

# CHAPTER 1

## INTRODUCTION

### 1.1 DATABASE MANAGEMENT SYSTEM

A database management system (DBMS) refers to the technology for creating and managing databases. Basically, DBMS is a software tool to organize (create, retrieve, update and manage) data in a database. The main aim of a DBMS is to supply a way to store up and retrieve database information that is both convenient and efficient.

Database systems are meant to handle large collection of information. Management of data involves both defining structures for storage of information and providing mechanisms that can do the manipulation of those stored information. Moreover, the database system must ensure the safety of the information stored, despite system crashes or attempts at unauthorized access.

### 1.2 HISTORY

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

The two-main early navigational data models were the hierarchical model, epitomized by IBM's IMS system, and the CODASYL model (network model), implemented in a number of products such as IDMS.

The relational model, first proposed in 1970 by Edgar F. Codd, departed from this tradition by insisting that applications should search for data by content, rather than by following links. The relational model employs set of ledger-style tables, each used for a different type of entity. Only in the mid-1980s did computing hardware become powerful enough to allow the wide deployment of relational systems (DBMSs plus applications).

By the early 1990s, however, relational systems dominated in all large-scale data processing applications, and as of 2015 they remain dominant: IBM DB2, Oracle, MySQL, and Microsoft SQL Server are the top DBMS. The dominant database language, standardized SQL for the relational model, has influenced database languages for other data models.

Object databases were developed in the 1980s to overcome the inconvenience of object-relational impedance mismatch, which led to the coining of the term "post-relational" and also the development of hybrid object-relational databases.

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

### **1.3 Objective**

The objectives of this study are summarized below:

- To automate the existing leave management in company.
- To add and manage employee and their details.
- To add and manage departments
- To provide separate username and password for employees who are working for different departments.
- To keep the track of employee leave details, so that admin can take action regarding whether to approve leave or not.

## CHAPTER 2

### REQUIREMENT SPECIFICATION

#### **2.1 Hardware requirements:**

The hardware required for the development of this project is :

- Processor : Intel Core i5
- Processor speed : 1.8 GHz
- RAM : 2 GB RAM
- System Type : 64-Bit Operating System

#### **2.2 Software requirements:**

The software required for the development of this project is:

- Software : WAMPP
- Operating System : Windows 10
- Front End : HTML , CSS
- Programming Language : PHP
- Data Base Environment : MySQL and PHP My Admin
- Server : APACHE

## CHAPTER 3

### DESIGN

#### 3.1 ENTITY - RELATIONSHIP DIAGRAM

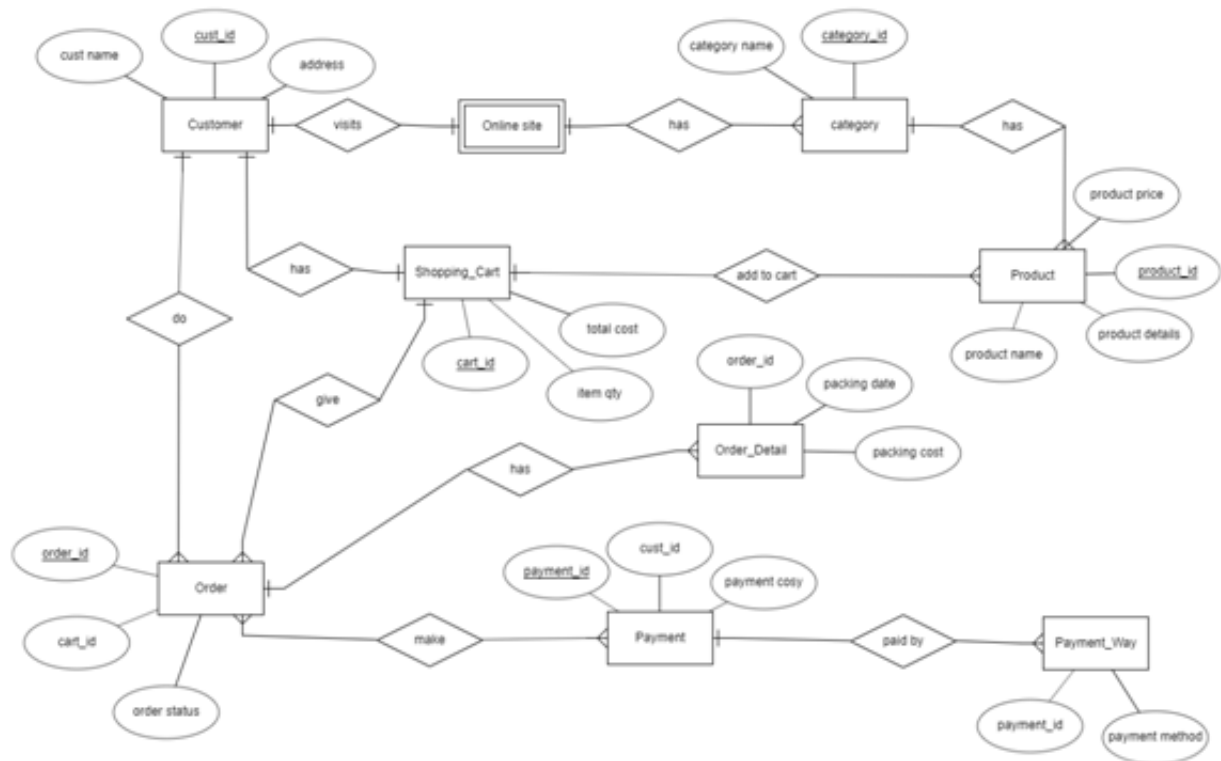


Fig 3.1: Entity Relationship Diagram of Online Shopping Management System

The ER Model figure shows the proposed system. It defines the conceptual view of a database. It works around real-world entities and the associations among them. At view level, the ER model is considered a good option for designing databases.

