to cargo people's interaction with us, with excellent services or products.</p>

Engaging our employees and nurturinvestment on the stock holds market we show concern , by engaging our employees and nurturinvestment on the stock holds market we show concern.

```
</div>
</div>
</div>
<div>
<img class="img-responsive" src="images/about-im.png">
</div>
<div class="container">
 <h1 class="text-center">WHY CHOOSE USE</h1>
 <div class="row text-center choose">
   <div class="col-sm-4">
         <h3>24 Hours Support</h3>
                                                                    forwarding
             We
                       are
                            Specialises
                                         in
                                             international
                                                           freight
                                                                                of
merchandise.
       </div>
       <div class="col-sm-4">
         <h3>Global supply Chain</h3>
             Efficiently unleash cross-media information without cross-media
value.
```

```
</div>
       <div class="col-sm-4">
         <h3>Mobile Shipment Tracking</h3>
             Ye Offers intellgent concepts for road & tail well as complex special
services.
       </div>
       </div>
        <div class="row text-center choose">
   <div class="col-sm-6">
         <h3>Careful Handling</h3>
             Cargo HUB are transported at some stage of their journey along world's
roads.
       </div>
       <div class="col-sm-6">
         <h3>Time On Door Delivery</h3>
             Ye Offers intellgent concepts for road & tail well as complex special
<br/>services.
       </div>
       </div>
 </div>
</div>
<?php include 'footer.php';?>
</body>
```

```
Admin.php:
<?php
session_start();
require_once('library.php');
isUser();
?>
               html
<!DOCTYPE
                      PUBLIC
                                 "-//W3C//DTD
                                                 HTML
                                                          4.01
                                                                 Transitional//EN"
"http://www.w3.org/TR/html4/loose.dtd">
<html><head>
<meta http-equiv="Content-Type" content="text/html; charset=ISO-8859-1">
<title>Courier / Cargo Tracking Script in PHP - Ver 0.97</title>
k href="css/mystyle.css" rel="stylesheet" type="text/css">
k href="css/style.css" rel="stylesheet" type="text/css">
k href="css/form.css" rel="stylesheet" type="text/css">
</head>
<body>
<?php
include("header.php");
?>
<div align="center"> <br>
    <br>>
    <table
             bgcolor="#ECECEC"
                                    border="0"
                                                 cellpadding="2"
                                                                   cellspacing="2"
align="center" width="50%">
    <td
          class="Partext1"
                            bgcolor="#EEEEEE"><div
                                                       align="left"><strong>Courier
Management System </strong></div>
```

```
<td
             class="newtext"
                               bgcolor="#FFFFFF"><div
                                                          align="left"><img
src="images/arrow_white.gif" border="0" height="8" width="7"> <a href="add-
courier.php" class="REDLink">Add Shipment</a></div>
    class="newtext"
                               bgcolor="#FFFFFF"><div
     <td
                                                          align="left"><img
src="images/arrow_white.gif" border="0" height="8" width="7"> <a href="courier-
list.php" class="REDLink">List Shipment</a></div>
    <td
             class="newtext"
                               bgcolor="#FFFFF"><div
                                                          align="left"><img
src="images/arrow_white.gif"
                            border="0"
                                          height="8"
                                                       width="7"> <a
href="report.php" class="REDLink">Report</a></div>
    class="newtext"
     <td
                               bgcolor="#FFFFF"><div
                                                          align="left"><img
                            border="0"
src="images/arrow_white.gif"
                                          height="8"
                                                       width="7"> <a
href="process.php?action=logOut" class="REDLink">Logout</a></div>
    <br>>
     <br>
     <br>>
     <br>>
  </div>
```

```
 
 </body></html>
    Database.php:
<?php
// database connection config
$dbHost = 'localhost';
$dbUser = 'root';
dbPass = ";
$dbName = 'courier_db';
$dbConn = mysqli_connect ($dbHost, $dbUser, $dbPass) or die ('MySQL connect failed. '.
mysqli_error($dbConn));
mysqli_select_db($dbConn,$dbName)
                           die('Cannot
                                    select
                                          database.
                        or
mysqli_error($dbConn));
function dbQuery($sql)
{
                          47
```

```
global $dbConn;
