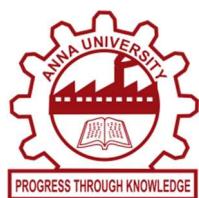


**CS6106 – DATABASE MANAGEMENT SYSTEM**

**Feb - May 2021**

# **COURIER MANAGEMENT SYSTEMS**

**Ajitesh M - 2019103503**



---

# Table of Contents

S.No	Title	Page No
1.	Objectives	3
2.	Abstract	4
3.	Introduction	5
4.	Database Design	
	1. ER – Diagram	7
	2. Schema design	9
5.	Implementation	
	1. Creating relations	10
	2. Creating views	12
	3. Creating triggers	15
	4. Creating events	16
	5. Inserting values	17
	6. User Interface design	21
	7. Connecting to database	27
	8. Staff login	28
	9. Staff details	31
	10.Logout	32
	11.Placing parcel	33
	12.Update delivery status	37
	13.Parcel tracking	40
	14.History of consignments	43
	15.Feedback/query filing	46
	16.Employee of the Month	48
	17.Branch details	49
6.	Conclusion	50
7.	References	51

---

# Objectives

The main objective of this mini-project is to explore the various database management concepts and implement them in a real-time application, which is achieved by developing an interactive web application which manages several logistics routines such as placing, updating and tracking delivery packages.

The Courier Management System proposed here avoids spending more manual hours in maintaining records and generating reports. Daily transactions are managed easily and the centralized database helps in avoiding conflicts between different branches and placing orders in a large scale. It also provides secured data storage.

I wished to develop a customer-friendly flexible delivery policy in order to tackle the issues me and my friends faced recently with one of the well-known courier service.

Along with my keen interests in web development and databases, I wanted to use this opportunity to explore and get hands-on experience with working on real time project. By the end of this mini-project, I would be able to improve my knowledge and experience with databases to the higher level.

---

# Abstract

## Courier Management System

CC Couriers - An interactive courier management application which provides the facilities of remote tracking and feedback system to customers and the facilities of placing a new parcel, updating the status of existing ones and display the history of consignments to staffs. Recently, I faced an issue with the delivery policy of one of the well known courier service. As I was unable to receive my parcel, I requested a change in delivery address to a friend of mine residing in the same town. But he was unable to do so because of their delivery terms and conditions. In order to tackle this issue faced by many of us, I've implemented a facility to change the delivery address anytime before it is out for delivery.

---

## Introduction

The project emphasizes the planning and execution of delivery of packages and goods by a Courier service company named **CC COURIERS** using a Courier Management System. The system maintains details about the branches of the company, staffs working under the company, parcels/packages placed, delivery status of the parcels and the pricing list for the parcels between various states. The system is used for daily activities such as placing orders, loading them, delivery status checking and managing the branches and more. It is very difficult to do these processes manually since they are much time-consuming and paper work needed for it also increases. Hence it is recommended to computerize the process by developing the relative software as the world is turning into information and technology. By doing so, the task of handling the parcels will get easier, quicker and also more efficient. Feedback/query system is also provided.

The Courier Management System proposed here avoids spending more manual hours in maintaining records and generating reports. Daily transactions are managed easily and the centralized database helps in avoiding conflicts between different branches and placing orders in a large scale. It avoids human errors in feeding parcel details or handling records, and provides support to customers by enabling them to track the status of their parcels. It also manages data in a secure manner and provides a GUI that does not require special training to make people work upon.

This is an interactive web application developed using PHP and MySQL database. HTML5 and CSS3, along with Bootstrap 4, is used for front-end development.

This web application provides the facility of tracking the package for the customers, who can access the delivery status of their package using an unique Tracking ID, which displays the date-time details of stages such as received and shipped from the origin office, out for delivery as received by the destination office and delivered to the receiver successfully.

Also provides the facility to update the receiver's address by another address of the same city anytime before it is out for delivery from the destination office. Feedback/query system is also provided.

For the employees, it provides the facilities of placing new order and update the delivery status of the orders. The delivery charge depends upon the weight of the parcel, source and destination state.

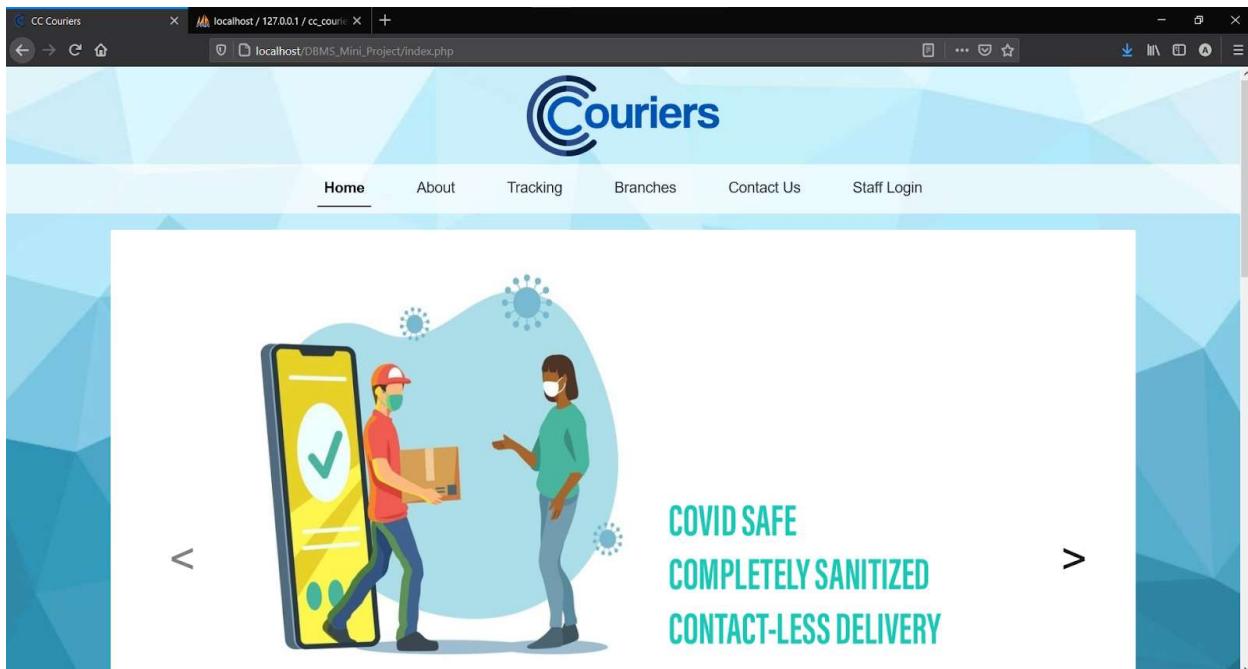


Fig 1 : Home page of the application

### Salient Features :

1. Remote tracking and delivery address update if needed.
2. Placing new a parcel.
3. Updating the delivery status of a parcel.
4. History of consignments delivered and yet to be delivered.
5. Feedback/query support.

# Database design

## ER - Diagram

The below mentioned Entity-Relationship diagram (ER -diagram) describes the structure of the Courier management system database.

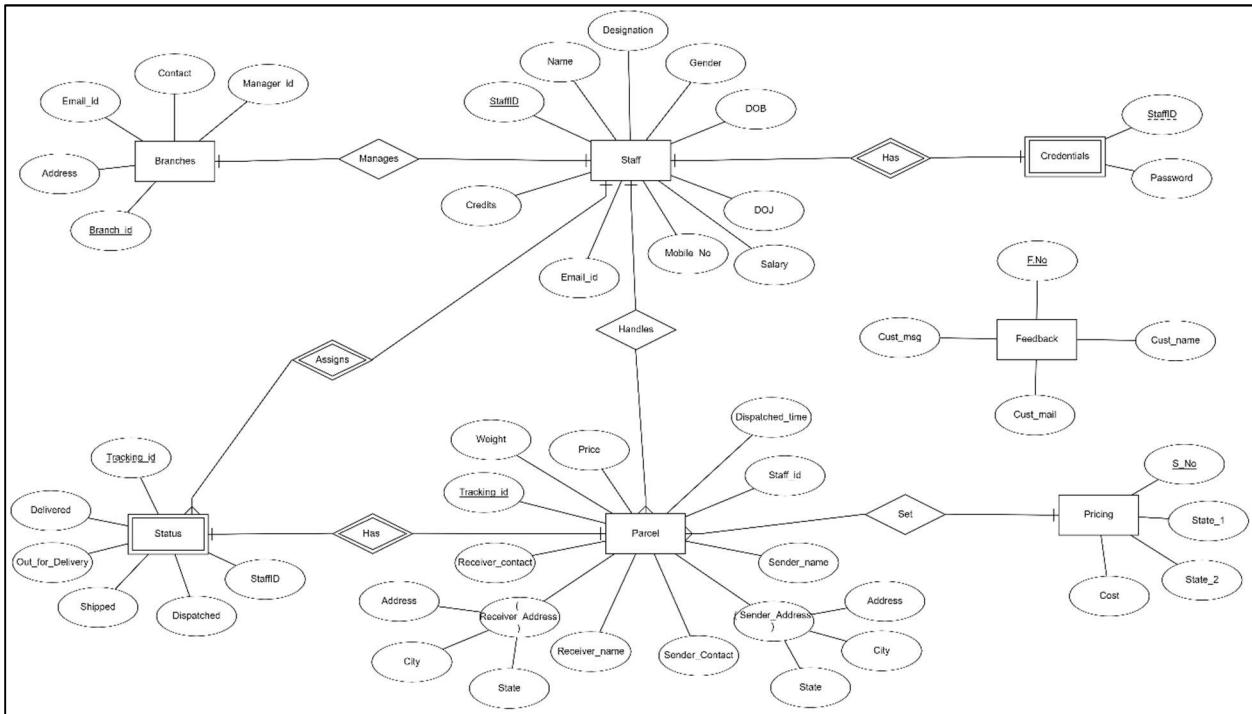


Fig 2 : Entity-relationship diagram of the database

## DESCRIPTION OF THE ER MODEL:

The Courier Management System database has various entities namely Staff, Credentials, Branches, Parcel, Status, Pricing and Feedback.

---

Staff entity stores the details of the staffs who work for the courier management service. It has a one-to-one relationship with the Credentials entity as every staff has a unique staff\_id and a password and every credential refer to a unique staff. The staff entity also holds a one-to-one relationship with the Branches entity because each branch is managed by a unique staff and each staff manages a unique branch. The staff entity holds a one-to-many relationship with the Parcel entity since a staff can handle many parcels while a parcel could only be handled by a single staff. The staff entity also holds a one-to-many relationship with the Status entity as a staff can assign the status of any number of parcels whereas a status of a parcel could only be assigned by a single staff.

The Parcel entity has a many-to-one relationship with the Pricing entity as a pricing could be set to any number of parcels and they may have the same price whereas a single parcel could have only one pricing. The Parcel entity has a one-to-one relationship with the Status entity as a single parcel could have a single status and a single status of a tracking\_id could have a unique parcel.

# Schema design

The schema design mentioned below is a skeleton structure that represents the logical view of the Courier Management System database as a whole. This database schema design organizes the data into separate entities, determines how to create relationships between organized entities, and how to apply the constraints on the data.

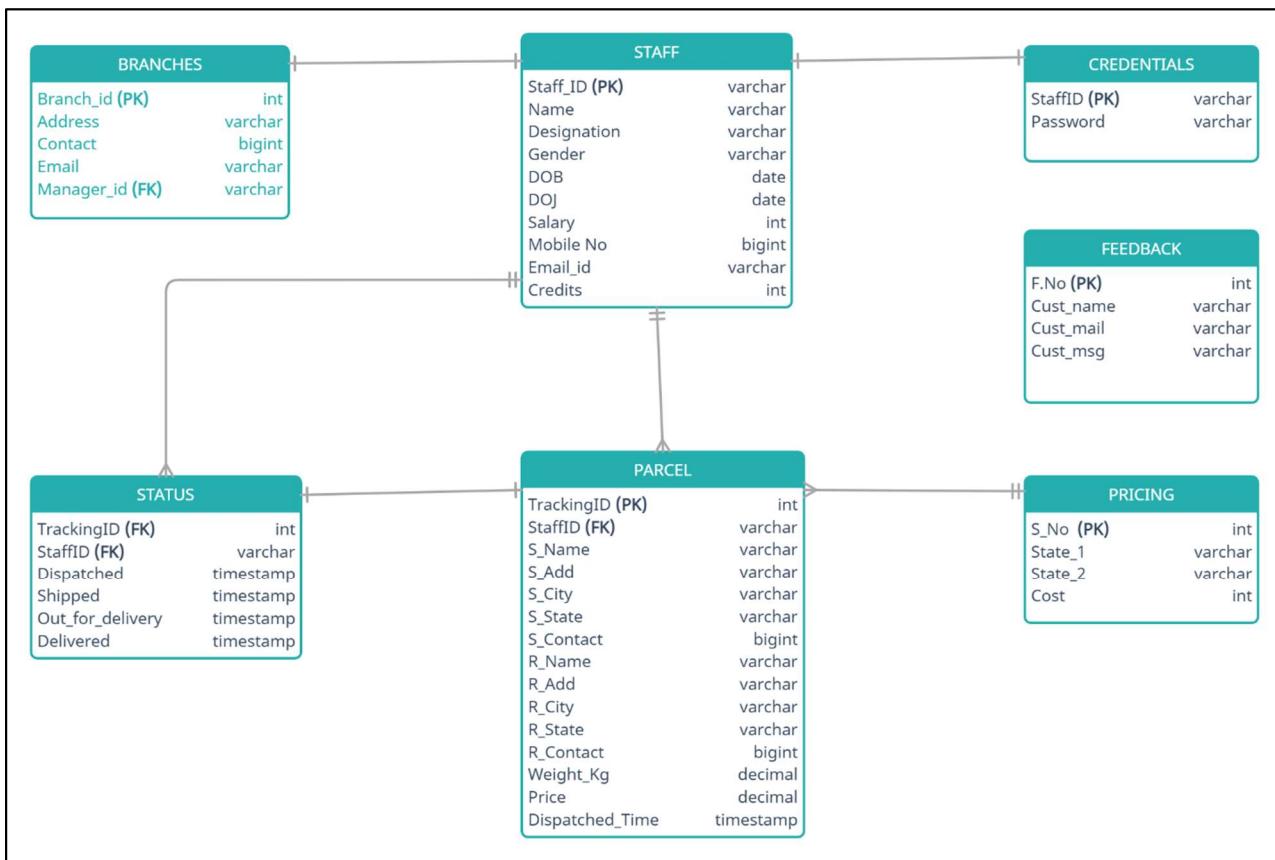


Fig 3 : Relation schema model

This schema design elucidates about the relationship each table is having with others. It also says about how the primary key and foreign key constraints are used in order to provide relationship between the tables.

All the tables have been normalized and are present in **Boyce – Codd Normal Form** since all the transitive functional dependencies have been removed by decomposing the tables.

---

# IMPLEMENTATION

This interactive web application is completely responsive thereby suitable for all screen sizes and user-friendly. The user interface is developed used HTML5, CSS3 and Bootstrap 4 which structures the application, styles its contents and develops mobile-friendly responsiveness respectively (Frontend development). The server-side scripting is implemented in PHP7, which is a server scripting language, and a powerful tool for making dynamic and interactive Web pages. Along with MySQL database management system, which is a widely-used relational DBMS comprises of the Backend development stack.

The PHP development environment, XAMPP, is a free and open-source cross-platform web server solution stack package developed by Apache Friends, consisting mainly of the Apache HTTP Server, MariaDB database, and interpreters for scripts written in the PHP and Perl programming languages. XAMPP v.3.2.3 used in this project provides Apache web server and MySQL database.

The various stages of the implementation of this project are briefed as follows,

## 1. CREATING RELATIONS :

The various relations(tables) used to represent the data in the database include :

```
STAFF (StaffID, Name, Designation, Gender, DOB, DOJ, Salary, Mobile, Email, Credits)
CREDENTIALS (StaffID, Pwd)
BRANCHES (Branch_id, Address, Contact, Email, Manager_id)
PARCEL (TrackingID, StaffID, S_Name, S_Add, S_City, S_State, S_Contact, R_Name, R_Add, R_City, R_State, R_Contact, Weight_Kg, Price, Dispatched_Time)
PRICING (S.No, State_1, State_2, Cost)
STATUS (TrackingID, StaffID, Dispatched, Shipped, Out_for_delivery, Delivered)
FEEDBACK ( F.No, Cust_name, Cust_mail, Cust_msg)
```

Server: 127.0.0.1 » Database: cc\_couriers

Structure SQL Search Query Export Import Operations Privileges

Run SQL query/queries on database cc\_couriers:

```

1 CREATE TABLE `credentials` (`StaffID` varchar(30) NOT NULL, `Pwd` varchar(30) NOT NULL);
2
3 CREATE TABLE `staff` (`StaffID` varchar(30) NOT NULL, `Name` varchar(30) NOT NULL,
4 `Designation` varchar(30) NOT NULL, `Gender` varchar(10) NOT NULL,
5 `DOB` date NOT NULL, `DOJ` date NOT NULL, `Salary` int(11) NOT NULL,
6 `Mobile` bigint(20) NOT NULL, `Email` varchar(30) NOT NULL,
7 `Credits` int(11) NOT NULL DEFAULT '0' );
8
9 CREATE TABLE `branches` (`Branch_id` int(11) NOT NULL, `Address` varchar(100) NOT NULL,
10 `Contact` bigint(20) NOT NULL, `Email` varchar(40) NOT NULL,
11 `Manager_id` varchar(30) NOT NULL);
12
13 CREATE TABLE `pricing` (`S.No` int(11) NOT NULL, `State_1` varchar(30) NOT NULL,
14 `State_2` varchar(30) NOT NULL, `Cost` int(11) NOT NULL);
15
16 CREATE TABLE `parcel` (`TrackingID` int(11) NOT NULL, `StaffID` varchar(30) NOT NULL,
17 `S_Name` varchar(30) NOT NULL, `S_Add` varchar(50) NOT NULL,
18 `S_City` varchar(20) NOT NULL, `S_State` varchar(20) NOT NULL,
19 `S_Contact` bigint(20) NOT NULL, `R_Name` varchar(30) NOT NULL,
20 `R_Add` varchar(50) NOT NULL, `R_City` varchar(20) NOT NULL,
21 `R_State` varchar(20) NOT NULL, `R_Contact` bigint(20) NOT NULL,
22 `Weight_Kg` decimal(10,2) NOT NULL, `Price` decimal(10,2) NOT NULL,
23 `Dispatched_Time` timestamp NOT NULL DEFAULT CURRENT_TIMESTAMP);
24
25 CREATE TABLE `status` (`TrackingID` int(11) NOT NULL, `StaffID` varchar(30) NOT NULL,
26 `Dispatched` timestamp NULL DEFAULT NULL,
27 `Shipped` timestamp NULL DEFAULT NULL,
28 `Out_for_delivery` timestamp NULL DEFAULT NULL,
29 `Delivered` timestamp NULL DEFAULT NULL);
30
31 CREATE TABLE `feedback` (`F.No` int(11) NOT NULL, `Cust_name` varchar(30) NOT NULL,
32 `Cust_mail` varchar(50) NOT NULL, `Cust_msg` varchar(500) NOT NULL);
33