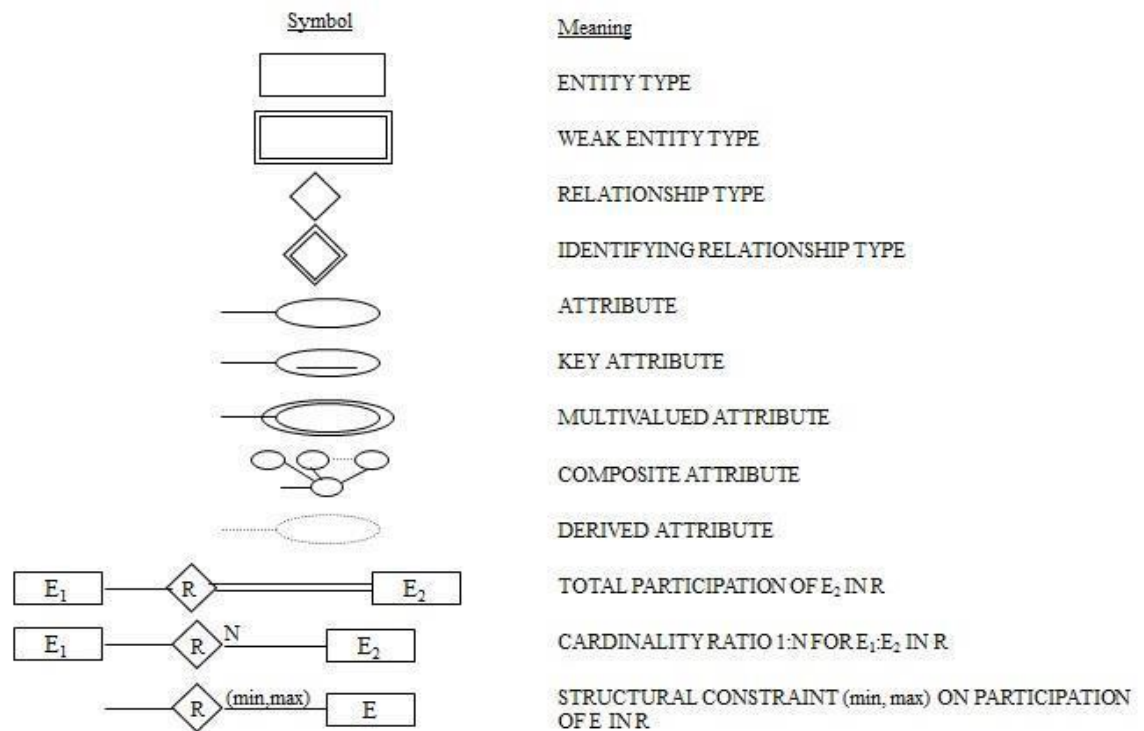


Fig- 3.1.1 ER DIAGRAM NOTATION SUMMARY

## 3.2 DESCRIPTION:

The ER Model figure shows the proposed system. It defines the conceptual view of the database. It works around real-world entities and the associations among them. At view level, the ER model is considered a good option for designing databases. So, let's see each entity

### Admin

This entity stores the information about admin who registers and logs in using his username and password. Attributes are id, username, password, creationDate, updationDate.

### Category

This entity stores the information about category. Attributes are id, categoryName, categoryDescription, creationDate, updationDate.

## Customer

This entity stores the information about customer. Attributes are custmoerid, customerName, address.

## Order

This entity stores the information about order. Attributes are id, userid, productid, quantity, orderDate, paymentMethod, orderStatus.

## Product

This entity stores the information about product. Attributes are productid, productName, productDetails, productPrice.

## Order Details

This entity stores the information about order details. Attributes are orderid, packingCost, packingDate.

## Shopping Cart

This entity stores the information about shopping cart. Attributes are cartid, itemQuantity, totalCost.

## Payment

This entity stores the information about payment. Attributes are paymentid, amount, customerid, paymentType.