      $result = mysqli_query($dbConn,$sql) or die(mysqli_error($dbConn));
      return $result;
}
function dbAffectedRows()
{
      global $dbConn;
      return mysqli_affected_rows($dbConn);
}
function dbFetchArray($result, $resultType = MYSQL_NUM) {
      return mysqli_fetch_array($result, $resultType);
}
function dbFetchAssoc($result)
{
      return mysqli_fetch_assoc($result);
}
function dbFetchRow($result)
      return mysqli_fetch_row($result);
}
function dbFreeResult($result)
                                           48
```

```
return mysqli_free_result($result);
  }
function dbNumRows($result)
                                return mysqli_num_rows($result);
  }
function dbSelect($dbName)
                                return mysqli_select_db($dbName);
  }
function dbInsertId()
                                return mysqli_insert_id();
  }
?>
Login.php:
<?php
session_start();
error_reporting(0);
require_once('database.php');
require_once('library.php');
$error = "";
if(isset($_POST['txtusername'])){
                                  \$error = checkUser(\$\_POST['txtusername'], \$\_POST['txtpassword'], \$\_POST['OfficeN'] + (All the properties of the proper
ame']);
                                                                                                                                                                                                                                            49
```

```
}//if
require_once('database.php');
$sql = "SELECT DISTINCT(off_name)
             FROM tbl_offices";
$result = dbQuery($sql);
?>
<?php
if(isset($_POST['registration'])){
header("registration.php");
?>
<html><head>
<meta http-equiv="Content-Type" content="text/html; charset=ISO-8859-1">
<title>Login</title>
k href="css/style.css" rel="stylesheet" type="text/css">
k href="css/mystyle.css" rel="stylesheet" type="text/css">
k href="css/form.css" rel="stylesheet" type="text/css">
<script language="javascript">
<!--
function memloginvalidate()
 if(document.form1.txtusername.value == "")
    alert("Please enter admin UserName.");
    document.form1.txtusername.focus();
    return false;
 if(document.form1.txtpassword.value == "")50
```

```
alert("Please enter admin Password.");
   document.form1.txtpassword.focus();
   return false;
  }
 }
</script></head>
<body onLoad="document.form1.txtusername.focus();">
<table id="Outer" bgcolor="#FFFFF" border="0" cellpadding="0" cellspacing="0"
align="center" width="780">
<table id="inner" border="0" cellpadding="3" cellspacing="3" height="500"
align="center" width="96%">
  k href="css/style.css" rel="stylesheet" type="text/css">
<style type="text/css">
<!--
.style2 {color: #FFFFFF}}
-->
</style>
<img src="images/trheader.jpg" height="109" width="780">
<center">
    <span class="redtext"><strong>
                                 </strong></span><br>
      <br>
```

```
</div>
  <tbody><tr>
    <img src="images/boxtopleftcorner.gif" alt="" height="13"
width="18">
    <img src="images/boxtoprightcorner.gif" alt="" height="13"
width="18">
   cellspacing="0"
           border="0"cellpadding="0"
                                   align="center"
    le
width="98%">
     border="0"
                         cellpadding="0"
                                  cellspacing="0"
     <td
        height="18"><table
width="100%">
      <table class="GreenBox" border="0" cellpadding="0" cellspacing="0"
align="center" width="300">
```

```
name="form1"
                          id="form1"
                                   method="post"
         <form
                                              onSubmit="return
memloginvalidate()">
                    bgcolor="#FFFFFF"
                                    border="0"
                                               cellpadding="3"
          <tdble
cellspacing="1" width="100%">
            
           <div class="headtext13" align="center"><strong>Administrator Login
Area </strong></div>
           <font color="#FF0000" style="font-size:12px;">
        <?php echo $error; ?>
        </font>
        width="115">    <font
                                                 style="font-
            <td
size:12px;">Username</font>
            :
            <input name="txtusername" class="forminput" id="txtusername" maxlength="30"</pre>
type="text">
               <font
                                                  style="font-
size:12px;">Password</font>.
                              53
```

```
:
              <input name="txtpassword" class="forminput" id="txtpassword"
maxlength="20" type="password">
             >
                  <fontstyle="font-
size:12px;">Office</font>
              :
              <select class="select" name="OfficeName">
               <?php
               while($data = dbFetchAssoc($result)){
               ?>
               <option value="<?php echo $data['off_name']; ?>"><?php echo</pre>
$data['off_name']; ?></option>
               <?php
               }//while
               ?>
               </select>
                
               
              <input name="Submit" class="green-button" value="Login"
type="submit" style="">
```