```

Fig 4.1 : SQL queries to create tables

phpMyAdmin

Server: 127.0.0.1 » Database: cc\_couriers

Structure SQL Search Query Export Import Operations Routines Events

Filters

Containing the word:

Table	Action	Rows	Type	Collation	Size	Overhead
branches	<input type="checkbox"/> Browse <input type="checkbox"/> Structure <input type="checkbox"/> Search <input type="checkbox"/> Insert <input type="checkbox"/> Empty <input type="checkbox"/> Drop	5	InnoDB	latin1_swedish_ci	32 KiB	-
credentials	<input type="checkbox"/> Browse <input type="checkbox"/> Structure <input type="checkbox"/> Search <input type="checkbox"/> Insert <input type="checkbox"/> Empty <input type="checkbox"/> Drop	11	InnoDB	latin1_swedish_ci	16 KiB	-
feedback	<input type="checkbox"/> Browse <input type="checkbox"/> Structure <input type="checkbox"/> Search <input type="checkbox"/> Insert <input type="checkbox"/> Empty <input type="checkbox"/> Drop	5	InnoDB	latin1_swedish_ci	16 KiB	-
parcel	<input type="checkbox"/> Browse <input type="checkbox"/> Structure <input type="checkbox"/> Search <input type="checkbox"/> Insert <input type="checkbox"/> Empty <input type="checkbox"/> Drop	22	InnoDB	latin1_swedish_ci	32 KiB	-
pricing	<input type="checkbox"/> Browse <input type="checkbox"/> Structure <input type="checkbox"/> Search <input type="checkbox"/> Insert <input type="checkbox"/> Empty <input type="checkbox"/> Drop	8	InnoDB	latin1_swedish_ci	16 KiB	-
staff	<input type="checkbox"/> Browse <input checked="" type="checkbox"/> Structure <input type="checkbox"/> Search <input type="checkbox"/> Insert <input type="checkbox"/> Empty <input type="checkbox"/> Drop	11	InnoDB	latin1_swedish_ci	32 KiB	-
status	<input type="checkbox"/> Browse <input type="checkbox"/> Structure <input type="checkbox"/> Search <input type="checkbox"/> Insert <input type="checkbox"/> Empty <input type="checkbox"/> Drop	18	InnoDB	latin1_swedish_ci	16 KiB	-
7 tables	<input type="checkbox"/> Sum	80	InnoDB	latin1_swedish_ci	160 KiB	0 B

Check all With selected:

Fig 4.2 : Created tables in the database

## 2. CREATING VIEWS :

A view is a virtual table based on the result-set of an SQL statement. A view contains rows and columns, just like a real table. The fields in a view are fields from one or more real tables in the database.

Here two different views are created on the relations parcel and status.

The screenshot shows the phpMyAdmin interface for a database named 'cc\_couriers'. The top navigation bar includes tabs for Structure, SQL, Search, Query, Export, Import, Operations, Privileges, and Routines. Below the tabs, there is a 'Filters' section with a search input field labeled 'Containing the word:' and a dropdown menu 'With selected:'.

Table	Action	Rows	Type	Collation	Size	Overhead
arrived	Browse Structure Search Insert Drop View	~0	View	---	-	-
delivered	Browse Structure Search Insert Drop View	~0	View	---	-	-
2 tables Sum		~0	InnoDB latin1_swedish_ci	0 B	0 B	

Fig 5.1 : List of views in the database

- Arrived view:** stores the details of consignments in transit.

The screenshot shows the phpMyAdmin interface for a database named 'cc\_couriers'. The top navigation bar includes tabs for Structure, SQL, Search, Query, Export, and Import. The 'SQL' tab is active.

The main area displays an SQL query to create a view named 'arrived':

```
1 CREATE OR REPLACE VIEW arrived AS
2 SELECT P.*, S.Shipped, S.Out_for_delivery, S.Delivered
3 FROM parcel P, status S
4 WHERE P.TrackingID = S.TrackingID AND S.Delivered is NULL;
```

Fig 5.2 : SQL query to create the view - Arrived

The screenshot shows the phpMyAdmin interface for the 'cc\_couriers' database. The left sidebar lists various tables and views, including 'arrived' and 'delivered'. The main area displays the 'Relation view' of the 'arrived' view, which contains 18 columns:

#	Name	Type	Collation	Attributes	Null	Default	Comments	Extra
1	TrackingID	int(11)			No	0		
2	StaffID	varchar(30)	latin1_swedish_ci		No	None		
3	S_Name	varchar(30)	latin1_swedish_ci		No	None		
4	S_Add	varchar(50)	latin1_swedish_ci		No	None		
5	S_City	varchar(20)	latin1_swedish_ci		No	None		
6	S_State	varchar(20)	latin1_swedish_ci		No	None		
7	S_Contact	bigint(20)			No	None		
8	R_Name	varchar(30)	latin1_swedish_ci		No	None		
9	R_Add	varchar(50)	latin1_swedish_ci		No	None		
10	R_City	varchar(20)	latin1_swedish_ci		No	None		
11	R_State	varchar(20)	latin1_swedish_ci		No	None		
12	R_Contact	bigint(20)			No	None		
13	Weight_Kg	decimal(10,2)			No	None		
14	Price	decimal(10,2)			No	None		
15	Dispatched_Time	timestamp			No	0000-00-00 00:00:00		
16	Shipped	timestamp			Yes	NULL		
17	Out_for_delivery	timestamp			Yes	NULL		
18	Delivered	timestamp			Yes	NULL		

Fig 5.3 : Structure of the view - Arrived

ii. **Delivered view** : stores the details of the consignments delivered.

The screenshot shows the phpMyAdmin interface with the 'Structure' tab selected. A SQL query is entered in the 'Run SQL query/queries on database cc\_couriers:' field:

```

1 CREATE OR REPLACE VIEW delivered AS
2 SELECT P.*, S.Shipped, S.Out_for_delivery, S.Delivered
3 FROM parcel P, status S
4 WHERE P.TrackingID = S.TrackingID AND S.Delivered is NOT NULL;

```

Fig 5.4 : SQL query to create view – Delivered

The screenshot shows the phpMyAdmin interface for the 'cc\_couriers' database. The left sidebar lists various schemas and tables. The 'Views' section contains a 'New' folder and a 'delivered' folder, which is currently selected. The main area displays the 'Table structure' for the 'delivered' view. The table has 18 columns:

#	Name	Type	Collation	Attributes	Null	Default	Comments	Extra
1	TrackingID	int(11)			No	0		
2	StaffID	varchar(30)	latin1_swedish_ci		No	None		
3	S_Name	varchar(30)	latin1_swedish_ci		No	None		
4	S_Add	varchar(50)	latin1_swedish_ci		No	None		
5	S_City	varchar(20)	latin1_swedish_ci		No	None		
6	S_State	varchar(20)	latin1_swedish_ci		No	None		
7	S_Contact	bigint(20)			No	None		
8	R_Name	varchar(30)	latin1_swedish_ci		No	None		
9	R_Add	varchar(50)	latin1_swedish_ci		No	None		
10	R_City	varchar(20)	latin1_swedish_ci		No	None		
11	R_State	varchar(20)	latin1_swedish_ci		No	None		
12	R_Contact	bigint(20)			No	None		
13	Weight_Kg	decimal(10,2)			No	None		
14	Price	decimal(10,2)			No	None		
15	Dispatched_Time	timestamp			No	0000-00-00 00:00:00		
16	Shipped	timestamp			Yes	NULL		
17	Out_for_delivery	timestamp			Yes	NULL		
18	Delivered	timestamp			Yes	NULL		

Fig 5.5 : Structure of the view – Delivered

### 3. CREATING TRIGGERS :

A trigger is a stored procedure in database which automatically invokes whenever a special event in the database occurs.

placeParcel : Trigger to insert the values of the attributes -TrackingId and dispatched\_time - of parcel relation into the status relation after inserting a new tuple in the parcel relation.



The screenshot shows the MySQL Workbench interface. The title bar indicates the server is 127.0.0.1 and the database is cc\_couriers. The toolbar has tabs for Structure, SQL, Search, Query, Export, and Import. The SQL tab is selected. Below the tabs is a search bar with the placeholder "Run SQL query/queries on database cc\_couriers:". The main area contains the following SQL code:

```
1 DELIMITER $$  
2 CREATE TRIGGER `placeParcel` AFTER INSERT ON `parcel` FOR EACH ROW BEGIN  
3     UPDATE staff SET Credits=Credits+5 WHERE StaffID=NEW.StaffID;  
4  
5     INSERT INTO status (TrackingID, StaffID, Dispatched)  
6     VALUES ( NEW.TrackingID, NEW.StaffID, NEW.Dispatched_Time);  
7 END  
8 $$  
9 DELIMITER ;|
```

Fig 6.1 : SQL query to create the trigger - placeParcel



The screenshot shows the MySQL Workbench Triggers editor. The title bar says "Triggers". The table lists one trigger:

Name	Table	Action	Time	Event
<input type="checkbox"/> placeParcel	parcel	Edit  Export  Drop	AFTER	INSERT

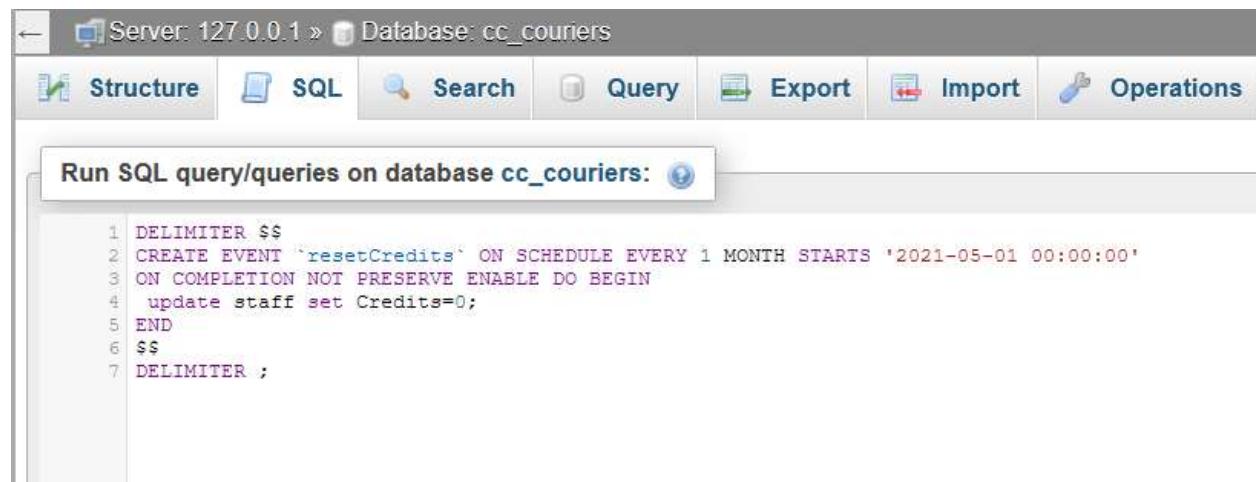
Below the table are buttons for "Check all" and "With selected: Export Drop".

Fig 6.2 : Result of successful creation of the trigger

#### 4. CREATING EVENTS :

MySQL Events are named object which contains one or more SQL statement. They are stored in the database and executed at one or more intervals, say once every week or month.

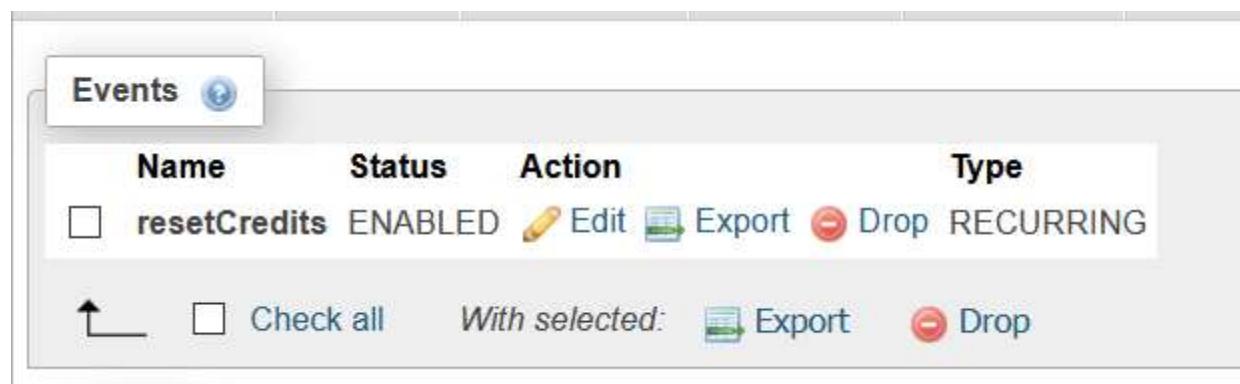
resetCredits: Event which resets the credits received by each staff to 0 at the beginning of every month in order to award - the Employee of the Month.



The screenshot shows the MySQL Workbench interface. The title bar indicates the connection is to 'Server 127.0.0.1' and the database is 'cc\_couriers'. The toolbar includes 'Structure', 'SQL', 'Search', 'Query', 'Export', 'Import', and 'Operations'. The main area is a query editor titled 'Run SQL query/queries on database cc\_couriers'. It contains the following SQL code:

```
1 DELIMITER $$  
2 CREATE EVENT `resetCredits` ON SCHEDULE EVERY 1 MONTH STARTS '2021-05-01 00:00:00'  
3 ON COMPLETION NOT PRESERVE ENABLE DO BEGIN  
4     update staff set Credits=0;  
5 END  
6 $$  
7 DELIMITER ;
```

Fig 7.1 : SQL query to create event - resetCredits



The screenshot shows the 'Events' tab in MySQL Workbench. A table lists the created event:

Name	Status	Action	Type
resetCredits	ENABLED	Edit  Export  Drop	RECURRING

Below the table are buttons for 'Check all' and 'With selected: Export Drop'.

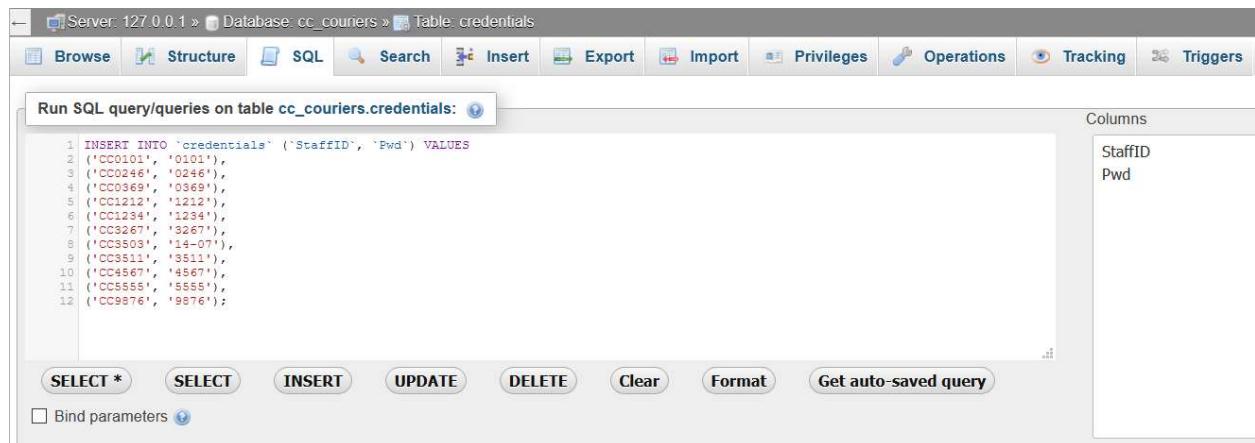
Fig 7.2 : Result of successful creation of the event

## 5. INSERTING VALUES INTO THE TABLES :

Data is populated into credentials, staff, branches and pricing tables by the database administrator.

Data in the remaining tables – parcel, status and feedback – are populated as the staff places and updates the parcel details and the customer files a feedback/query.

1. **Credentials relation** : Stores the login credentials (Staff Id and password) of each of the employee.

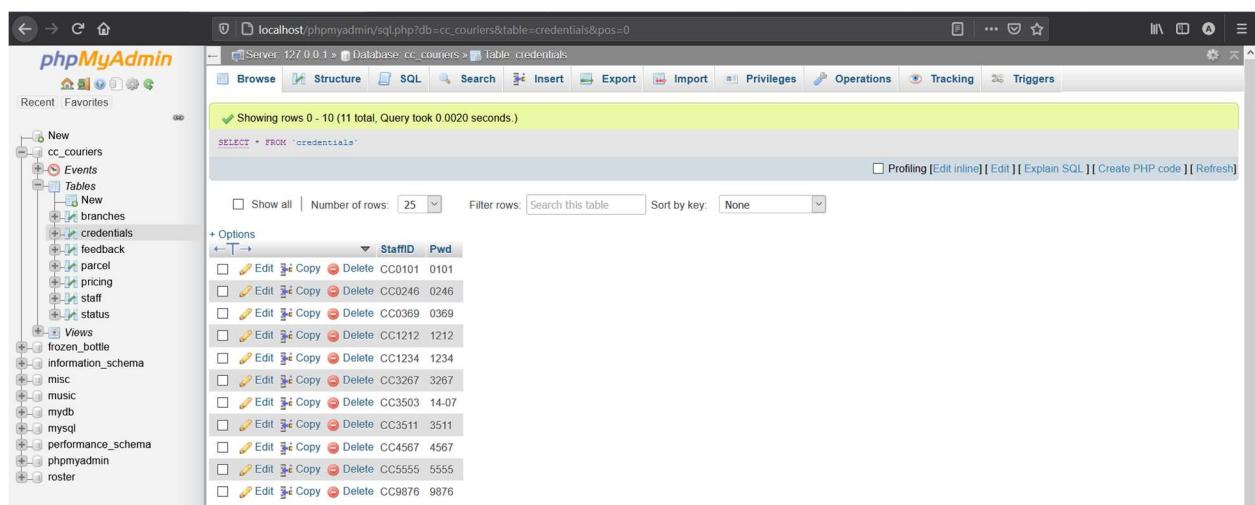


The screenshot shows the MySQL Workbench interface. The title bar indicates the connection is to 'Server: 127.0.0.1' and the database is 'cc\_couriers'. The current table is 'credentials'. The toolbar has various tabs like Browse, Structure, SQL, Search, Insert, Export, Import, Privileges, Operations, Tracking, and Triggers. Below the toolbar, there's a button for 'Run SQL query/queries on table cc\_couriers.credentials:'. A code editor window contains the following SQL query:

```
1 INSERT INTO `credentials` (`StaffID`, `Pwd`) VALUES
2 ('CC0101', '0101'),
3 ('CC0246', '0246'),
4 ('CC0369', '0369'),
5 ('CC1212', '1212'),
6 ('CC1234', '1234'),
7 ('CC3267', '3267'),
8 ('CC3503', '14-07'),
9 ('CC3511', '3511'),
10 ('CC4567', '4567'),
11 ('CC5555', '5555'),
12 ('CC9876', '9876');
```

Below the code editor are several buttons: SELECT \*, SELECT, INSERT, UPDATE, DELETE, Clear, Format, and Get auto-saved query. There is also a checkbox for Bind parameters. To the right of the code editor, there's a 'Columns' panel showing the columns StaffID and Pwd.