### 3.3 SCHEMA DIAGRAM

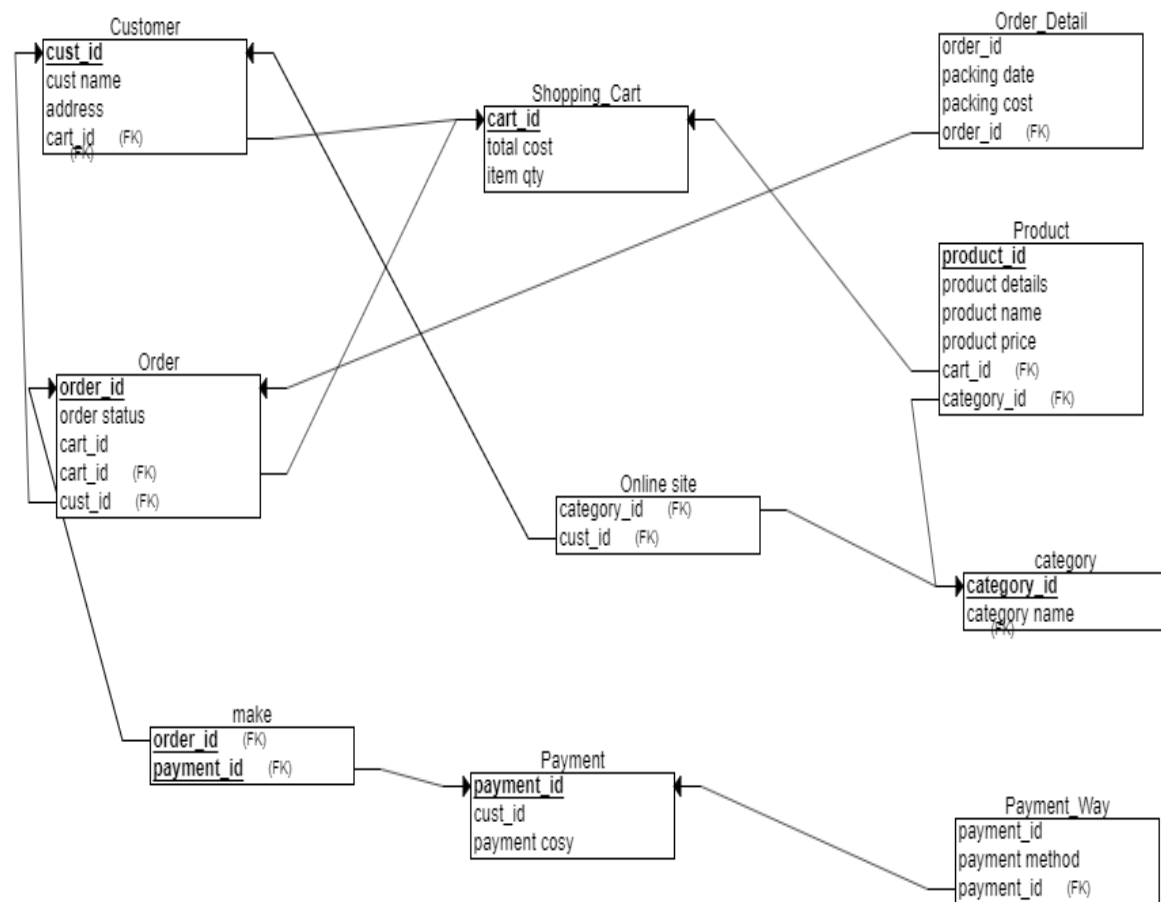


Fig 3.3 Schema Diagram of Online Shopping Management System

### 3.4 SEVEN STEPS FOR ER TO SCHEMA CONVERSION

#### **Step 1: Mapping of Regular Entity Types.**

For each regular (strong) entity type *E* in the ER schema, create a relation *R* that includes all the simple attributes of *E*. Include only the simple component attributes of a composite attribute. Choose one of the key attributes of *E* as the primary key for *R*. If the chosen key of *E* is a composite, then the set of simple attributes that form it will together form the primary key of *R*. If multiple keys were identified for *E* during the conceptual design, the information describing the attributes that form each additional key is kept in order to specify secondary (unique) keys of relation *R*. Knowledge about keys is also kept for indexing purposes and other types of analyses.

#### **Step 2: Mapping of Weak Entity Types.**

For each weak entity type *W* in the ER schema with owner entity type *E*, create a relation *R* and include all simple attributes (or simple components of composite attributes) of *W* as attributes of *R*. In addition, include as foreign key attributes of *R*, the primary key attribute(s) of the relation(s) that correspond to the owner entity type(s); this takes care of mapping the identifying relationship type of *W*. The primary key of *R* is the combination of the primary key(s) of the owner(s) and the partial key of the weak entity type *W*, if any. If there is a weak entity type *E2* whose owner is also a weak entity type *E1*, then *E1* should be mapped before *E2* to determine its primary key first.

#### **Step 3: Mapping of Binary 1:1 Relationship Types.**

For each binary 1:1 relationship type *R* in the ER schema, identify the relations *S* and *T* that correspond to the entity types participating in *R*. There are three possible approaches:

1. The foreign key approach.
2. The merged relationship approach, and

The first approach is the most useful and should be followed unless special conditions exist, as we discuss below.

##### **1. Foreign key approach:**

Choose one of the relations—*S*, say—and include as a foreign key in *S* the primary key of *T*. It is better to choose an entity type with total participation in *R* in the role of *S*. Include all the simple attributes (or simple components of composite attributes) of the 1:1 relationship type *R* as attributes of *S*.