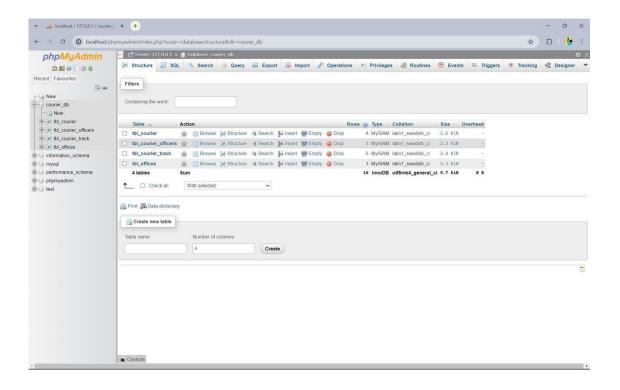
```
</form>
      
    <img src="images/boxbtmleftcorner.gif" alt="" height="12"
width="18">
   <img src="images/boxbtmrightcorner.gif" alt="" height="12"
width="18">
  <br>>
  <br/>td>
 ="0" cellpadding="0" cellspacing="0" align="center" width="780">
 
                55
```

```
<div align="right"></div>
</body></html>
    Cargo.php:
    <?php include 'service-master.php';?>
    <div class="container">
    <h1 style="text-align:center">Cargo Services</h1>
    </div>
    <?php include 'service-description.php'; ?>
    <?php include 'footer.php';?>
```

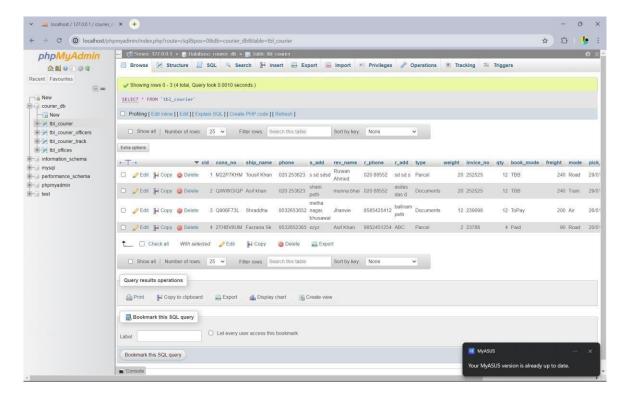
#### **Database Design**

The data in the system has to be stored and retrieved from database. Designing the database is part of system design. Data elements and data structures to be stored have been identified at analysis stage. They are structured and put together to design the data storage and retrieval system. A database is a collection of interrelated data stored with minimum redundancy to serve many users quickly and efficiently. The general objective is to make database access easy, quick, inexpensive and flexible for the user. Relationships are established between the data items and unnecessary data items are removed. Normalization is done to get an internal consistency of data and to have minimum redundancy and maximum stability. This ensures minimizing data storage required, minimizing chances of data inconsistencies and optimizing for updates. The MySQL database has been chosen for developing the relevant databases.