Fig 8.1 : SQL query to insert data into credentials table

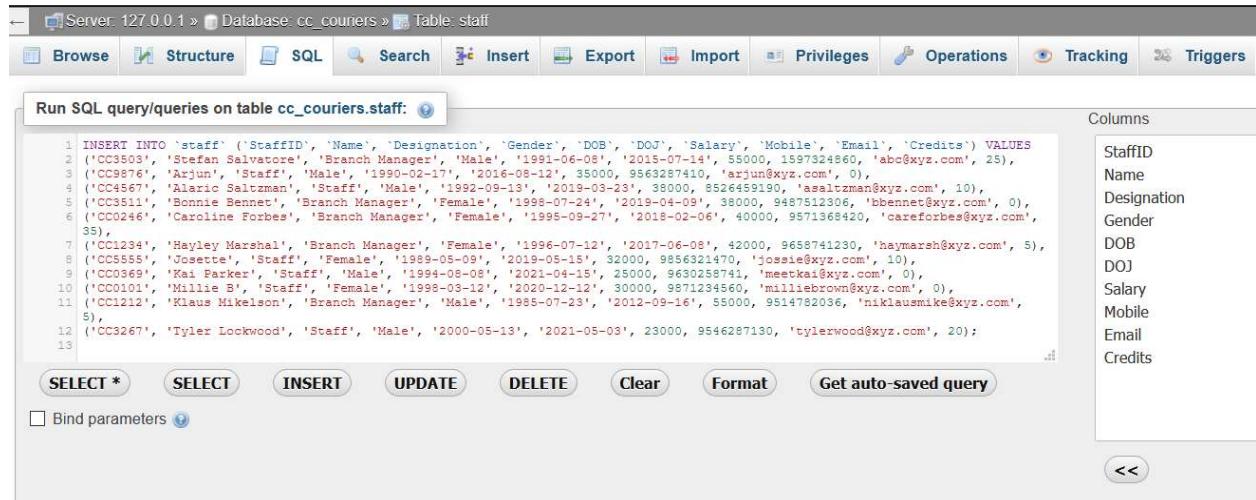


The screenshot shows the phpMyAdmin interface. The left sidebar shows the database structure with tables like cc\_couriers, events, branches, credentials, feedback, parcel, pricing, staff, status, views, frozen\_bottle, information\_schema, misc, music, mydb, mysql, performance\_schema, phpmyadmin, and roster. The main area shows the 'credentials' table with the following data:

StaffID	Pwd
CC0101	0101
CC0246	0246
CC0369	0369
CC1212	1212
CC1234	1234
CC3267	3267
CC3503	14-07
CC3511	3511
CC4567	4567
CC5555	5555
CC9876	9876

Fig 8.2 : Data stored in credentials table

2. **Staff relation** : Stores various details regarding the staff such as their id, name, salary, contact details, etc.



The screenshot shows the phpMyAdmin interface for a database named 'cc\_couriers'. The current table is 'staff'. The SQL query entered is:

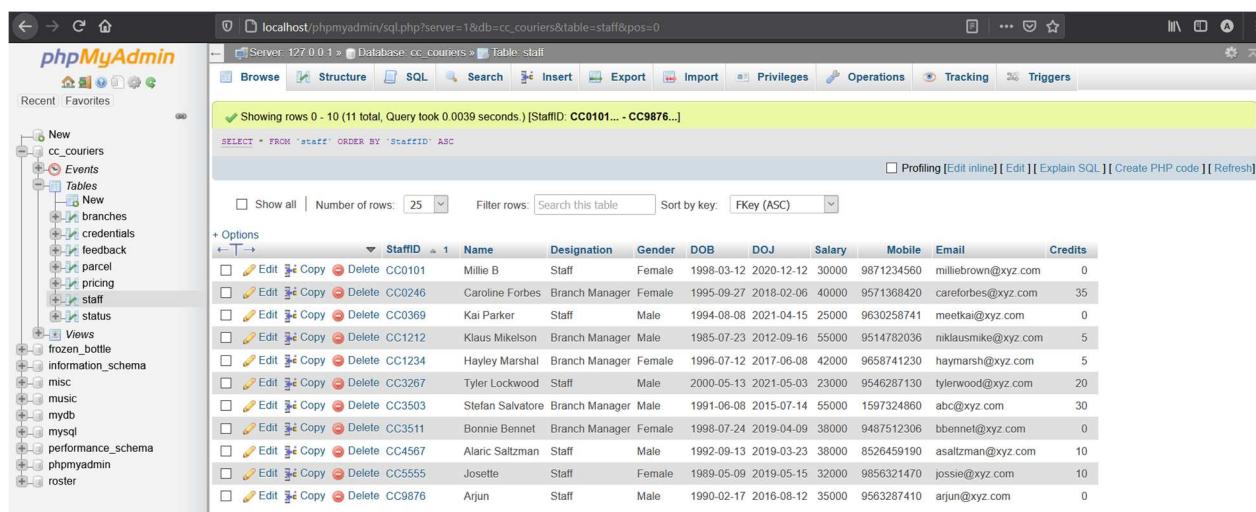
```

1 INSERT INTO `staff` ('StaffID', 'Name', 'Designation', 'Gender', 'DOB', 'DOJ', 'Salary', 'Mobile', 'Email', 'Credits') VALUES
2 ('CC3503', 'Stefan Salvatore', 'Branch Manager', 'Male', '1991-06-08', '2015-07-14', 55000, 1597324860, 'abc@xyz.com', 25),
3 ('CC9876', 'Arjun', 'Staff', 'Male', '1990-02-17', '2016-08-12', 35000, 9563287410, 'arjun@xyz.com', 0),
4 ('CC4567', 'Alaric Saltzman', 'Staff', 'Male', '1992-09-13', '2019-03-23', 38000, 8526459190, 'asaltzman@xyz.com', 10),
5 ('CC3511', 'Bonnie Bennet', 'Branch Manager', 'Female', '1998-07-24', '2019-04-09', 38000, 9487512306, 'bbennet@xyz.com', 0),
6 ('CC0246', 'Caroline Forbes', 'Branch Manager', 'Female', '1995-08-27', '2018-02-06', 40000, 9571368420, 'careforbes@xyz.com', 35),
7 ('CC1234', 'Hayley Marshal', 'Branch Manager', 'Female', '1996-07-12', '2017-06-08', 42000, 9658741230, 'haymarsh@xyz.com', 5),
8 ('CC5555', 'Josette', 'Staff', 'Female', '1989-05-09', '2019-05-15', 32000, 9856321470, 'jossie@xyz.com', 10),
9 ('CC0369', 'Kai Parker', 'Staff', 'Male', '1994-08-08', '2021-04-15', 25000, 9630258741, 'meetkai@xyz.com', 0),
10 ('CC0101', 'Millie B', 'Staff', 'Female', '1998-03-12', '2020-12-12', 30000, 9871234560, 'milliebrown@xyz.com', 0),
11 ('CC1212', 'Klaus Mikelson', 'Branch Manager', 'Male', '1985-07-23', '2012-09-16', 55000, 9514782036, 'niklausmike@xyz.com', 5),
12 ('CC3267', 'Tyler Lockwood', 'Staff', 'Male', '2000-05-13', '2021-05-03', 23000, 9546287130, 'tylerwood@xyz.com', 20);
13

```

The results pane on the right shows the columns: StaffID, Name, Designation, Gender, DOB, DOJ, Salary, Mobile, Email, and Credits. The data from the query is listed below.

Fig 8.3 : SQL query to insert data into staff table



The screenshot shows the phpMyAdmin interface for the 'staff' table. The data is displayed in a grid format:

	StaffID	Name	Designation	Gender	DOB	DOJ	Salary	Mobile	Email	Credits
<input type="checkbox"/>	CC0101	Millie B	Staff	Female	1998-03-12	2020-12-12	30000	9871234560	milliebrown@xyz.com	0
<input type="checkbox"/>	CC0246	Caroline Forbes	Branch Manager	Female	1995-09-13	2018-02-06	40000	9571368420	careforbes@xyz.com	35
<input type="checkbox"/>	CC0369	Kai Parker	Staff	Male	1994-08-08	2021-04-15	25000	9630258741	meetkai@xyz.com	0
<input type="checkbox"/>	CC1212	Klaus Mikelson	Branch Manager	Male	1985-07-23	2012-09-16	55000	9514782036	niklausmike@xyz.com	5
<input type="checkbox"/>	CC3267	Tyler Lockwood	Staff	Male	2000-05-13	2021-05-03	23000	9546287130	tylerwood@xyz.com	20
<input type="checkbox"/>	CC3503	Stefan Salvatore	Branch Manager	Male	1991-06-08	2015-07-14	55000	1597324860	abc@xyz.com	30
<input type="checkbox"/>	CC3511	Bonnie Bennet	Branch Manager	Female	1998-07-24	2019-04-09	38000	9487512306	bbennet@xyz.com	0
<input type="checkbox"/>	CC4567	Alaric Saltzman	Staff	Male	1992-09-13	2019-03-23	38000	8526459190	asaltzman@xyz.com	10
<input type="checkbox"/>	CC5555	Josette	Staff	Female	1989-05-09	2019-05-15	32000	9856321470	jossie@xyz.com	10
<input type="checkbox"/>	CC9876	Arjun	Staff	Male	1990-02-17	2016-08-12	35000	9563287410	arjun@xyz.com	0

Fig 8.4 : Data stored in staff table

3. **Branches relation** : Stores the details of the various branch offices such as branch id, address, contact, email along with the manager id of that branch.

The screenshot shows the MySQL Workbench interface with the following details:

- Server:** 127.0.0.1
- Database:** cc\_couriers
- Table:** branches
- Toolbar:** Browse, Structure, SQL, Search, Insert, Export, Import, Privileges, Operations, Tracking, Triggers
- Query Editor:** Run SQL query/queries on table cc\_couriers.branches:  
1 INSERT INTO `branches` (`Branch\_id`, `Address`, `Contact`, `Email`, `Manager\_id`) VALUES  
2 (1, '11, St.Joseph Church Campus, Dindigul - 624001', 4512424892, 'cccourierdgl@xyz.com', 'CC3511'),  
3 (2, 'B211, 11th Avenue, Ashok Nagar, Chennai - 600083', 443147782, 'cccourierschennai@xyz.com', 'CC0246'),  
4 (3, '5E, Municipal Office Street, Tiruppur - 641604', 421495625, 'cccourierstiruppur@xyz.com', 'CC1234'),  
5 (4, '7th Arun complex, Brough Road, Erode - 638001', 4245276485, 'cccourierserode@xyz.com', 'CC1212'),  
6 (5, '207, Abirami Nagar, Mettupalayam Road, Rounadapalayam, Coimbatore - 641030', 3563943598, 'cccourierscbe@xyz.com',  
7 'CC3503')  
7 |
- Columns:** Branch\_id, Address, Contact, Email, Manager\_id
- Buttons:** SELECT\*, SELECT, INSERT, UPDATE, DELETE, Clear, Format, Get auto-saved query
- Checkboxes:** Bind parameters

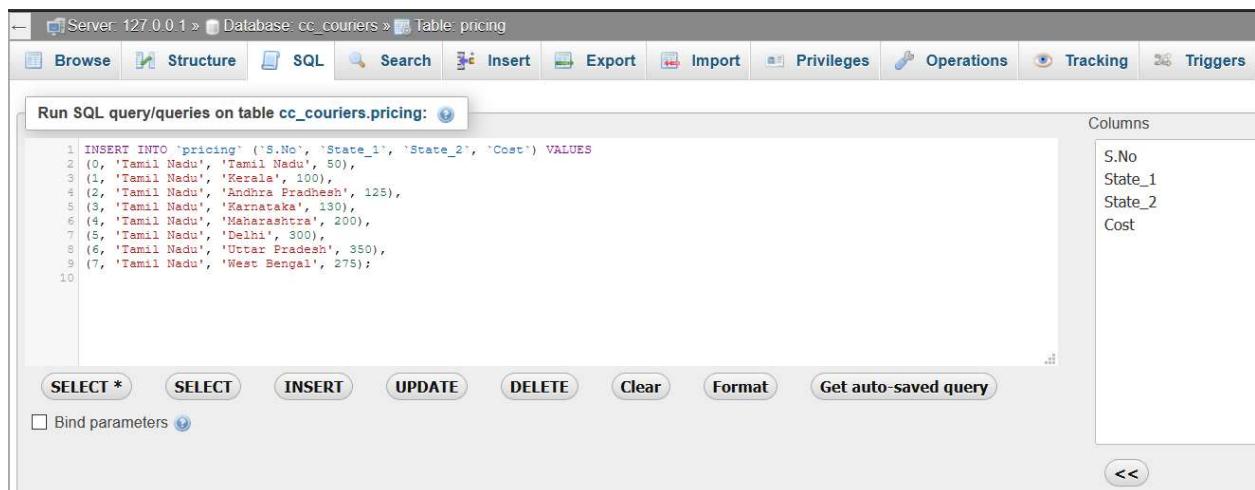
Fig 8.5 : SQL query to insert data into branches table

The screenshot shows the phpMyAdmin interface for the 'cc\_couriers' database. The left sidebar lists databases, tables, and other schema objects. The main area displays the 'branches' table with 5 rows of data. Each row includes edit, copy, and delete options. The table has columns for Branch\_id, Address, Contact, Email, and Manager\_id.

	Branch_id	Address	Contact	Email	Manager_id
<input type="checkbox"/> <a href="#">Edit</a> <a href="#">Copy</a> <a href="#">Delete</a>	1	11, St.Joseph Church Campus, Dindigul - 624001	4512424892	cccourierdgl@xyz.com	CC3511
<input type="checkbox"/> <a href="#">Edit</a> <a href="#">Copy</a> <a href="#">Delete</a>	2	21, 11th Avenue, Ashok Nagar, Chennai - 600083	4443147782	cccourierschennai@xyz.com	CC0246
<input type="checkbox"/> <a href="#">Edit</a> <a href="#">Copy</a> <a href="#">Delete</a>	3	5E, Municipal Office Street, Tirupur - 641604	4214956251	cccourierstirupur@xyz.com	CC1234
<input type="checkbox"/> <a href="#">Edit</a> <a href="#">Copy</a> <a href="#">Delete</a>	4	72, Arun complex, Brough Road, Erode - 638001	4245276485	cccouriererode@xyz.com	CC1212
<input type="checkbox"/> <a href="#">Edit</a> <a href="#">Copy</a> <a href="#">Delete</a>	5	207, Abirami Nagar, Mettupalayam Road, Koundapalayam - 641601	3563343593	cccourierscbe@xyz.com	CC3503

Fig 8.6 : Data stored in branches table

4. **Pricing relation** : Stores the details of delivery charge per kg (as Cost) of the transportation between the corresponding states. Also the states present in this relation depicts the availability of the service in various states mentioned in it.



The screenshot shows the MySQL Workbench interface. The title bar indicates the connection is to 'Server: 127.0.0.1' and the database is 'cc\_couriers'. The current table is 'pricing'. The SQL tab contains the following SQL code:

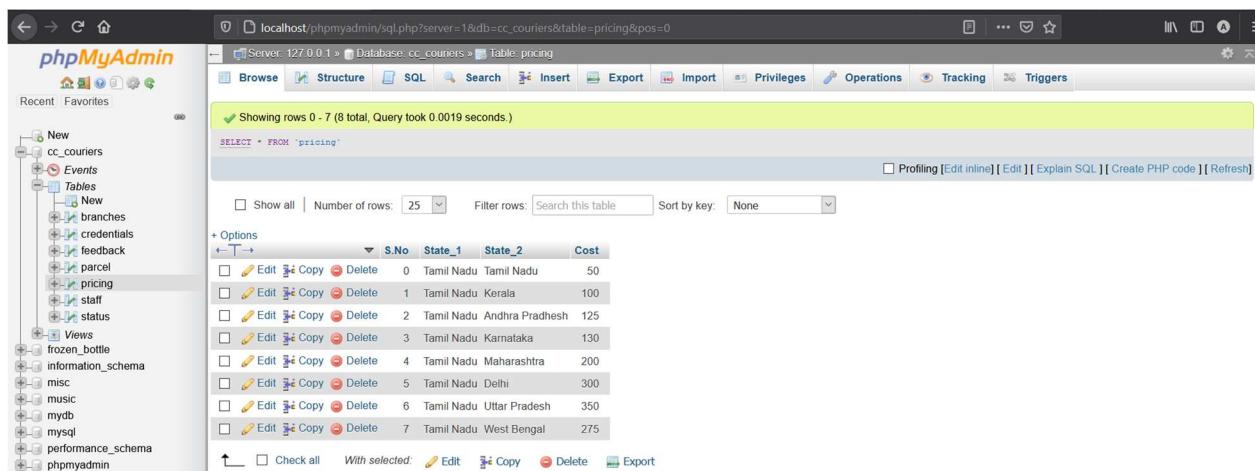
```

1 INSERT INTO `pricing` (`S.No`, `State_1`, `State_2`, `Cost`) VALUES
2 (0, 'Tamil Nadu', 'Tamil Nadu', 50),
3 (1, 'Tamil Nadu', 'Kerala', 100),
4 (2, 'Tamil Nadu', 'Andhra Pradesh', 125),
5 (3, 'Tamil Nadu', 'Karnataka', 130),
6 (4, 'Tamil Nadu', 'Maharashtra', 200),
7 (5, 'Tamil Nadu', 'Delhi', 300),
8 (6, 'Tamil Nadu', 'Uttar Pradesh', 350),
9 (7, 'Tamil Nadu', 'West Bengal', 275);
10

```

To the right of the SQL window, there is a 'Columns' panel listing the table's columns: S.No, State\_1, State\_2, and Cost. Below the SQL window are several buttons: SELECT \*, SELECT, INSERT, UPDATE, DELETE, Clear, Format, and Get auto-saved query. There is also a checkbox for Bind parameters.