## **2. Merged relation approach:**

An alternative mapping of a 1:1 relationship type is to merge the two entity types and the relationship into a single relation. This is possible when both participations are total, as this would indicate that the two tables will have the exact same number of tuples at all times.

### **Step 3. Cross-reference or relationship relation approach:**

The third option is to set up a third relation R for the purpose of cross-referencing the primary keys of the two relations S and T representing the entity types. As we will see, this approach is required for binary M:N relationships. The relation R is called a relationship relation (or sometimes a lookup table), because each tuple in R represents a relationship instance that relates one tuple from S with one tuple from T. The relation R will include the primary key attributes of S and T as foreign keys to S and T. The primary key of R will be one of the two foreign keys, and the other foreign key will be a unique key of R. The drawback is having an extra relation, and requiring an extra join operation when combining related tuples from the tables.

### **Step 4: Mapping of Binary 1:N Relationship Types.**

For each regular binary 1:N relationship type R, identify the relation S that represents the participating entity type at the N-side of the relationship type. Include as foreign key in S the primary key of the relation T that represents the other entity type participating in R; we do this because each entity instance on the N-side is related to at most one entity instance on the 1-side of the relationship type. Include any simple attributes (or simple components of composite attributes) of the 1:N relationship type as attributes of S.

### **Step 5: Mapping of Binary M:N Relationship Types.**

For each binary M:N relationship type R, create a new relation S to represent R. Include as foreign key attributes in S the primary keys of the relations that represent the participating entity types; their combination will form the primary key of S. Also include any simple attributes of the M:N relationship type (or simple components of composite attributes) as attributes of S.

Notice that we cannot represent an M:N relationship type by a single foreign key attribute in one of the participating relations (as we did for 1:1 or 1:N relationship types) because of the M:N cardinality ratio; we must create a separate relationship relation S.

**Step 6: Mapping of Multivalued Attributes.**

For each multivalued attribute A, create a new relation R. This relation R will include an attribute corresponding to A, plus the primary key attribute K—as a foreign key in R—of the relation that represents the entity type or relationship type that has A as a multivalued attribute.

The primary key of R is the combination of A and K. If the multivalued attribute is composite, we include its simple components.

**Step 7: Mapping of N-array Relationship Types.**

For each n-array relationship type R, where  $n > 2$ , create a new relation S to represent R. Include as foreign key attributes in S the primary keys of the relations that represent the participating entity types. Also include any simple attributes of the n-array relationship type (or simple components of composite attributes) as attributes of S. The primary key of S is usually a combination of all the foreign keys that reference the relations representing the participating entity types. However, if the cardinality constraints on any of the entity types E participating in R is 1, then the primary key of S should not include the foreign key attribute that references the relation E ‘corresponding to E.

## 3.5 DATABASE DESCRIPTION:

### 3.5.1 Tables in Backend

Table	Action	Rows	Type	Collation	Size	Overhead
admin	Browse Structure Search Insert Empty Drop	~1	InnoDB	latin1_swedish_ci	16 KiB	-
category	Browse Structure Search Insert Empty Drop	~5	InnoDB	latin1_swedish_ci	16 KiB	-
orders	Browse Structure Search Insert Empty Drop	~3	InnoDB	latin1_swedish_ci	16 KiB	-
ordertrackhistory	Browse Structure Search Insert Empty Drop	~5	InnoDB	latin1_swedish_ci	16 KiB	-
productreviews	Browse Structure Search Insert Empty Drop	~1	InnoDB	latin1_swedish_ci	16 KiB	-
products	Browse Structure Search Insert Empty Drop	~19	InnoDB	latin1_swedish_ci	16 KiB	-
subcategory	Browse Structure Search Insert Empty Drop	~12	InnoDB	latin1_swedish_ci	16 KiB	-
userlog	Browse Structure Search Insert Empty Drop	~9	InnoDB	latin1_swedish_ci	16 KiB	-
users	Browse Structure Search Insert Empty Drop	~1	InnoDB	latin1_swedish_ci	16 KiB	-
wishlist	Browse Structure Search Insert Empty Drop	~0	InnoDB	latin1_swedish_ci	16 KiB	-
<b>10 tables</b>	<b>Sum</b>	<b>56</b>	<b>InnoDB</b>	<b>latin1_swedish_ci</b>	<b>160 KiB</b>	<b>0 B</b>

Table 3.5.1: Table description for online shopping system.

### 3.5.2 Admin

#	Name	Type	Collation	Attributes	Null	Default	Extra	Action
1	id	int(11)			No	None	AUTO_INCREMENT	Change  Drop  More
2	username	varchar(255)	latin1_swedish_ci		No	None		Change  Drop  More
3	password	varchar(255)	latin1_swedish_ci		No	None		Change  Drop  More
4	creationDate	timestamp			No	CURRENT_TIMESTAMP		Change  Drop  More
5	updatationDate	varchar(255)	latin1_swedish_ci		No	None		Change  Drop  More

Table 3.5.2: Table description for admin record.