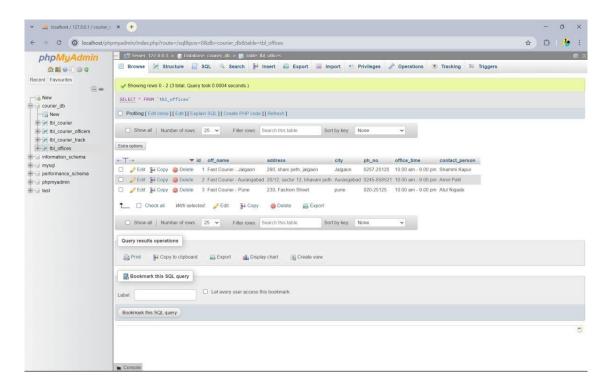
#### **Database tables and Structure:**



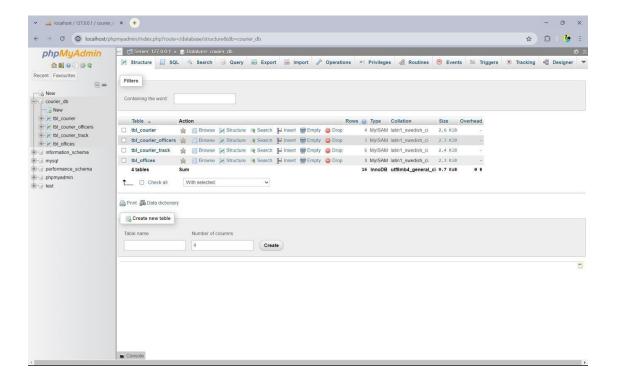
#### Table-courier:



#### Table-offices:

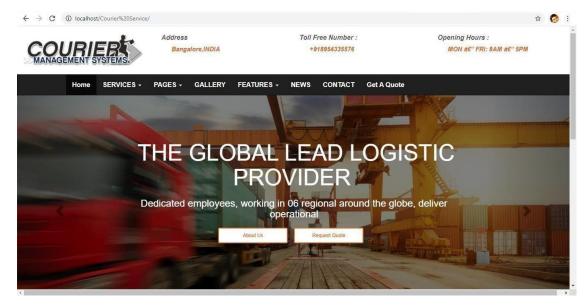


#### Courier-database:



#### **Output Screen of the Project (Screenshots)**

• It is the main Home page of the application. In this we had mentioned various login. And the user can able to send the quotes to the courier office

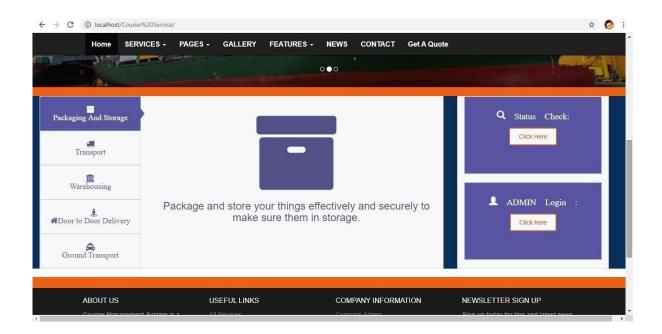


- In this page the employee can able to register their account in the database by providing this all the fields.

  This all values will be saved in the database and this value will be used to login as the Employee
- LOGO

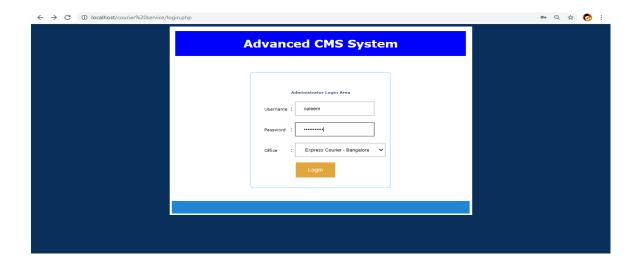


## • Employee Login Page:



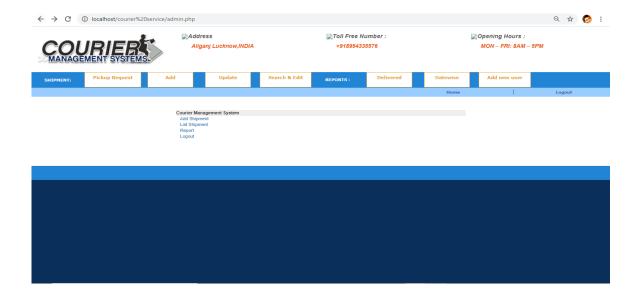
• In this page the Employee can able to login with the register Username and password once the username and password is correct then only the user can able to login otherwise it will display the error message.