Fig 8.7 : SQL query to insert data into pricing table



The screenshot shows the phpMyAdmin interface. The left sidebar shows the database structure for 'cc\_couriers', including tables like 'branches', 'credentials', 'feedback', 'parcel', 'pricing', 'staff', and 'status'. The main area shows the 'pricing' table with the following data:

	S.No	State_1	State_2	Cost
<input type="checkbox"/>	0	Tamil Nadu	Tamil Nadu	50
<input type="checkbox"/>	1	Tamil Nadu	Kerala	100
<input type="checkbox"/>	2	Tamil Nadu	Andhra Pradesh	125
<input type="checkbox"/>	3	Tamil Nadu	Karnataka	130
<input type="checkbox"/>	4	Tamil Nadu	Maharashtra	200
<input type="checkbox"/>	5	Tamil Nadu	Delhi	300
<input type="checkbox"/>	6	Tamil Nadu	Uttar Pradesh	350
<input type="checkbox"/>	7	Tamil Nadu	West Bengal	275

Fig 8.8 : Data stored in pricing table

## 6. USER INTERFACE DESIGN :

The user interface of this interactive web application is developed using HTML5, CSS3 and Bootstrap 4. HTML is the standard markup language for documents designed to be displayed in a web browser. CSS is the style sheet language used to style an HTML document and describes how HTML elements should be displayed. Bootstrap is a free and open-source CSS framework directed at responsive, mobile-first front-end web development. It contains CSS- and JavaScript-based design templates for typography, forms, buttons, navigation, and other interface components.



Fig 9.1 : Home page of the application with a carousal

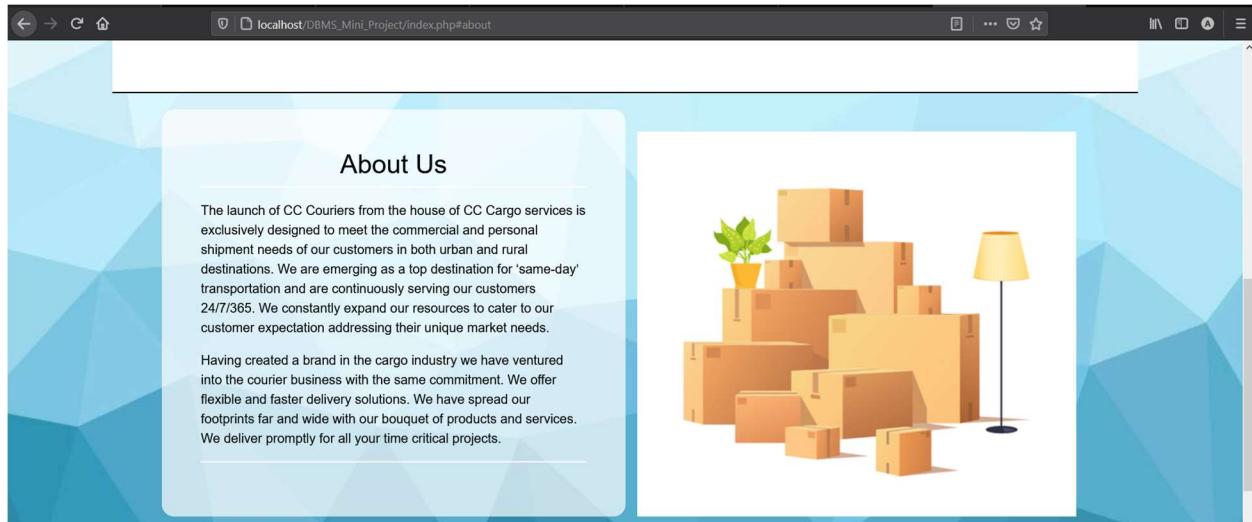


Fig 9.2 : The page featuring the legacy of CC couriers

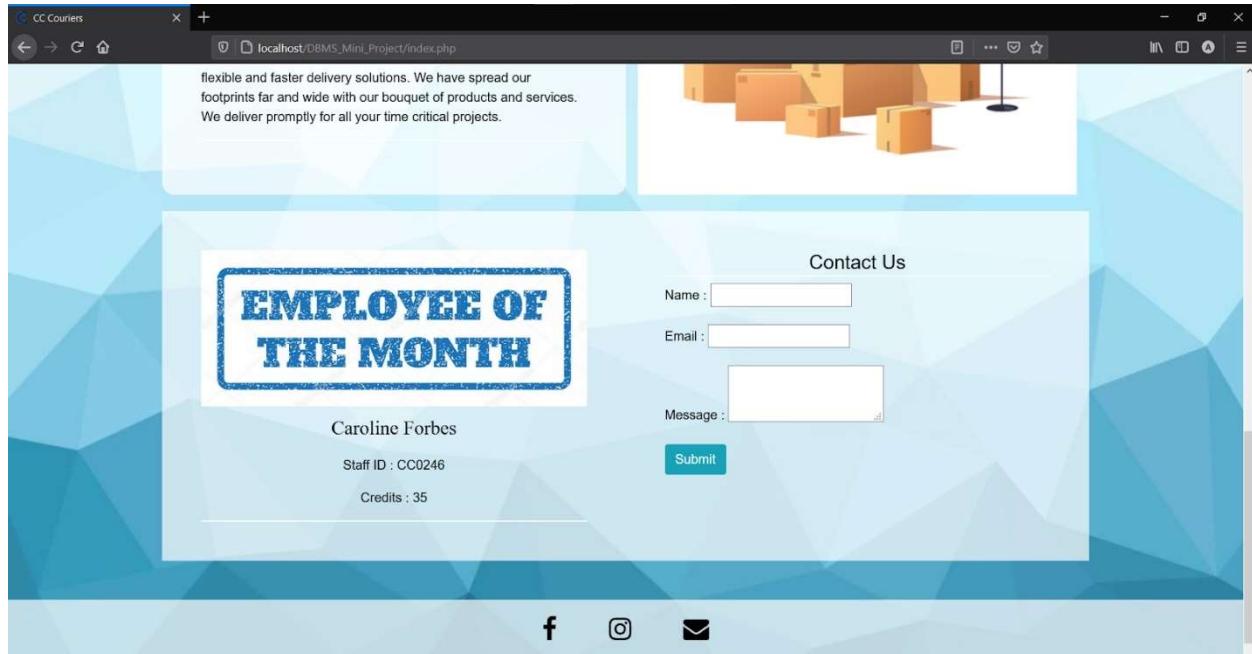


Fig 9.3 : Segment of the home page featuring the Employee of the Month and the Contact form

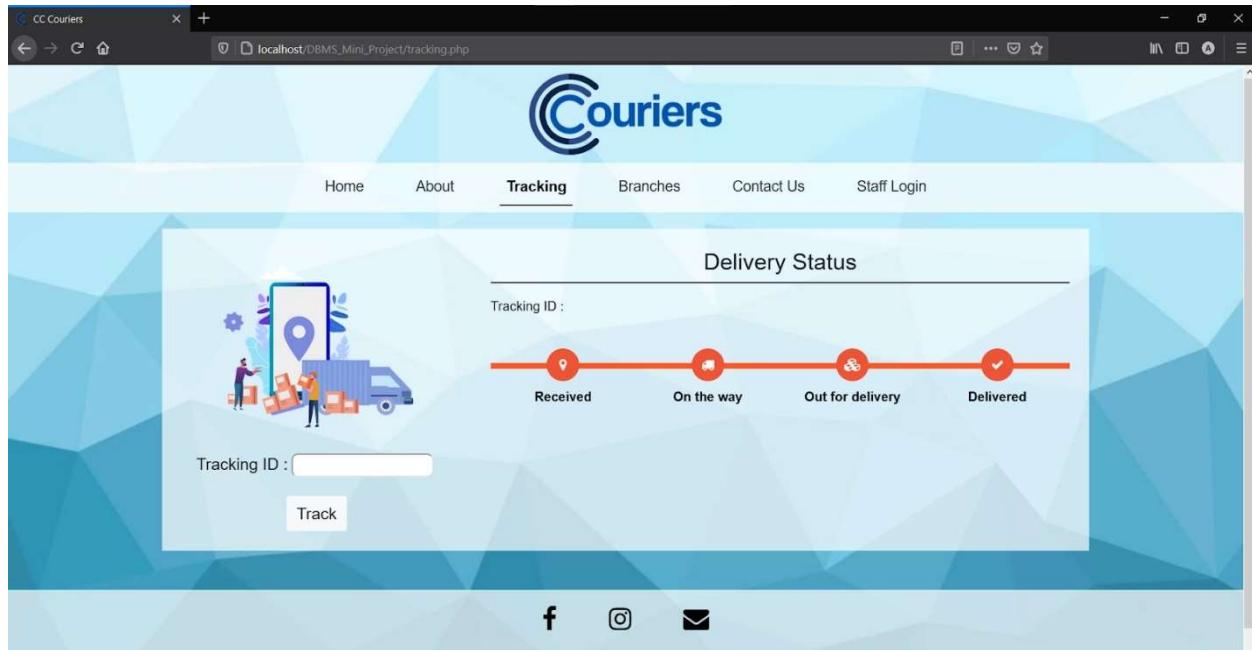


Fig 9.4 : Tracking page of the application

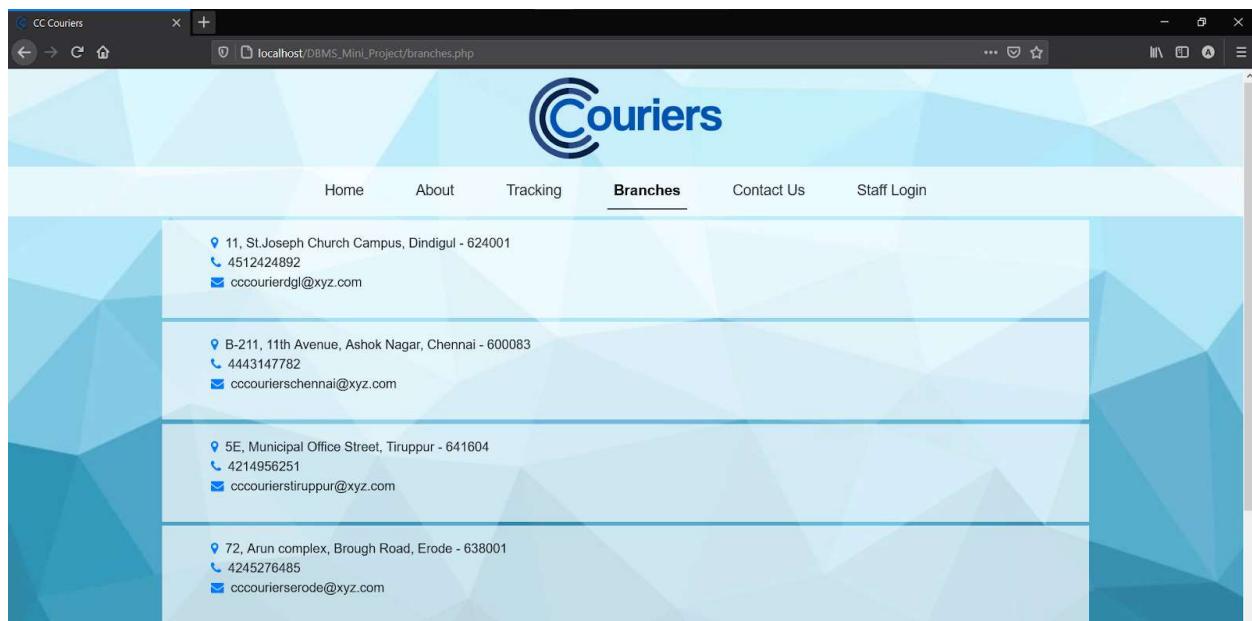


Fig 9.5 : Branches page displaying the details of the branch offices stored in branches relation



Fig 9.6 : Staff Login page

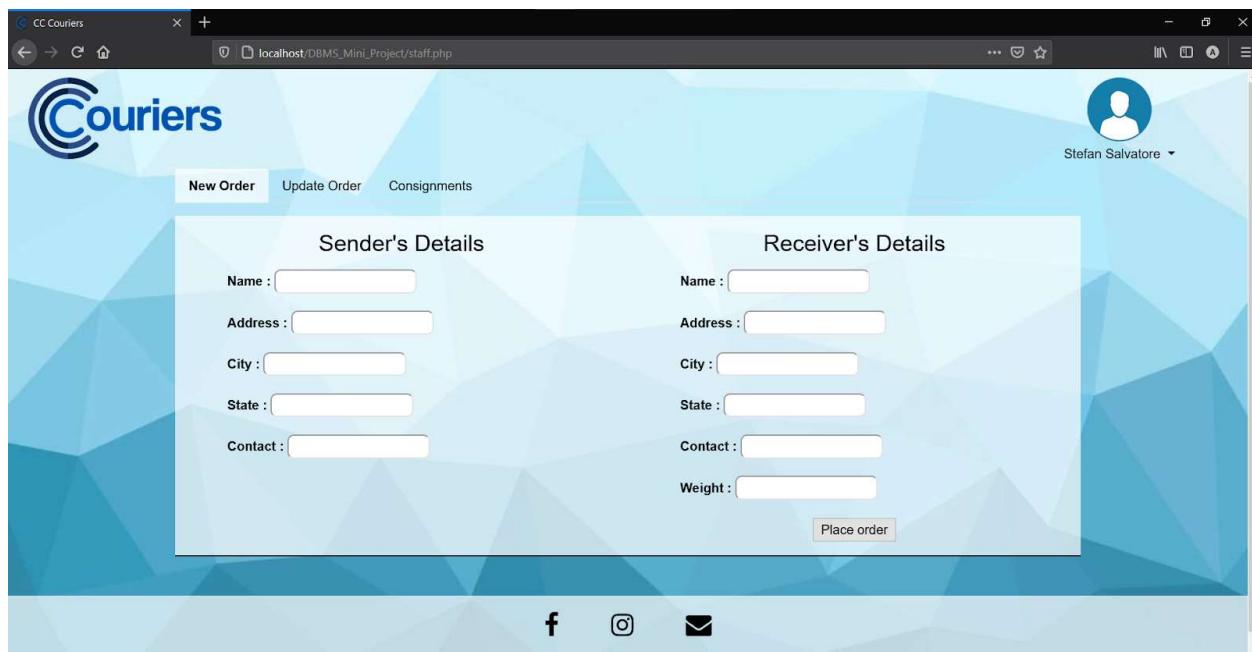


Fig 9.7 : Placing a new order tab of the staff page

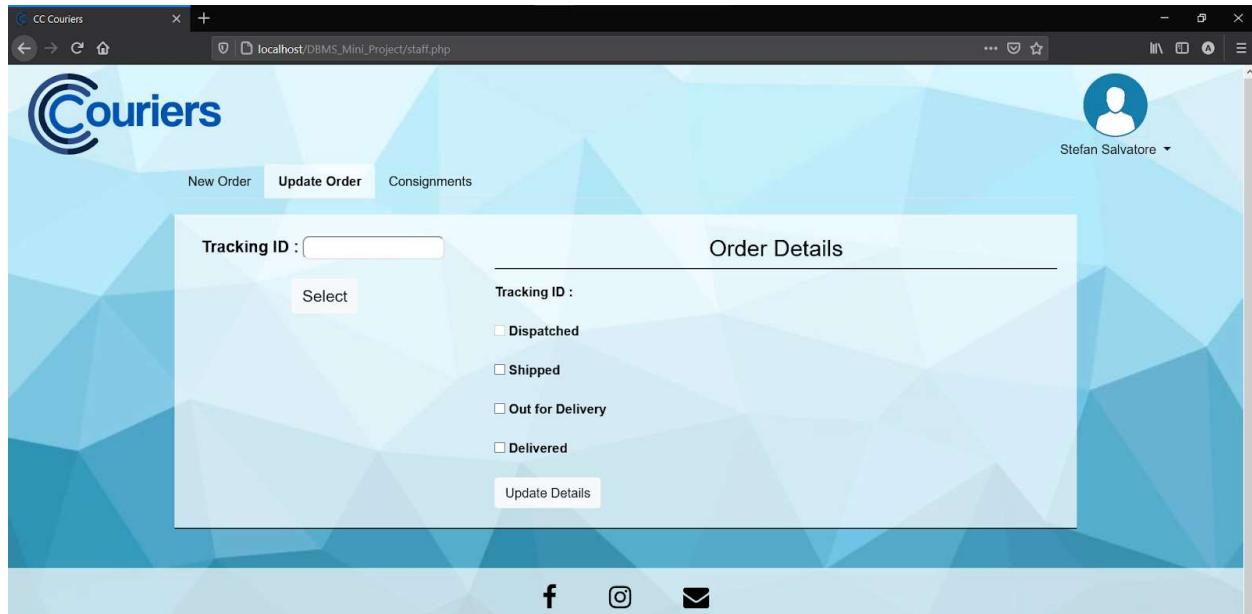


Fig 9.8 : Updating the existing parcel details tab of staff page

TrackingID	StaffID	Sender	Receiver	Weight	Price	Dispatched	Shipped	Out for delivery	Delivered
Arrived	Delivered								
10807	CC0246	Lizzie, 5, Salvatore Boarding House, Coimbatore, Tamil Nadu - 9487512036	Jossie, 8, French quarters, Agra, Uttar Pradesh - 941212360	2.75	962.50	2021-05-06 16:04:38	2021-05-09 21:54:42		
10809	CC0246	Lizzie, 5, Salvatore Boarding House, Coimbatore, Tamil Nadu - 9487512036	Jossie, 8, French quarters, Agra, Uttar Pradesh - 941212360	2.75	962.50	2021-05-06 16:33:57			
10812	CC0246	Karthick, 1, TRN Gardens, Udumalpet, Tamil Nadu - 9417417410	Dhamini, 7/8, Arun excelo, Avadi, Chennai, Tamil Nadu - 9412365870	2.00	100.00	2021-05-06 19:00:20	2021-05-06 19:05:31		