### 3.5.3 Category

#	Name	Type	Collation	Attributes	Null	Default	Extra	Action
1	id	int(11)			No	None	AUTO_INCREMENT	Change  Drop  More
2	categoryName	varchar(255)	latin1_swedish_ci		Yes	NULL		Change  Drop  More
3	categoryDescription	longtext	latin1_swedish_ci		Yes	NULL		Change  Drop  More
4	creationDate	timestamp			No	CURRENT_TIMESTAMP		Change  Drop  More
5	updatationDate	varchar(255)	latin1_swedish_ci		Yes	NULL		Change  Drop  More

Table 3.5.3: Table description for category.

### 3.5.4 Order

#	Name	Type	Collation	Attributes	Null	Default	Extra	Action
<input type="checkbox"/> 1	<b>id</b>	int(11)			No	None	AUTO_INCREMENT	Change  Drop  More
<input type="checkbox"/> 2	<b>userId</b>	int(11)			Yes	NULL		Change  Drop  More
<input type="checkbox"/> 3	<b>productId</b>	varchar(255)	latin1_swedish_ci		Yes	NULL		Change  Drop  More
<input type="checkbox"/> 4	<b>quantity</b>	int(11)			Yes	NULL		Change  Drop  More
<input type="checkbox"/> 5	<b>orderDate</b>	timestamp			No	CURRENT_TIMESTAMP		Change  Drop  More
<input type="checkbox"/> 6	<b>paymentMethod</b>	varchar(50)	latin1_swedish_ci		Yes	NULL		Change  Drop  More
<input type="checkbox"/> 7	<b>orderStatus</b>	varchar(55)	latin1_swedish_ci		Yes	NULL		Change  Drop  More

Table 3.5.4: Table description for order record.

### 3.5.5 Order track history

#	Name	Type	Collation	Attributes	Null	Default	Extra	Action
<input type="checkbox"/> 1	<b>id</b>	int(11)			No	None	AUTO_INCREMENT	Change  Drop  More
<input type="checkbox"/> 2	<b>orderId</b>	int(11)			Yes	NULL		Change  Drop  More
<input type="checkbox"/> 3	<b>status</b>	varchar(255)	latin1_swedish_ci		Yes	NULL		Change  Drop  More
<input type="checkbox"/> 4	<b>remark</b>	mediumtext	latin1_swedish_ci		Yes	NULL		Change  Drop  More
<input type="checkbox"/> 5	<b>postingDate</b>	timestamp			No	CURRENT_TIMESTAMP		Change  Drop  More

Table 3.5.5: Table description for order track history.

### 3.5.6 Product reviews

#	Name	Type	Collation	Attributes	Null	Default	Extra	Action
<input type="checkbox"/> 1	<b>id</b>	int(11)			No	None	AUTO_INCREMENT	Change  Drop  More
<input type="checkbox"/> 2	<b>category</b>	int(11)			No	None		Change  Drop  More
<input type="checkbox"/> 3	<b>subCategory</b>	int(11)			Yes	NULL		Change  Drop  More
<input type="checkbox"/> 4	<b>productName</b>	varchar(255)	latin1_swedish_ci		Yes	NULL		Change  Drop  More
<input type="checkbox"/> 5	<b>productCompany</b>	varchar(255)	latin1_swedish_ci		Yes	NULL		Change  Drop  More
<input type="checkbox"/> 6	<b>productPrice</b>	int(11)			Yes	NULL		Change  Drop  More
<input type="checkbox"/> 7	<b>productPriceBeforeDiscount</b>	int(11)			Yes	NULL		Change  Drop  More
<input type="checkbox"/> 8	<b>productDescription</b>	longtext	latin1_swedish_ci		Yes	NULL		Change  Drop  More
<input type="checkbox"/> 9	<b>productImage1</b>	varchar(255)	latin1_swedish_ci		Yes	NULL		Change  Drop  More
<input type="checkbox"/> 10	<b>productImage2</b>	varchar(255)	latin1_swedish_ci		Yes	NULL		Change  Drop  More
<input type="checkbox"/> 11	<b>productImage3</b>	varchar(255)	latin1_swedish_ci		Yes	NULL		Change  Drop  More
<input type="checkbox"/> 12	<b>shippingCharge</b>	int(11)			Yes	NULL		Change  Drop  More
<input type="checkbox"/> 13	<b>productAvailability</b>	varchar(255)	latin1_swedish_ci		Yes	NULL		Change  Drop  More
<input type="checkbox"/> 14	<b>postingDate</b>	timestamp			Yes	CURRENT_TIMESTAMP		Change  Drop  More
<input type="checkbox"/> 15	<b>updateDate</b>	varchar(255)	latin1_swedish_ci		Yes	NULL		Change  Drop  More

Table 3.5.6: Table description for product reviews.

### 3.5.7 Product

#	Name	Type	Collation	Attributes	Null	Default	Extra	Action
<input type="checkbox"/> 1	<b>id</b>	int(11)			No	None	AUTO_INCREMENT	Change  Drop  More
<input type="checkbox"/> 2	<b>categoryid</b>	int(11)			Yes	NULL		Change  Drop  More
<input type="checkbox"/> 3	<b>subcategory</b>	varchar(255)	latin1_swedish_ci		Yes	NULL		Change  Drop  More
<input type="checkbox"/> 4	<b>creationDate</b>	timestamp			Yes	CURRENT_TIMESTAMP		Change  Drop  More
<input type="checkbox"/> 5	<b>updationDate</b>	varchar(255)	latin1_swedish_ci		Yes	NULL		Change  Drop  More

Table 3.5.7: Table description for products.