## Login page:



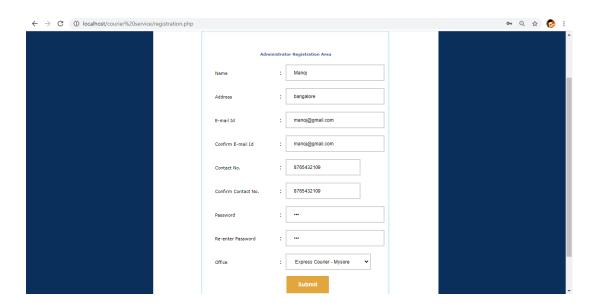
## • Admin Home Page:

• This is the admin dashboard here admin can able to add the courier details.



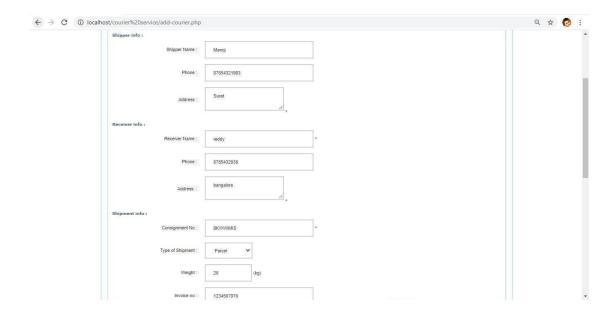
## • Courier officers Register Page:

• In this page the admin is able to add the new officers by providing their personal details and login details.



#### • View Profile:

• In this the employee can able to add the courier details by filling this all the details.



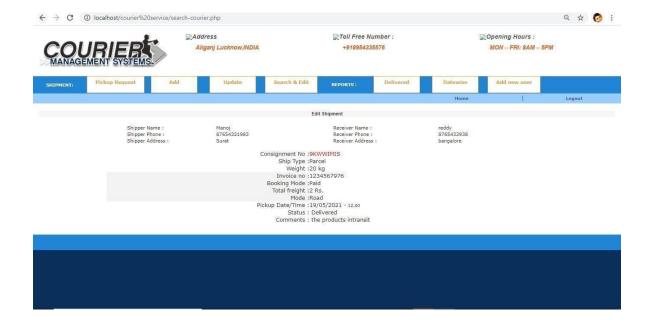
#### View Courier Details:

• In this page the employee can able to view the list of courier details which will be added by the courier officers.



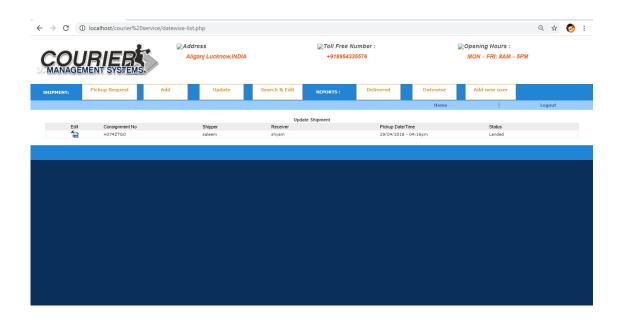
#### • View Courier details:

• In this page the admin can able to view the individual courier status



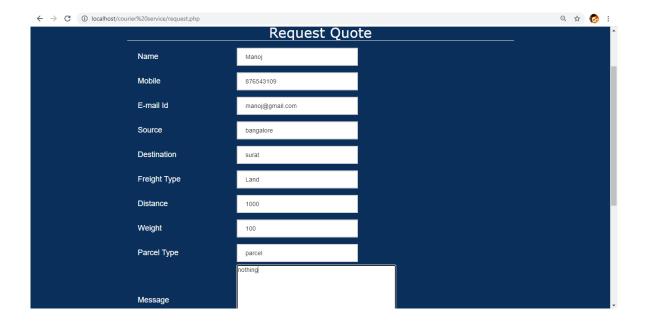
#### Date wise courier search:

• In this page the admin can able to view the date wise courier status



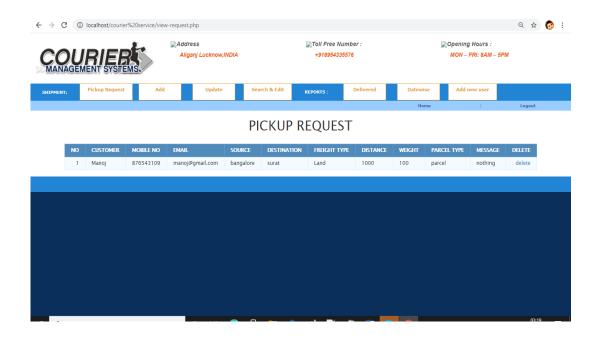
## • Request Quote:

• In this the user can able to send the quotation to the admin for the courier



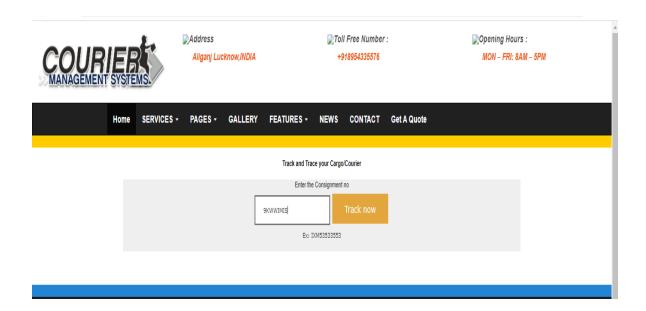
#### Viewing the Quotation:

• In this page the admin can able to view the quotation which will be send by the admin

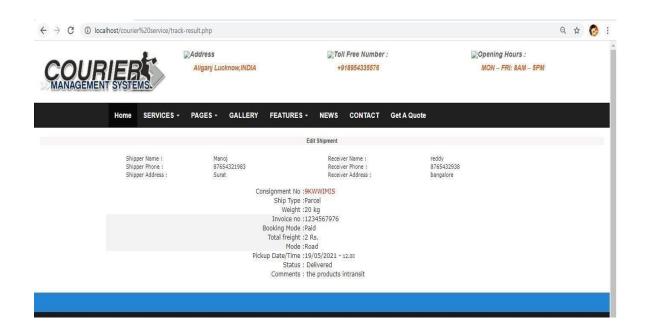


#### • Track Courier Status:

• In this page the user can able to track their courier status by entering the consignment number



- Track Courier Status:
- In this page the customer can able to view the details of the courier



## **CONCLUSION**

The Courier Management System developed during this project represents a significant step forward in the efficient and effective management of courier services. By leveraging modern technologies such as Power BI for data visualization and Python for backend processing, the system provides robust functionality, user-friendly interfaces, and comprehensive data analytics capabilities.

Through this project, we successfully implemented core features such as parcel booking, tracking, user management, and reporting. The system's N-Tier architecture ensures scalability, maintainability, and security, making it a reliable solution for real-world applications. Additionally, our focus on data-driven insights allows for continuous improvement and optimization of courier operations.

This project has not only provided a practical solution for managing courier services but also offered valuable learning experiences in software development, database management, and data visualization. The skills and knowledge gained from this project are Instrumental in advancing our expertise in the field of data science and application development.

Moving forward, the system can be expanded with additional features such as real-time tracking, mobile app integration, and advanced analytics to further enhance its capabilities and user experience. This project lays a strong foundation for future innovations in courier management and demonstrates the potential of integrating technology to solve complex logistical challenges.

# **BIBILIOGRAPHY**

#### **□** ONLINE REFERENCES:

- www.python.org
- www.corsea.org
- www.wikipedia.org
- <a href="https://www.w3schools.com/">https://www.w3schools.com/</a>
- https://chatgpt.com/
- https://www.geeksforgeeks.org/css-syntax/?ref=lbp
- https://htmlhelp.com/reference/css/structure.html
- https://www.programiz.com/javascript/online-compiler/