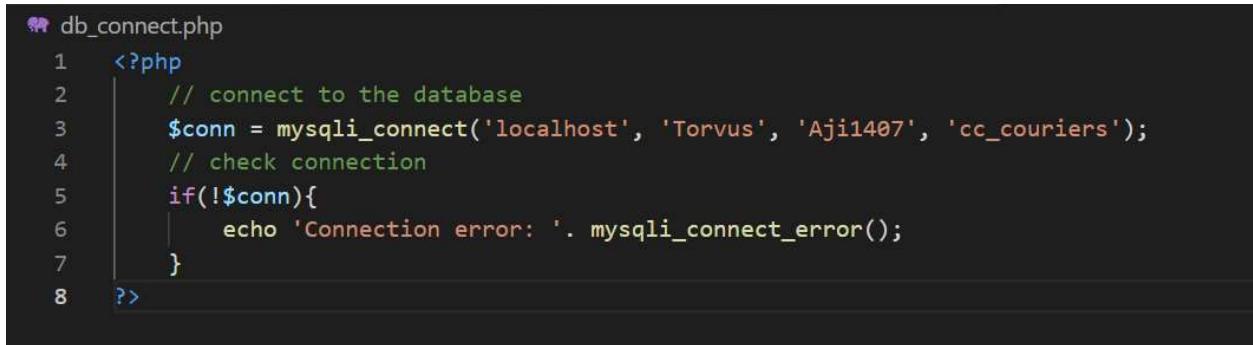
Fig 9.9 : Page displaying the consignments in transit which are the contents of the arrived view

The screenshot shows a web application titled "CC Couriers" with a blue header and a white background. At the top right is a user profile icon for "Stefan Salvatore". Below the header, there are three tabs: "New Order", "Update Order", and "Consignments". The "Consignments" tab is active, showing two sub-tabs: "Arrived" and "Delivered", with "Delivered" selected. The main content area contains a table with the following data:

TrackingID	StaffID	Sender	Receiver	Weight	Price	Dispatched	Shipped	Out for delivery	Delivered
10808	CC0246	Lizzie, 5, Salvatore Boarding House, Coimbatore, Tamil Nadu - 9487512036	Jossie, 8, French quarters, Agra, Uttar Pradesh - 941212360	2.75	962.50	2021-05-06 16:31:43	2021-05-06 18:29:19	2021-05-29 15:07:28	2021-05-29 15:10:31
10813	CC0246	Karthick, 1, TRN Gardens, Udumalpet, Tamil Nadu - 9417417410	Dhamini, 7/8, Arun excelo, Avadi, Chennai, Tamil Nadu - 9412365870	2.00	100.00	2021-05-06 19:05:01	2021-05-06 19:13:22	2021-05-06 19:17:41	2021-05-06 19:18:05
10818	CC3267	Jai, 6, Angel Residency, Trichy, Tamil Nadu - 9178542036	Arya, 75, Main cross street, Alapuzha, Kerala - 9514236870	2.00	200.00	2021-05-10 23:11:38	2021-05-11 00:35:47	2021-05-11 00:36:25	2021-05-11 00:36:37

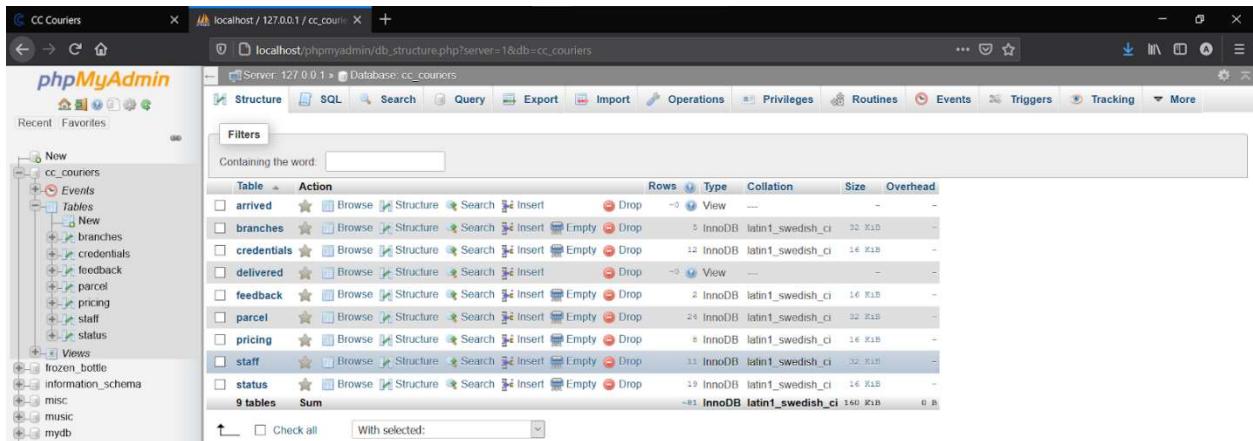
Fig 9.10 : Page displaying the delivered consignments which are the contents of the delivered view

## 7. CONNECTING TO DATABASE :



```
db_connect.php
1 <?php
2     // connect to the database
3     $conn = mysqli_connect('localhost', 'Torvus', 'Aji1407', 'cc_couriers');
4     // check connection
5     if(!$conn){
6         echo 'Connection error: '. mysqli_connect_error();
7     }
8 ?>
```

Fig 10.1 : Code for connecting the database



The screenshot shows the phpMyAdmin interface for the 'cc\_couriers' database. The left sidebar lists tables such as arrived, branches, credentials, delivered, feedback, parcel, pricing, staff, and status. The main area displays the structure of the 'arrived' table, which has 5 rows and 10 columns: ID, tracking\_id, tracking\_no, date, time, address, city, state, zip, and status. Other tables like branches, credentials, delivered, feedback, parcel, pricing, staff, and status are also listed with their respective row counts.

Fig 10.2 : CC Couriers Database

## 8. STAFF LOGIN :

```
login.php
1  <?php
2      include("db_connect.php");
3
4      $id = $pwd = '';
5      $errors = array('id' => '', 'pwd' => '', 'login' => '');
6
7      if(isset($_POST['submit'])){
8          if(empty($_POST['id'])){
9              $errors['id'] = "*Required";
10         }else{
11             $id = $_POST['id'];
12         }
13         if(empty($_POST["pwd"])){
14             $errors['pwd'] = "*Required";
15         }else{
16             $pwd = $_POST['pwd'];
17         }
18         if(array_filter($errors)){
19             //echo errors
20         }else{
21             $id = mysqli_real_escape_string($conn, $id);
22             $pwd = mysqli_real_escape_string($conn, $pwd);
23
24             $sql = "SELECT * FROM credentials WHERE StaffID='$id' AND Pwd='$pwd'";
25             $result = mysqli_query($conn, $sql);
26             if(mysqli_num_rows($result) > 0){
27                 $user = mysqli_fetch_assoc($result);
28                 session_start();
29                 $_SESSION['id'] = $user['StaffID'];
30                 header("Location: staff.php");
31             }else{
32                 $sql = "SELECT * FROM credentials WHERE StaffID='id'";
33                 $result = mysqli_query($conn, $sql);
34                 if(mysqli_num_rows($result) == 0){
35                     $errors['login'] = 'Enter valid Staff ID';
36                 }else{
37                     $user = mysqli_fetch_assoc($result);
38                     if($pwd != $user['Pwd']){
39                         $errors['login'] = 'Incorrect Password';
40                     }
41                 }
42             }
43         }
44     }
```

Fig 11.1 : PHP code snippet for validating the login credentials and redirecting to the staff page on successful login. It stores the credentials in the session on successful login.

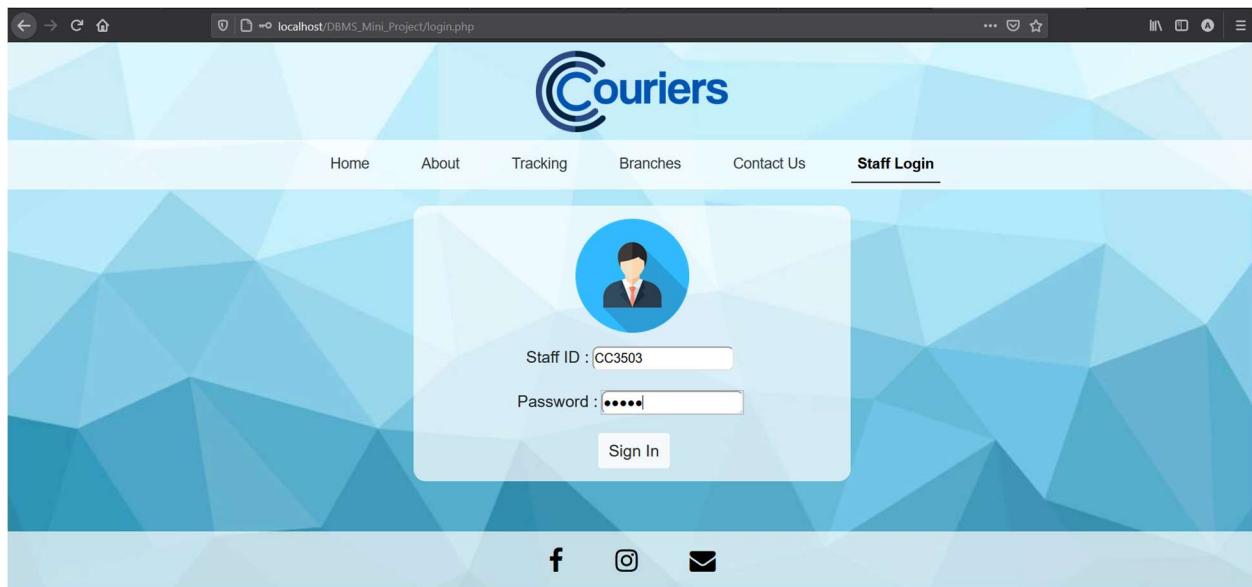


Fig 11.2 : Staff login page with login credentials as per the credentials table



Fig 11.3 : Staff login page prompting its mandatory to enter the corresponding details for successful login



Fig 11.4 : Staff login page displaying the prompt – Incorrect Password



Fig 11.5 : Staff login page displaying the prompt – Enter valid Staff Id – when the entered staff id is not present in the credentials table

## 9. STAFF DETAILS :

```
<?php
    include("db_connect.php");
    session_start();
    $id = $_SESSION['id'];
    $sql = "SELECT * FROM staff WHERE StaffID='$id'";
    $result = mysqli_query($conn, $sql);
    $staff = mysqli_fetch_assoc($result);
?
>
```

Fig 12.1 : PHP code snippet that executes the corresponding SQL query to fetch the details of the logged in staff

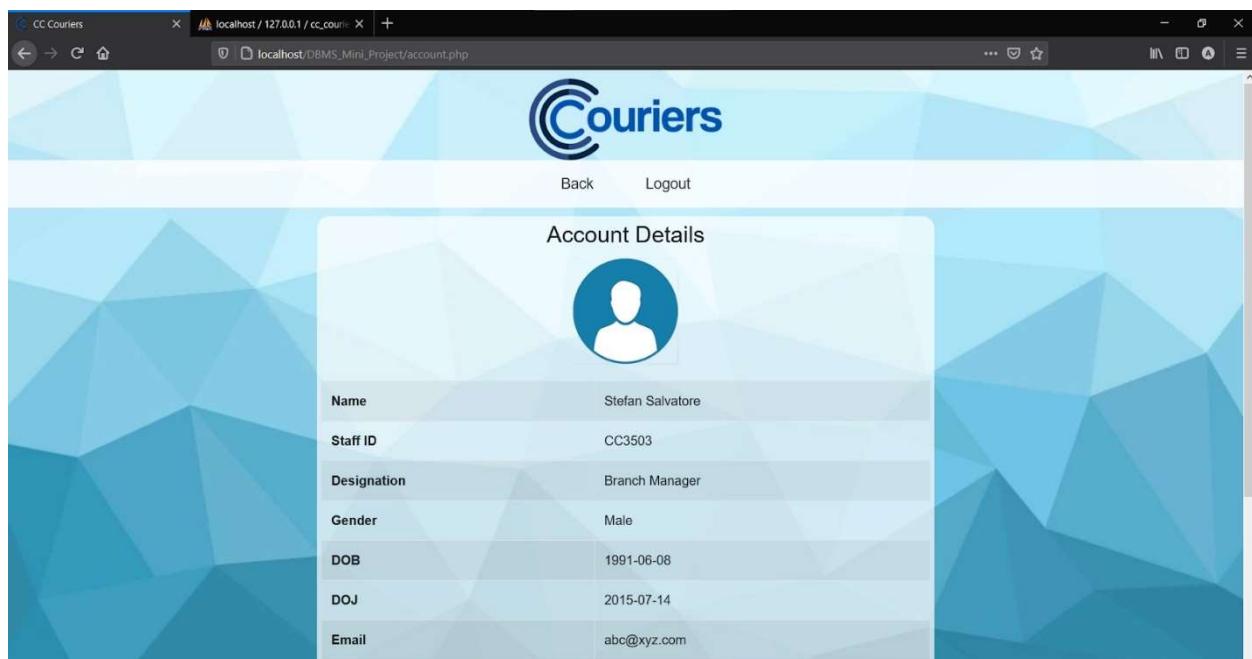


Fig 12.2 : Page displaying the fetched staff details from the staff table

## 10. LOGOUT :

```
logout.php
1 <?php
2     session_start();
3     session_destroy();
4     header("Location: login.php");
5 ?>
```

Fig 13.1 : PHP code snippet to logout the current staff by destroying the session storing the credentials of the logged in staff id. On successful logout, it redirects to the login page.



Fig 13.2 : The logout option is available in the dropdown near the logged in staff which redirects to the login page.

## 11. PLACING A NEW PARCEL :

```
72     if(array_filter($errors)){
73         //echo errors
74     }else{
75         $price = 0;
76         $sql = "SELECT * FROM pricing WHERE State_1 = '$sstate' AND State_2 = '$rstate'";
77         $result = mysqli_query($conn, $sql);
78         if(mysqli_num_rows($result) > 0){
79             $pricing = mysqli_fetch_assoc($result);
80             $price = $pricing['Cost'] * $wgt;
81
82             $sql = "INSERT INTO parcel (StaffID, S_Name, S_Add, S_City, S_State, S_Contact, R_Name, R_Add, R_City, R_State, R_Contact,
83             if(mysqli_query($conn, $sql)){
84                 $tid = mysqli_insert_id($conn);
85                 $_SESSION['tid'] = $tid;
86                 header("Location: receipt.php");
87             }else{
88                 echo "Error : " . mysqli_error($conn);
89             }
90         }else{
91             echo '<script type="text/javascript">';
92             echo 'setTimeout(function () { swal("Service Not Available", "CC Couriers will reach your place soon !!", "info");',
93             echo '}, 1000);</script>';
94         }
95     }
}
```

Fig 14.1 : PHP code snippet which stores the data collected from the new order form into the parcel table ( and to the status table by the trigger placeParcel ) only if the delivery details provided can be serviced. Here the delivery charge is calculated based on the values stored in the pricing table. It redirects the page to receipt page along with its unique tracking id. It also alerts the staff when the service is not available for the address details provided.

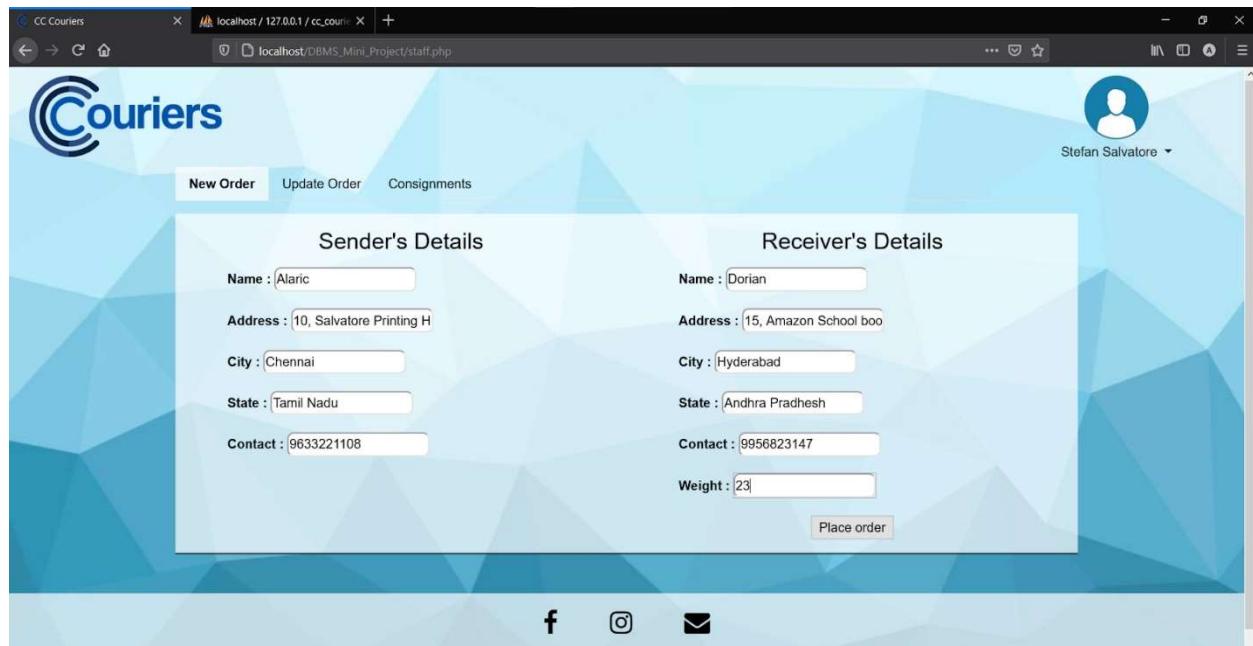


Fig 14.2 : Page displaying the form with details which is stored into the parcel and status table when the place order button is hit.