### 3.5.8 Subcategory

#	Name	Type	Collation	Attributes	Null	Default	Extra	Action
<input type="checkbox"/> 1	<b>id</b>	int(11)			No	None	AUTO_INCREMENT	Change  Drop  More
<input type="checkbox"/> 2	<b>categoryid</b>	int(11)			Yes	NULL		Change  Drop  More
<input type="checkbox"/> 3	<b>subcategory</b>	varchar(255)	latin1_swedish_ci		Yes	NULL		Change  Drop  More
<input type="checkbox"/> 4	<b>creationDate</b>	timestamp			Yes	CURRENT_TIMESTAMP		Change  Drop  More
<input type="checkbox"/> 5	<b>updationDate</b>	varchar(255)	latin1_swedish_ci		Yes	NULL		Change  Drop  More

Table 3.5.8: Table description for subcategory.

### 3.5.9 Userlog

#	Name	Type	Collation	Attributes	Null	Default	Extra	Action
<input type="checkbox"/> 1	<b>id</b>	int(11)			No	None	AUTO_INCREMENT	Change  Drop  Primary  More
<input type="checkbox"/> 2	<b>userEmail</b>	varchar(255)	latin1_swedish_ci		Yes	NULL		Change  Drop  Primary  More
<input type="checkbox"/> 3	<b>userip</b>	binary(16)			Yes	NULL		Change  Drop  Primary  More
<input type="checkbox"/> 4	<b>loginTime</b>	timestamp			Yes	CURRENT_TIMESTAMP		Change  Drop  Primary  More
<input type="checkbox"/> 5	<b>logout</b>	varchar(255)	latin1_swedish_ci		Yes	NULL		Change  Drop  Primary  More
<input type="checkbox"/> 6	<b>status</b>	int(11)			Yes	NULL		Change  Drop  Primary  More

Table 3.5.9: Table description for userlogin information.

## 3.5.10 User

#	Name	Type	Collation	Attributes	Null	Default	Extra	Action
<input type="checkbox"/> 1	<b>id</b>	int(11)			No	None	AUTO_INCREMENT	Change  Drop  More
<input type="checkbox"/> 2	<b>name</b>	varchar(255)	latin1_swedish_ci		Yes	NULL		Change  Drop  More
<input type="checkbox"/> 3	<b>email</b>	varchar(255)	latin1_swedish_ci		Yes	NULL		Change  Drop  More
<input type="checkbox"/> 4	<b>contactno</b>	bigint(11)			Yes	NULL		Change  Drop  More
<input type="checkbox"/> 5	<b>password</b>	varchar(255)	latin1_swedish_ci		Yes	NULL		Change  Drop  More
<input type="checkbox"/> 6	<b>shippingAddress</b>	longtext	latin1_swedish_ci		Yes	NULL		Change  Drop  More
<input type="checkbox"/> 7	<b>shippingState</b>	varchar(255)	latin1_swedish_ci		Yes	NULL		Change  Drop  More
<input type="checkbox"/> 8	<b>shippingCity</b>	varchar(255)	latin1_swedish_ci		Yes	NULL		Change  Drop  More
<input type="checkbox"/> 9	<b>shippingPincode</b>	int(11)			Yes	NULL		Change  Drop  More
<input type="checkbox"/> 10	<b>billingAddress</b>	longtext	latin1_swedish_ci		Yes	NULL		Change  Drop  More
<input type="checkbox"/> 11	<b>billingState</b>	varchar(255)	latin1_swedish_ci		Yes	NULL		Change  Drop  More
<input type="checkbox"/> 12	<b>billingCity</b>	varchar(255)	latin1_swedish_ci		Yes	NULL		Change  Drop  More
<input type="checkbox"/> 13	<b>billingPincode</b>	int(11)			Yes	NULL		Change  Drop  More
<input type="checkbox"/> 14	<b>regDate</b>	timestamp			No	CURRENT_TIMESTAMP		Change  Drop  More
<input type="checkbox"/> 15	<b>updateDate</b>	varchar(255)	latin1_swedish_ci		Yes	NULL		Change  Drop  More

Table 3.5.10: Table description for user information.

## 3.5.11 Wishlist

#	Name	Type	Collation	Attributes	Null	Default	Extra	Action
<input type="checkbox"/> 1	<b>id</b>	int(11)			No	None	AUTO_INCREMENT	Change  Drop  Primary  More
<input type="checkbox"/> 2	<b>userId</b>	int(11)			Yes	NULL		Change  Drop  Primary  More
<input type="checkbox"/> 3	<b>productId</b>	int(11)			Yes	NULL		Change  Drop  Primary  More
<input type="checkbox"/> 4	<b>postingDate</b>	timestamp			No	CURRENT_TIMESTAMP		Change  Drop  Primary  More

Table 3.5.11: Table description for wishlist.