**Table: parcel**

	TrackingID	StaffID	S_Name	S_Address	S_City	S_State	S_Contact	R_Name	R_Address	R_City	R_State	R_Contact	Weight_Kg	Price	Dispatched_Time
Copy	10816	CC3267	Rishi	78, Talabadi	Coimbatore	Tamil Nadu	9852630147	Celine	34, Cheta gardens	Palakkad	Kerala	9514273680	2.00	200.00	2021-05-10 22:54:23
Copy	10817	CC3267	Jai	6, Angel Residency	Trichy	Tamil Nadu	9178542036	Arya	75, Main cross street	Alappuzha	Kerala	9514236870	2.00	200.00	2021-05-10 22:56:39
Copy	10818	CC3267	Jai	6, Angel Residency	Trichy	Tamil Nadu	9178542036	Arya	75, Main cross street	Alappuzha	Kerala	9514236870	2.00	200.00	2021-05-10 23:11:38
Copy	10819	CC4567	Karthick	1, TRN Gardens	Udumalpet	Tamil Nadu	9417417410	Dhamini	75, Main cross street	Chennai	Tamil Nadu	9412365870	1.25	62.50	2021-05-11 00:30:37
Copy	10820	CC4567	Eswar	12, MJ Gardens	Erode	Tamil Nadu	9922663300	Chanya	45, BB street	Theni	Tamil Nadu	9956823147	1.00	50.00	2021-05-11 00:45:00
Copy	10821	CC1212	Aathmika	1, TRN Gardens	Udumalpet	Tamil Nadu	9512034687	Sanjuktha	15, Eden palace	Theni	Tamil Nadu	9912365407	7.00	350.00	2021-05-13 19:42:10
Copy	10822	CC5555	Sudharshan	Greenpark layout	Udumalpet	Tamil Nadu	9485760123	Sharq	27, CB Street	Coimbatore	Tamil Nadu	9514200368	10.00	500.00	2021-05-17 10:52:00
Copy	10823	CC5555	Sudharshan	Greenpark layout	Udumalpet	Tamil Nadu	9485760123	Sharq	27, CB Street	Coimbatore	Tamil Nadu	9514200368	10.00	500.00	2021-05-17 10:53:18
Copy	10824	CC3503	Ajitesh M	1, TRN Gardens	Udumalpet	Tamil Nadu	9488393922	Aathmika M	120, Hostel, GTMC	Theni	Tamil Nadu	9488393911	2.50	125.00	2021-05-28 21:37:57
Copy	10825	CC3503	Alaric	10, Salvatore Printing House	Chennai	Tamil Nadu	9633221108	Dorian	15, Amazon School bookstore	Hyderabad	Andhra Pradesh	9956823147	23.00	2875.00	2021-05-29 15:15:53

Fig 14.3 : The details of the placed parcel along with its delivery charge (price), staff id and the dispatched time (which is the time when the parcel received by the source office = CURRENT\_TIMESTAMP) in the parcel table.

**Table: status**

	TrackingID	StaffID	Dispatched	Shipped	Out_for_delivery	Delivered
	10807	CC0246	2021-05-06 19:11:20	2021-05-09 21:54:42	NULL	NULL
	10808	CC0246	2021-05-06 19:11:20	2021-05-06 18:29:19	2021-05-29 15:07:28	2021-05-29 15:10:31
	10809	CC0246	2021-05-06 19:11:20	NULL	NULL	NULL
	10812	CC0246	2021-05-06 19:11:20	2021-05-06 19:05:31	NULL	NULL
	10813	CC0246	2021-05-06 19:11:20	2021-05-06 19:13:22	2021-05-06 19:17:41	2021-05-06 19:18:05
	10814	CC1234	2021-05-10 15:00:20	2021-05-29 15:08:05	NULL	NULL
	10815	CC3267	2021-05-10 22:47:00	2021-05-29 15:08:18	NULL	NULL
	10816	CC3267	2021-05-10 22:54:23	2021-05-29 15:08:27	NULL	NULL
	10817	CC3267	2021-05-10 22:56:39	2021-05-29 15:08:38	NULL	NULL
	10818	CC3267	2021-05-10 23:11:38	2021-05-11 00:35:47	2021-05-11 00:36:25	2021-05-11 00:36:37
	10819	CC4567	2021-05-11 00:30:37	NULL	NULL	NULL
	10820	CC4567	2021-05-11 00:45:00	NULL	NULL	NULL
	10821	CC1212	2021-05-13 19:42:10	2021-05-13 19:42:58	NULL	NULL
	10822	CC5555	2021-05-17 10:52:00	2021-05-29 15:09:51	NULL	NULL
	10823	CC5555	2021-05-17 10:53:18	2021-05-29 15:10:00	NULL	NULL
	10824	CC3503	2021-05-28 21:37:57	2021-05-29 15:10:12	NULL	NULL
	10825	CC3503	2021-05-29 15:15:53	NULL	NULL	NULL

Fig 14.4 : Result of the trigger – placeParcel – which stored the tracking id, staff id and dispatched time of the new parcel placed (in parcel table) into the status table. Also the trigger adds 5 credits to the staff for each parcel placed by them which is used to announce the Employee of the Month.

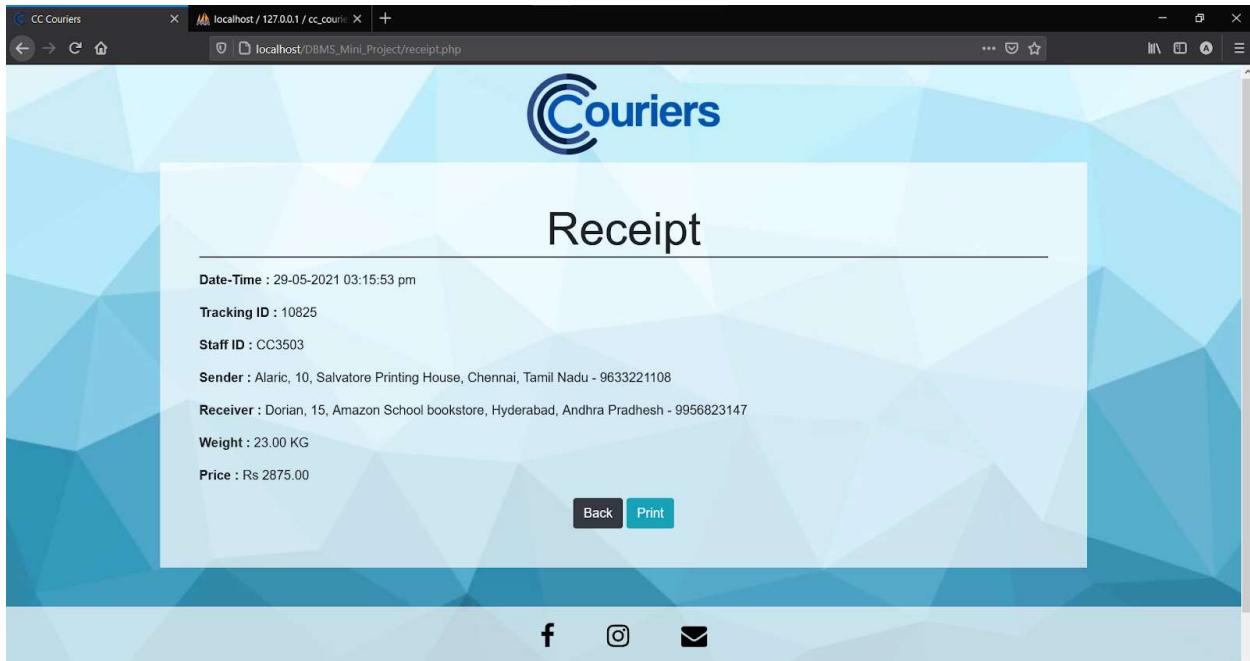


Fig 14.5 : Receipt page on successful placement of parcel display its details along with its unique tracking id and delivery charges calculated based on the values provided by the pricing table on. The back button takes us back to staff page and the print button prints the receipt out.

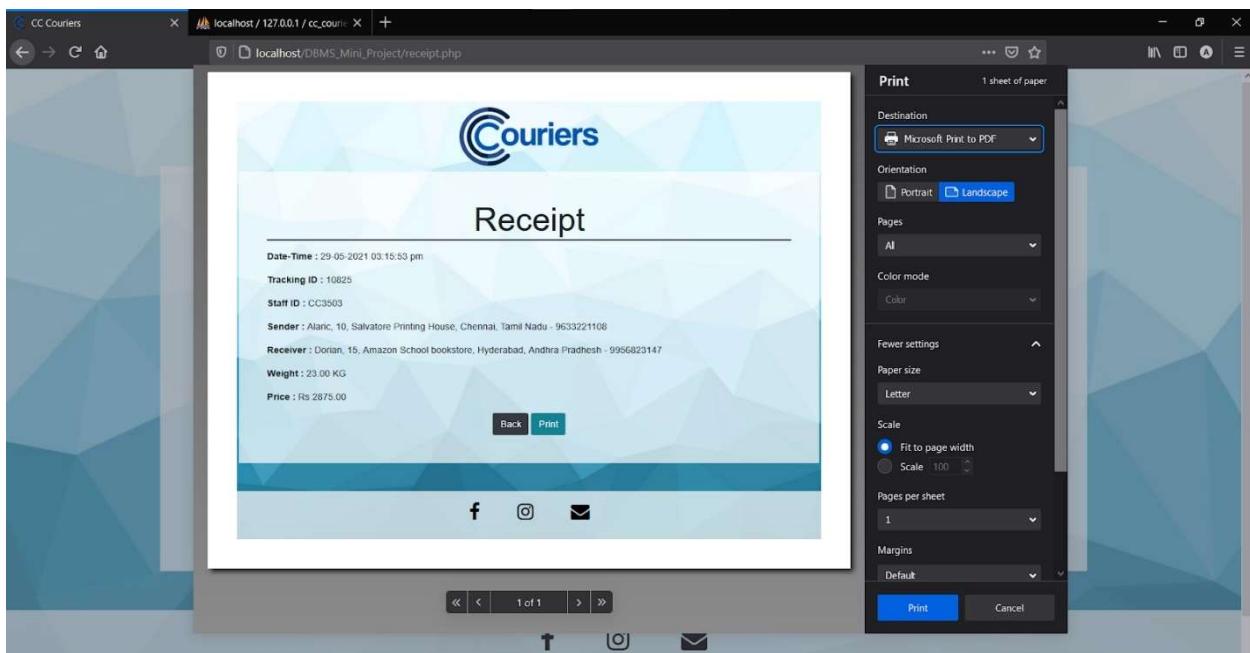


Fig 14.6 : The generated receipt is printed when the print button is hit.

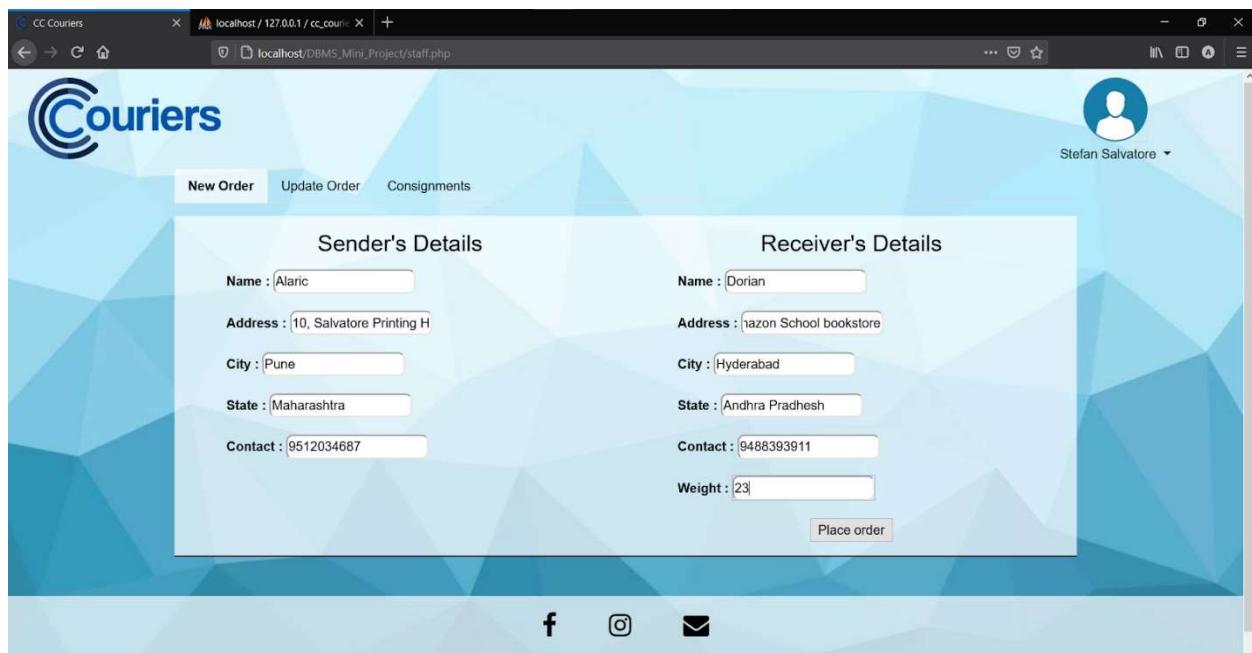


Fig 14.7 : Page displaying the form with details where the service is not available. These service details are extracted from the pricing table which depicts the states between which the service is available along with delivery charge/kg.

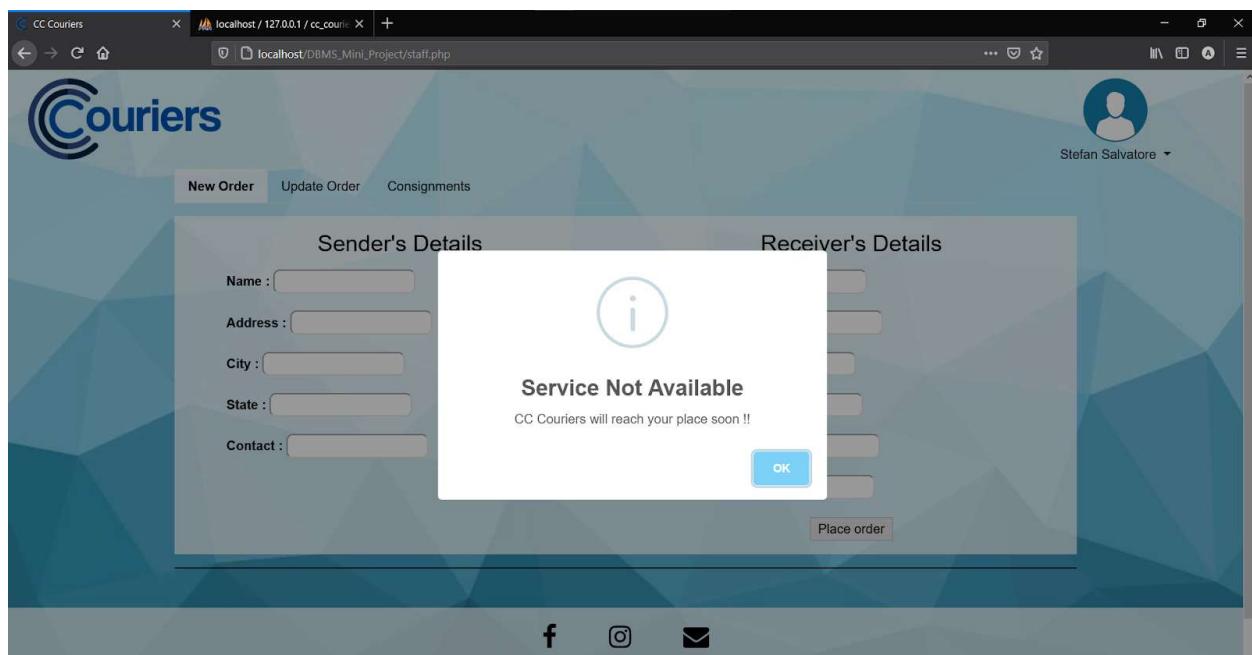


Fig 14.8 : Alert prompting that the delivery service is not available for the address details provided.

## 12. UPDATING DELIVERY STATUS OF EXISTING PARCEL :

```
97     if(isset($_POST['sel_order'])){
98         if(empty($_POST['inp_tid'])){
99             $errors['status'] = '*Required Field';
100        }else{
101            $inp_tid = $_POST['inp_tid'];
102        }
103        $sql = "SELECT * FROM status WHERE TrackingID = '$inp_tid'";
104        $result = mysqli_query($conn, $sql);
105        if(mysqli_num_rows($result)){
106            $del_status = mysqli_fetch_assoc($result);
107            $status['disp'] = $del_status['Dispatched'];
108            $status['ship'] = $del_status['Shipped'];
109            $status['out'] = $del_status['Out_for_delivery'];
110            $status['del'] = $del_status['Delivered'];
111            $inp_tid = $del_status['TrackingID'];
112            $_SESSION['up_tid'] = $inp_tid;
113            if(!is_null($status['del'])){
114                $disable_del = $disable_out = $disable_ship = "disabled";
115            }elseif(!is_null($status['out'])){
116                $disable_out = $disable_ship = "disabled";
117            }elseif(!is_null($status['ship'])){
118                $disable_ship = "disabled";
119            }
120            if(is_null($status['ship'])){
121                $disable_del = $disable_out = "disabled";
122            }elseif(is_null($status['out'])){
123                $disable_del = "disabled";
124            }
125        }else{
126            $errors['status'] = 'Enter a valid tracking ID';
127            echo "Error : " . mysqli_error();
128        }
129    }
130    if(isset($_POST['update'])){
131        $checked = $_POST['status_upd'];
132        //echo 'value : '.$checked;
133        $inp_tid = $_SESSION['up_tid'];
134        if($checked == 'delivered'){
135            $sql = "UPDATE status SET Delivered=CURRENT_TIMESTAMP WHERE TrackingID='$inp_tid' ";
136        }elseif($checked == 'out_for_delivery'){
137            $sql = "UPDATE status SET Out_for_delivery=CURRENT_TIMESTAMP WHERE TrackingID='$inp_tid' ";
138        }elseif($checked == 'shipped'){
139            $sql = "UPDATE status SET Shipped=CURRENT_TIMESTAMP WHERE TrackingID='$inp_tid' ";
140        }
141        if(mysqli_query($conn, $sql)){
142
143        }else{
144            echo "Error : ". mysqli_error($conn);
145        }
146    }
147 }
```

Fig 15.1 : PHP code snippet with corresponding SQL query for selecting the record of the entered tracking id from status table. The fetched results are displayed and it is updated to the subsequent delivery status with the current timestamp. Also this script ensures that status is updated in a hierarchical manner.

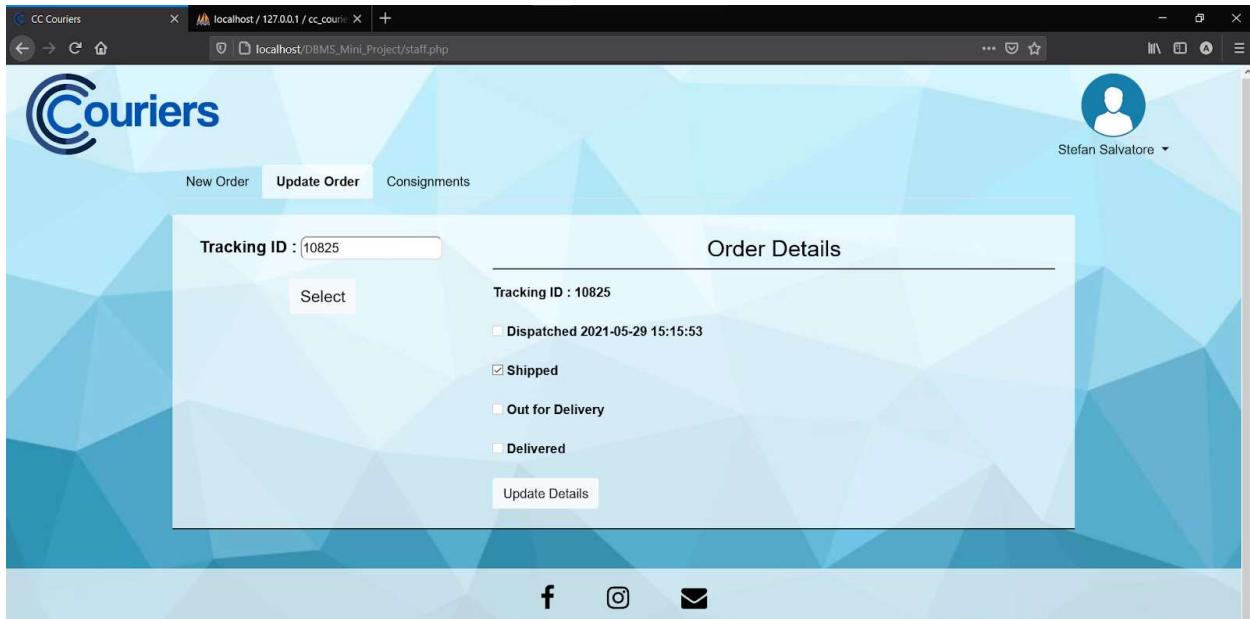


Fig 15.2 : The record of the entered tracking id is fetched from the status table is displayed and the subsequent status is checked for update. Also note that only one status track is enabled, the lower and higher levels are disabled to ensure the hierarchy of the delivery statuses.

The screenshot shows the 'status' table in phpMyAdmin. The table has columns: TrackingID, StaffID, Dispatched, Shipped, Out\_for\_delivery, and Delivered. There are 17 rows of data. The 'Shipped' column contains the current time for most entries, while others are NULL. The 'TrackingID' column lists values such as 108025, 10808, 10809, 10812, etc.

	TrackingID	StaffID	Dispatched	Shipped	Out_for_delivery	Delivered
	108025	CC0246	2021-05-06 19:11:20	2021-05-09 21:54:42	NULL	NULL
	10808	CC0246	2021-05-06 19:11:20	2021-05-06 18:29:19	2021-05-29 15:07:28	2021-05-29 15:10:31
	10809	CC0246	2021-05-06 19:11:20	NULL	NULL	NULL
	10812	CC0246	2021-05-06 19:11:20	2021-05-06 19:05:31	NULL	NULL
	10813	CC0246	2021-05-06 19:11:20	2021-05-06 19:13:22	2021-05-06 19:17:41	2021-05-06 19:18:05
	10814	CC1234	2021-05-10 15:00:29	2021-05-29 15:08:05	NULL	NULL
	10815	CC3267	2021-05-10 22:47:00	2021-05-29 15:08:18	NULL	NULL
	10816	CC3267	2021-05-10 22:54:23	2021-05-29 15:08:27	NULL	NULL
	10817	CC3267	2021-05-10 22:56:39	2021-05-29 15:08:38	NULL	NULL
	10818	CC3267	2021-05-10 23:11:38	2021-05-11 00:35:47	2021-05-11 00:36:25	2021-05-11 00:36:37
	10819	CC4567	2021-05-11 00:30:37	NULL	NULL	NULL
	10820	CC4567	2021-05-11 00:45:00	NULL	NULL	NULL
	10821	CC1212	2021-05-13 19:42:10	2021-05-13 19:42:58	NULL	NULL
	10822	CC5555	2021-05-17 10:52:00	2021-05-29 15:09:51	NULL	NULL
	10823	CC5555	2021-05-17 10:53:18	2021-05-29 15:10:00	NULL	NULL
	10824	CC3503	2021-05-28 21:37:57	2021-05-29 15:10:12	NULL	NULL
	10825	CC3503	2021-05-29 15:15:53	2021-05-29 15:21:16	NULL	NULL

Fig 15.3 : Note that the record of the tracking id 108025 is modified with Shipped = current time.

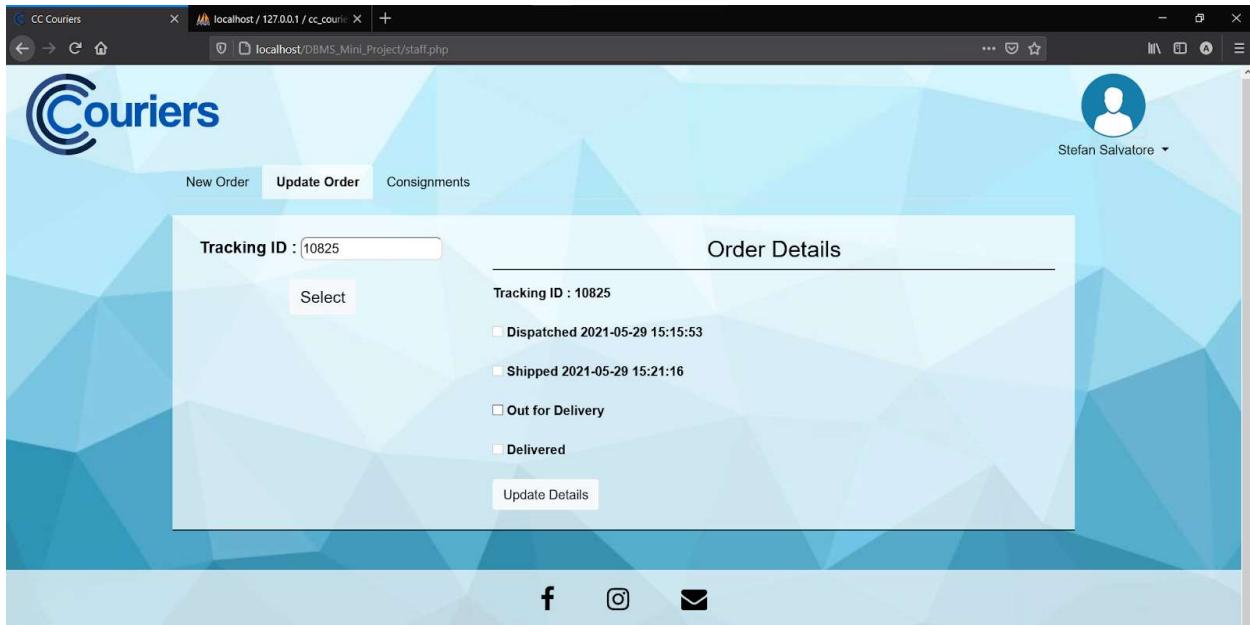


Fig 15.4 : The successful status update is reflected when the record of the same tracking id is fetched.

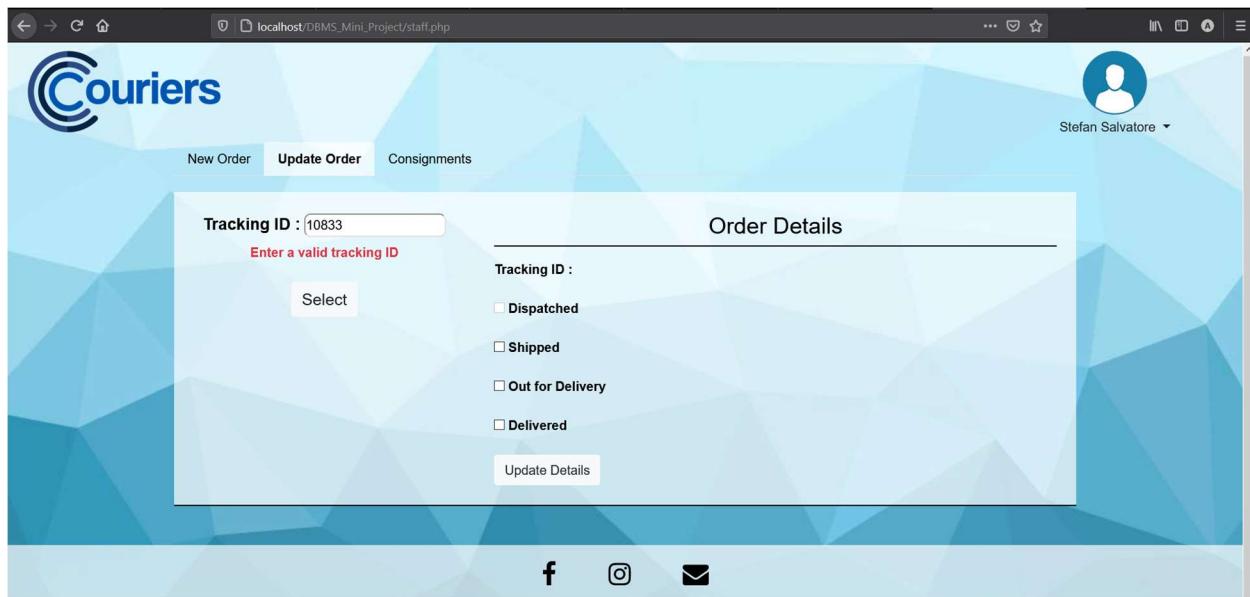


Fig 15.5 : When an invalid tracking id is entered

## 13. DELIVERY STATUS TRACKING :

```
tracking.php
1  <?php
2      include("db_connect.php");
3      $tid = '';
4      $error = '';
5      $status = array('Dispatched' => '', 'Shipped' => '', 'Out_for_delivery' => '', 'Delivered' => '', );
6      $hide = 'hidden';
7      session_start();
8      $trackid = '';
9      if(isset($_POST['track'])){
10          if(empty($_POST['tid'])){
11              $error = "*Required";
12          }else{
13              $tid = $_POST['tid'];
14              $_SESSION['track_tid'] = $tid;
15              if(empty($error)){
16                  $hide = '';
17                  $trackid = $_SESSION['track_tid'];
18                  $sql = "SELECT * FROM status WHERE TrackingID='$tid'";
19                  $result = mysqli_query($conn, $sql);
20                  if(mysqli_num_rows($result) > 0){
21                      $status = mysqli_fetch_assoc($result);
22                      $active = array();
23                      if(! is_null($status['Delivered'])){
24                          $active['Delivered'] = $active['Out_for_delivery'] = $active['Shipped'] = 'active';
25                      }elseif(! is_null($status['Out_for_delivery'])){
26                          $active['Delivered'] = '';
27                          $active['Out_for_delivery'] = $active['Shipped'] = 'active';
28                      }elseif(! is_null($status['Shipped'])){
29                          $active['Delivered'] = $active['Out_for_delivery'] = '';
30                          $active['Shipped'] = 'active';
31                      }
32                  }else{
33                      $error = "Invalid Tracking ID";
34                  }
35              }
36          }
37      }
38      $hidden = 'hidden';
39      if(isset($_POST['view'])){
40          $trackid = $_SESSION['track_tid'];
41          $hidden = $hide = '';
42      }
43      $name = $add = $contact = '';
44      $errors = array('name' => '', 'add' => '', 'cont' => '');
45      if(isset($_POST['update'])){
46          $hidden = $hide = '';
47          $trackid = $_SESSION['track_tid'];
48          if(empty($_POST['fname'])){
49              $errors['name'] = "*Required";
50          }else{
51              $name = $_POST['fname'];
52          }
53          if(empty($_POST['fadd'])){
54              $errors['add'] = "*Required";
55          }else{
56              $add = $_POST['fadd'];
57          }
58          if(empty($_POST['fcontact'])){
59              $errors['cont'] = "*Required";
60          }else{
```

```

61     $contact = $_POST['fcontact'];
62 }
63 if(! array_filter($errors)){
64     $trackid = $_SESSION['track_tid'];
65     $sql = "UPDATE parcel SET R_Name = '$name', R_Add = '$add', R_Contact = $contact WHERE TrackingID = $trackid";
66 if(mysqli_query($conn, $sql)){
67     echo '<script type="text/javascript">';
68     echo "setTimeout(function () { swal('Address Updated', 'Receiver address updated successfully !!', 'success');";
69     echo '}, 1000);</script>';
70     $hide = $hidden = 'hidden';
71     $trackid = '';
72 }else{
73     echo 'Update Error : '.mysqli_error($conn);
74 }
75 }
76 ?>
77

```

Fig 16.1 : PHP code snippet with the corresponding SQL query to fetch the record of the entered tracking id. On successful record fetch, the receiver's address can be updated before the package is updated for out for delivery. Also this form collecting the new address remains hidden until the user hits the update delivery address button.



Fig 16.2 : The result of fetching the delivery status record of the entered tracking id from status table. The progress bar here gives a pictorial view on the progress of the package. Also the option to change the receiver's address is given.

Unable to receive on the expected date?

Drop to a friend nearby in the your city. [Update Delivery Address](#)

**Friend's Details**

Name :

Address :

Contact :

[Update](#)

Fig 16.3 : The form in the tracking page collecting the values to update the receiver's address residing in the same city.

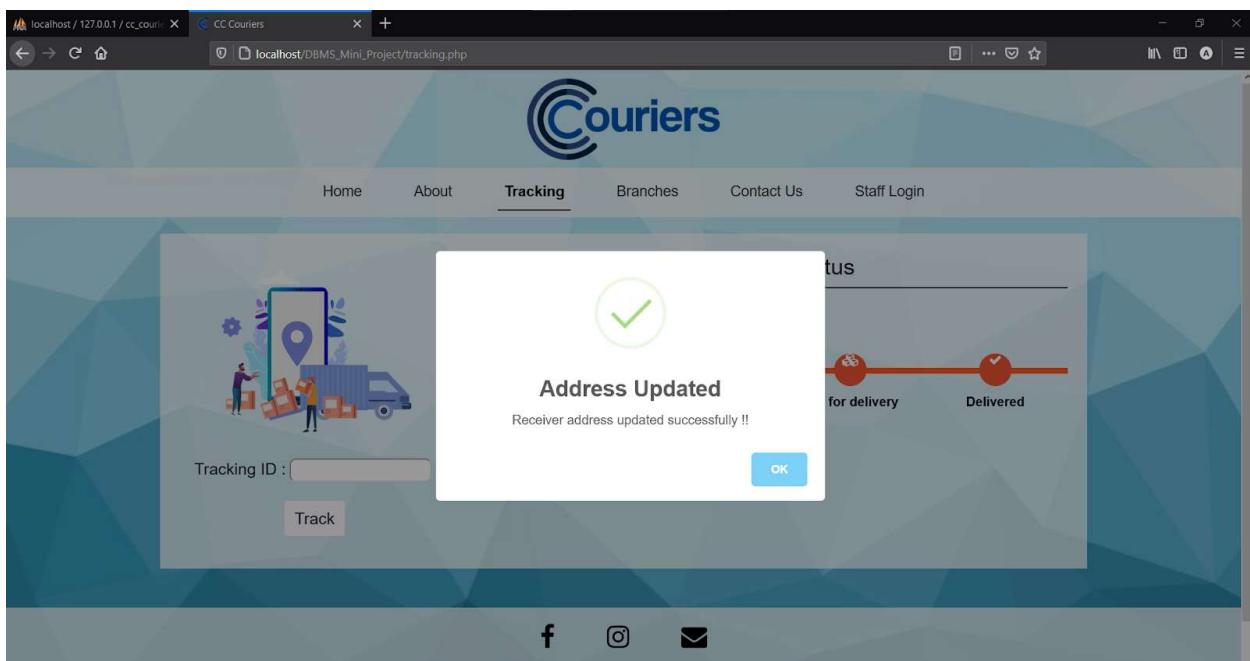


Fig 16.4 : The Success alert prompts the user that the delivery address is successfully updated.

+ status	<a href="#">Copy</a>	<a href="#">Delete</a>	10823	CC5555	Sudharshan	Greenpark layout	Udumalpet	Tamil Nadu	9485760123	Sharq	27, CB Street	Coimbatore	Tamil Nadu	9514200368	10.00	500.00	2021-05-17 10:53:18
+ Views	<a href="#">Copy</a>	<a href="#">Delete</a>	10824	CC3503	Ajitesh M	1, TRN Gardens	Udumalpet	Tamil Nadu	9488393922	Emma	12, Charminar cross street	Theni	Tamil Nadu	9912365407	2.50	125.00	2021-05-28 21:37:57
+ New	<a href="#">Copy</a>	<a href="#">Delete</a>	10825	CC3503	Alanc	10, Salvatore Printing House	Chennai	Tamil Nadu	9633221108	Dorian	15, Amazon School bookstore	Hyderabad	Andhra Pradesh	9056823147	23.00	2875.00	2021-05-29 15:15:53
+ arrived	<a href="#">Copy</a>	<a href="#">Delete</a>	10826	CC3503	Karthick	78, Tatabad	Coimbatore	Tamil Nadu	9417417410	Celine	34, Cheta gardens	Alapuzha	Kerala	9514236870	3.50	350.00	2021-05-29 15:26:35

Fig 16.5 : The result of successful updation of receiver's address is reflected in the parcel table (to the record tracking id = 10824.)



Fig 16.6 : When an Invalid tracking id is entered.