## CHAPTER 4

### SNAPSHOT

#### 4.1 Admin

Shopping Portal | Admin Back to Portal


Sign In

Login

© 2017 Shopping Portal All rights reserved.

Fig 4.1: Description of admin database

#### 4.2 Add Category

Shopping Portal | Admin Admin 

Order Management

Manage users

Create Category

Sub Category

Insert Product

Manage Products

User Login Log

Logout

Category

Category Name

Enter category Name

Description


Create

Manage Categories

Show 10 entries Search:

Fig 4.2: Description of creating category.

## 4.3 Manage Users

Shopping Portal | Admin Admin 

Order Management
Manage users
Create Category
Sub Category
Insert Product
Manage Products
User Login Log
Logout

### Manage Users


Show  entries Search:

#	Name	Email	Contact no	Shipping Address/City/State/Pincode	Billing Address/City/State/Pincode
1	Anusha	anusharavikumar666@gmail.com	9113817729	biet college,davangere,karnataka-577004	biet college,davangere,karnataka-577004

Showing 1 to 1 of 1 entries

Fig 4.4: Description of managing users.

## 4.4 Add Subcategory

Shopping Portal | Admin Admin 

Order Management
Manage users
Create Category
Sub Category
Insert Product
Manage Products
User Login Log
Logout

### Sub Category

Category

SubCategory Name

Create

### Sub Category

Show  entries Search:



#	Category	Description	Creation date	Last Updated	Action
1	BEAUTY	skin care	2019-11-19 12:30:08		 

Fig 4.4: Description of creating subcategory.



## 4.5 Insert Product

Shopping Portal | Admin Admin

- Order Management
- Manage users
- Create Category
- Sub Category
- Insert Product**
- Manage Products
- User Login Log
- Logout

### Insert Product

Category

Sub Category

Product Name

Product Company

Product Price Before Discount

Product Price After Discount(Selling Price)

Product Description 

**B** **I** **U**

Font Format Font Size... Font Family...

Fig 4.5: Description of insert product.

## 4.6 Dashboard

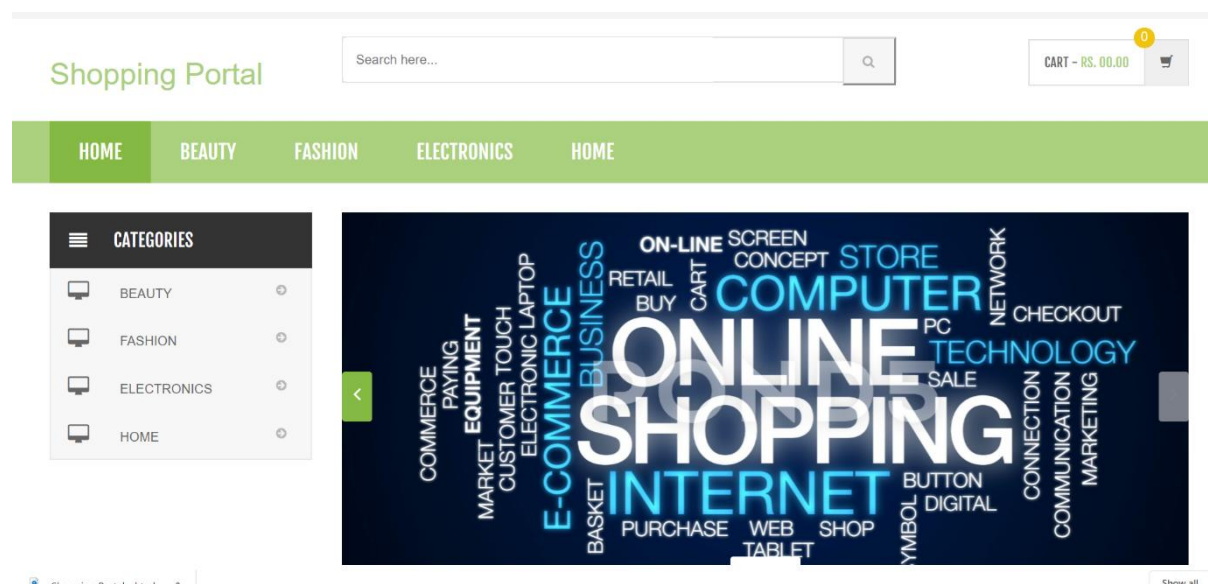


Fig 4.6: Dashboard.