## 14. DISPLAYING THE HISTORY OF CONSIGNMENTS :

```

148     $sql = "SELECT * FROM arrived";
149     $result = mysqli_query($conn, $sql);
150     $arr = mysqli_fetch_all($result, MYSQLI_ASSOC);
151     $sql = "SELECT * FROM delivered";
152     $result = mysqli_query($conn, $sql);
153     $delivered = mysqli_fetch_all($result, MYSQLI_ASSOC);

```

Fig 17.1 : PHP code snippet with corresponding SQL queries to fetch the entries of the arrived and delivered views to display the consignments in transit and the delivered consignments resp.

localhost/phpmyadmin/sql.php?db=cc\_couriers&table=arrived&pos=0

	TrackingID	StaffID	S_Name	S_Add	S_City	S_State	S_Contact	R_Name	R_Add	R_City	R_State	R_Contact	Weight_Kg	Price	Delivered
<input type="checkbox"/>	10807	CC0246	Lizzie	5, Salvatore Boarding House	Coimbatore	Tamil Nadu	9487512036	Jossie	8, French quarters	Agra	Uttar Pradesh	941212360	2.75	962.50	20
<input type="checkbox"/>	10809	CC0246	Lizzie	5, Salvatore Boarding House	Coimbatore	Tamil Nadu	9487512036	Jossie	8, French quarters	Agra	Uttar Pradesh	941212360	2.75	962.50	20
<input type="checkbox"/>	10812	CC0246	Karthick	1, TRN Gardens	Udumalpet	Tamil Nadu	9417417410	Dhamini	7/8, Arun excelo, Avadi	Chennai	Tamil Nadu	9412365870	2.00	100.00	20
<input type="checkbox"/>	10814	CC1234	Rajesh	6, Angel Residency	Theni	Tamil Nadu	9955487612	Sanjuktha	15, Eden palace	Kochi	Kerala	9912365407	1.00	100.00	20
<input type="checkbox"/>	10815	CC3267	Rishi	78, Talabadi	Coimbatore	Tamil Nadu	9633652140	Celine	34, Cheta gardens	Palakkad	Kerala	9514236870	2.75	275.00	20
<input type="checkbox"/>	10816	CC3267	Rishi	78, Talabadi	Coimbatore	Tamil Nadu	9852630147	Celine	34, Cheta gardens	Palakkad	Kerala	9514273680	2.00	200.00	20
<input type="checkbox"/>	10817	CC3267	Jai	6, Angel Residency	Trichy	Tamil Nadu	9178542036	Arya	75, Main cross street	Alapuzha	Kerala	9514236870	2.00	200.00	20
<input type="checkbox"/>	10820	CC4567	Eswar	12, MJ Gardens	Erode	Tamil Nadu	9922663300	Chanya	45, BB street	Theni	Tamil Nadu	9956823147	1.00	50.00	20
<input type="checkbox"/>	10821	CC1212	Aathmika	1, TRN Gardens	Udumalpet	Tamil Nadu	9512034687	Sanjuktha	15, Eden palace	Theni	Tamil Nadu	9912365407	7.00	350.00	20
<input type="checkbox"/>	10822	CC5555	Sudharshan	1, Greenpark layout	Udumalpet	Tamil Nadu	9485760123	Shariq	27, CB Street	Coimbatore	Tamil Nadu	9514200368	10.00	500.00	20
<input type="checkbox"/>	10823	CC5555	Sudharshan	1, Greenpark layout	Udumalpet	Tamil Nadu	9485760123	Shariq	27, CB Street	Coimbatore	Tamil Nadu	9514200368	10.00	500.00	20
<input type="checkbox"/>	10824	CC3503	Ajitesh M	1, TRN Gardens	Udumalpet	Tamil Nadu	9488393922	Emma	12, Charminar cross street	Theni	Tamil Nadu	9912365407	2.50	125.00	20
<input type="checkbox"/>	10825	CC3503	Alaric	10, Salvatore Printing	Chennai	Tamil Nadu	9633221108	Dorian	15, Amazon School	Hyderabad	Andhra Pradesh	9956823147	23.00	2875.00	20

Fig 17.2 : Entries of the arrived view with the consignments in transit (ie) Delivered = NULL .

localhost / cc\_courier

localhost/DBMS\_Mini\_Project/staff.php

Consignments											
Arrived		Delivered									
TrackingID	StaffID	Sender		Receiver		Weight	Price	Dispatched	Shipped	Out for delivery	Delivered
10807	CC0246	Lizzie, 5, Salvatore Boarding House, Coimbatore, Tamil Nadu - 9487512036		Jossie, 8, French quarters, Agra, Uttar Pradesh - 941212360		2.75	962.50	2021-05-06 16:04:38	2021-05-09 21:54:42		
10809	CC0246	Lizzie, 5, Salvatore Boarding House, Coimbatore, Tamil Nadu - 9487512036		Jossie, 8, French quarters, Agra, Uttar Pradesh - 941212360		2.75	962.50	2021-05-06 16:33:57			
10812	CC0246	Karthick, 1, TRN Gardens, Udumalpet, Tamil Nadu - 9417417410		Dhamini, 7/8, Arun excelo, Avadi, Chennai, Tamil Nadu		2.00	100.00	2021-05-06 19:00:20	2021-05-06 19:05:31		
10814	CC1234	Rajesh, 6, Angel Residency, Theni, Tamil Nadu - 9955487612		Sanjuktha, 15, Eden palace, Kochi, Kerala - 9912365407		1.00	100.00	2021-05-10 15:00:29	2021-05-29 15:08:05		

Fig 17.3 : Displaying the consignments in transit, which are the entries of the arrived view.

	TrackingID	StaffID	S_Name	S_Add	S_City	S_State	S_Contact	R_Name	R_Add	R_City	R_State	R_Contact	Weight_Kg	Price	Dispatched
<input type="checkbox"/>	10818	CC3267	Jai	6, Angel Residency	Trichy	Tamil Nadu	9178542036	Arya	75, Main cross street	Alappuzha	Kerala	9514236870	2.00	200.00	2021-05-10 23:11:38...
<input type="checkbox"/>	10813	CC0246	Karthick	1, TRN Gardens	Udumalpet	Tamil Nadu	9417417410	Dhamini	7/8, Arun excelo,	Chennai	Tamil Nadu	9412365870	2.00	100.00	2021-05-06 16:31:43...
<input type="checkbox"/>	10808	CC0246	Lizzie	5, Salvatore Boarding House	Coimbatore	Tamil Nadu	9487512036	Jossie	8, French quarters	Agra	Uttar Pradesh	941212360	2.75	962.50	2021-05-06 19:05:01

Fig 17.4 : Entries of the delivered view.

TrackingID	StaffID	Sender	Receiver	Weight	Price	Dispatched	Shipped	Out for delivery	Delivered
10808	CC0246	Lizzie, 5, Salvatore Boarding House, Coimbatore, Tamil Nadu - 9487512036	Jossie, 8, French quarters, Agra, Uttar Pradesh - 941212360	2.75	962.50	2021-05-06 16:31:43	2021-05-06 18:29:19	2021-05-29 15:07:28	2021-05-29 15:10:31
10813	CC0246	Karthick, 1, TRN Gardens, Udumalpet, Tamil Nadu - 9417417410	Dhamini, 7/8, Arun excelo, Avadi, Chennai, Tamil Nadu - 9412365870	2.00	100.00	2021-05-06 19:05:01	2021-05-06 19:13:22	2021-05-06 19:17:41	2021-05-06 19:18:05
10818	CC3267	Jai, 6, Angel Residency, Trichy, Tamil Nadu - 9178542036	Arya, 75, Main cross street, Alappuzha, Kerala - 9514236870	2.00	200.00	2021-05-10 23:11:38	2021-05-11 00:35:47	2021-05-11 00:36:25	2021-05-11 00:36:37

Fig 17.5 : Displaying the delivered consignments, which are the entries of the delivered view.

## 15. FEEDBACK/QUERY FILING :

```
$name = $email = $msg = '';
$error = array('name' => '', 'email' => '', 'msg' => '');
if(isset($_POST['submit'])){
    if(empty($_POST['name'])){
        $error['name'] = "*Required";
    }else{
        $name = $_POST['name'];
    }
    if(empty($_POST['email'])){
        $error['email'] = "*Required";
    }else{
        if(email_validation($_POST['email'])){
            $email = $_POST['email'];
        }else{
            $error['email'] = "*Invalid email";
        }
    }
    if(empty($_POST['msg'])){
        $error['msg'] = "*Required";
    }else{
        $msg = $_POST['msg'];
    }
    if(!array_filter($error)){
        $sql = "INSERT INTO feedback (Cust_name, Cust_mail, Cust_msg) VALUES ('$name', '$email', '$msg')";
        if(mysqli_query($conn, $sql)){
            echo '<script type="text/javascript">';
            echo 'setTimeout(function () { swal("Thank You", "Your response recorded successfully !!", "success"); }, 1000);</script>';
        }else{
            echo "Insert Error : " . mysqli_error($conn);
        }
    }
}
```

Fig 18.1 : PHP code snippet to store the contents from the contact form provide for feedback and queries into the feedback table.

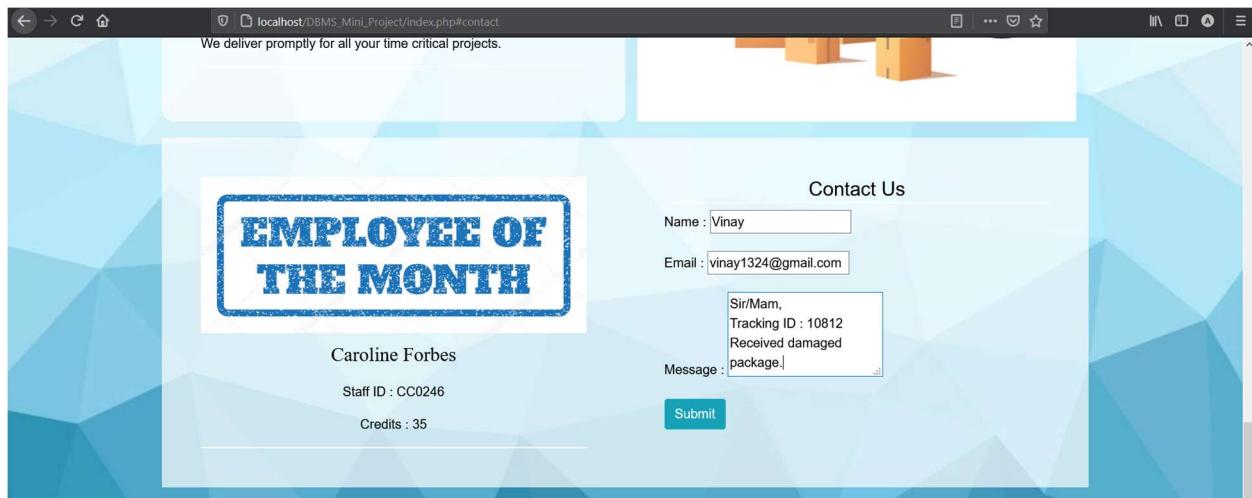


Fig 18.2 : The Contact form with details entered.

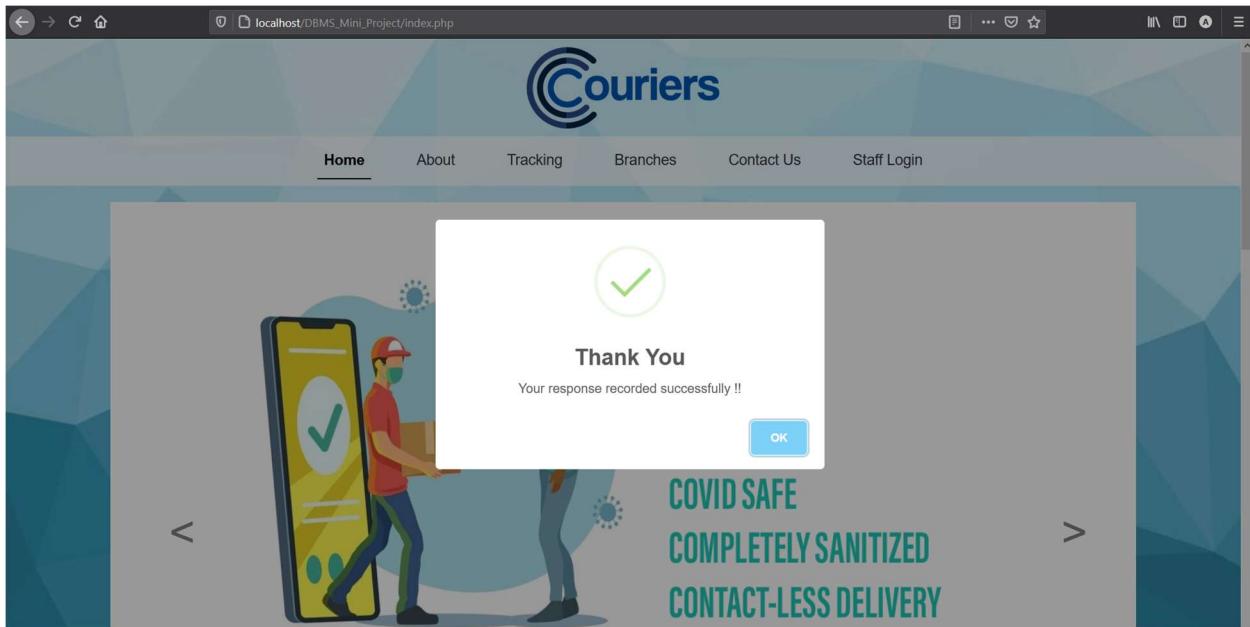


Fig 18.3 : On successful submission of the feedback/query success alert is prompted.

A screenshot of the phpMyAdmin interface, specifically the 'feedback' table in the 'cc\_couriers' database. The table structure includes columns for F.No, Cust\_name, Cust\_mail, and Cust\_msg. Six rows of data are visible, with the last row (F.No 6) being the most recent addition. The 'Cust\_msg' column for this entry contains the text: 'Sir/Mam, Tracking ID : 10812 Received damaged pa...'. Other entries in the table include comments from customers like Ajitesh Mahalingam, Torvus, Ajitesh M, Madhu, and Vinay.

Fig 18.4 : The result of successful submission is reflected in the feedback table at entry f.no 6.

## 16. EMPLOYEE OF THE MONTH :

```
$sql = "SELECT * FROM staff WHERE credits = (SELECT MAX(credits) FROM staff)";
$result = mysqli_query($conn, $sql);
if(mysqli_num_rows($result) > 0){
    $empmonth = mysqli_fetch_all($result, MYSQLI_ASSOC);

}else{
    echo "Error : ". mysqli_error($conn);
}
```

Fig 19.1 : PHP code snippet with the corresponding SQL query to fetch the record of the staff with maximum credits and displayed as the Employee of the month. Note that each staff is awarded 5 credits for each parcel they place (which is added to the credits in staff table by the placeParcel trigger) and the event – resetCredits – resets the credits of all employees to 0 on every month beginning.



Fig 19.2 : The Employee of the month featured in the home page.

## 17. DISPLAYING BRANCH DETAILS :

```
branches.php
1 <?php
2     include("db_connect.php");
3     $sql = "SELECT * FROM branches";
4     $result = mysqli_query($conn, $sql);
5     $branches = mysqli_fetch_all($result, MYSQLI_ASSOC);
6 ?>
```

Fig 20.1 : PHP code snippet with corresponding SQL query to fetch the records of the branches table.

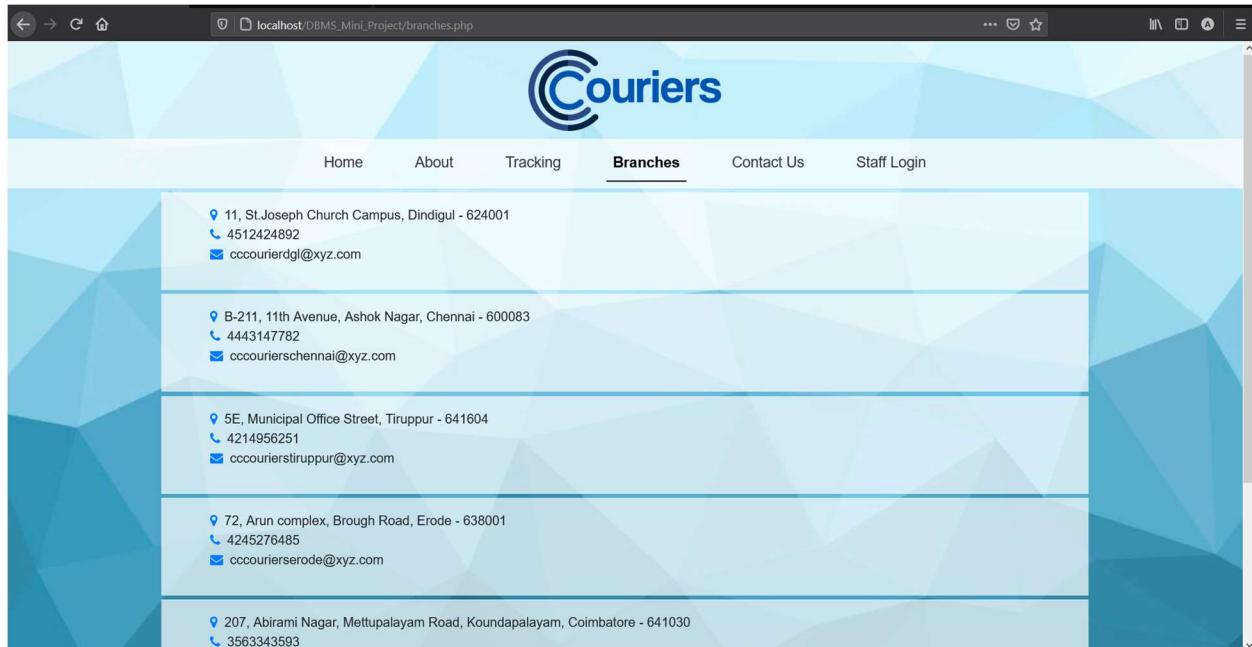


Fig 20.2 : The Branches page displaying the address and contact details of various branch offices as per the entries fetched from the branches table.

---

## Conclusion

From understanding the user and admin requirements to system design and finally consolidation of everything, each step requires in-depth understanding and commitment toward achieving the objectives of this project. CC Couriers is a customer-friendly delivery service with flexible policies and proper management system.

Although the Courier Management System developed in this project is not fully integrated to the real world needs of a system, the prototype and implementation demonstrates easy navigation in the system and how data are stored in a systematic view. The study of how a relational database will be developed and maintained for industry purposes was done and been implemented. Overall, the main motive of this project is to gain more knowledge about the usage of database systems in the industrial view which was done efficiently. Also it is indeed a great learning experience.

---

# References

- ❖ <https://www.w3schools.com/>
- ❖ <https://www.javatpoint.com/dbms-tutorial>
- ❖ <https://www.guru99.com>
- ❖ [www.stackoverflow.com](http://www.stackoverflow.com)
- ❖ <https://www.youtube.com/>
- ❖ <https://getbootstrap.com/docs/4.0/getting-started/introduction/>
- ❖ Database System Concepts, Seventh Edition by Abraham Silberschatz, Henry F. Korth, S. Sudarshan