## 4.7 Category

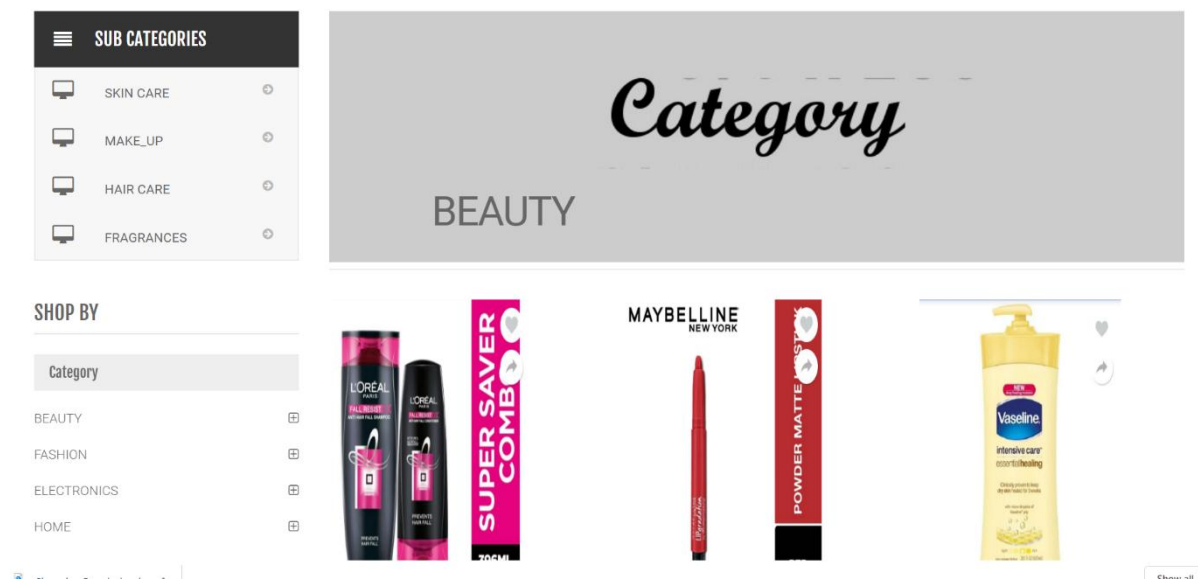


Fig 4.7: Category.

## 4.8 Subcategory

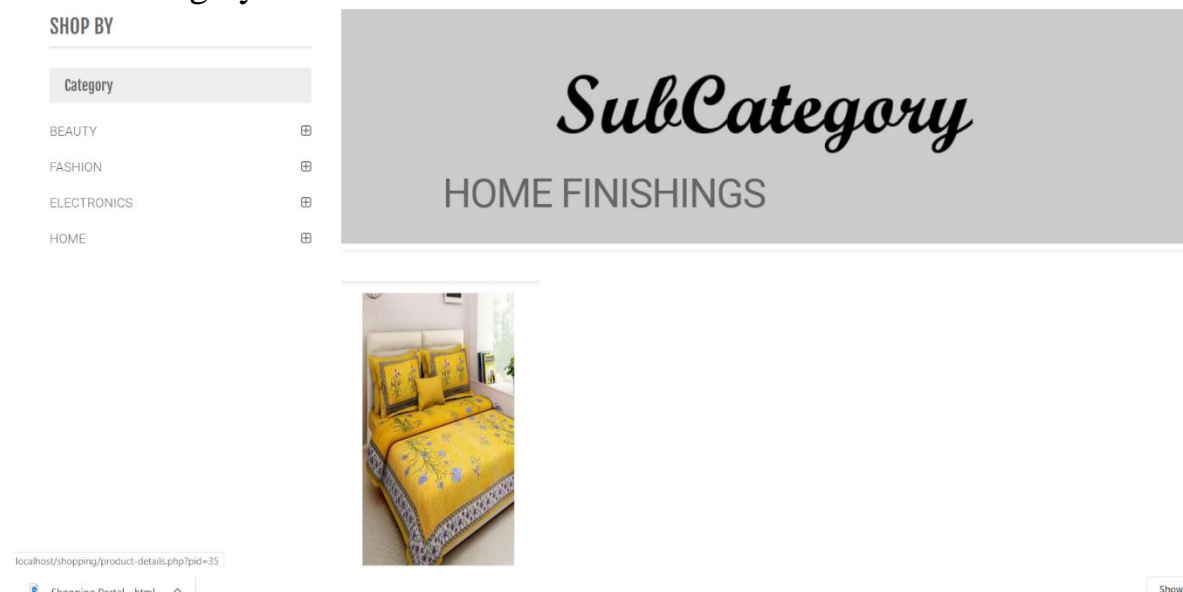




Fig 4.8: Subcategory.

## 4.9 Cart

Remove	Image	Product Name	Quantity	Price Per unit	Shipping Charge	Grandtotal
		DELL INSPIRON 14 5000 SERIES CORE I3 8TH GEN ★★★★☆ (0 Reviews)	1	RS 45990.00	RS 100.00	46090.00
<a href="#">CONTINUE SHOPPING</a>		<a href="#">UPDATE SHOPPING CART</a>				

**SHIPPING ADDRESS**  
 biet college  
 davangere

**BILLING ADDRESS**  
 biet college  
 davangere

**GRAND TOTAL** 46090.00  
[PROCEED TO CHECKOUT](#)

Fig 4.9: Cart.

## 4.10 Payment

[HOME](#)
[BEAUTY](#)
[FASHION](#)
[ELECTRONICS](#)
[HOME](#)

[Home](#) / [Payment Method](#)

### CHOOSE PAYMENT METHOD

**SELECT YOUR PAYMENT METHOD**

☒ COD
 ☐ Internet Banking
 ☐ Debit / Credit card

[submit](#)

Fig 4.10: Payment type

## 4.11 Track

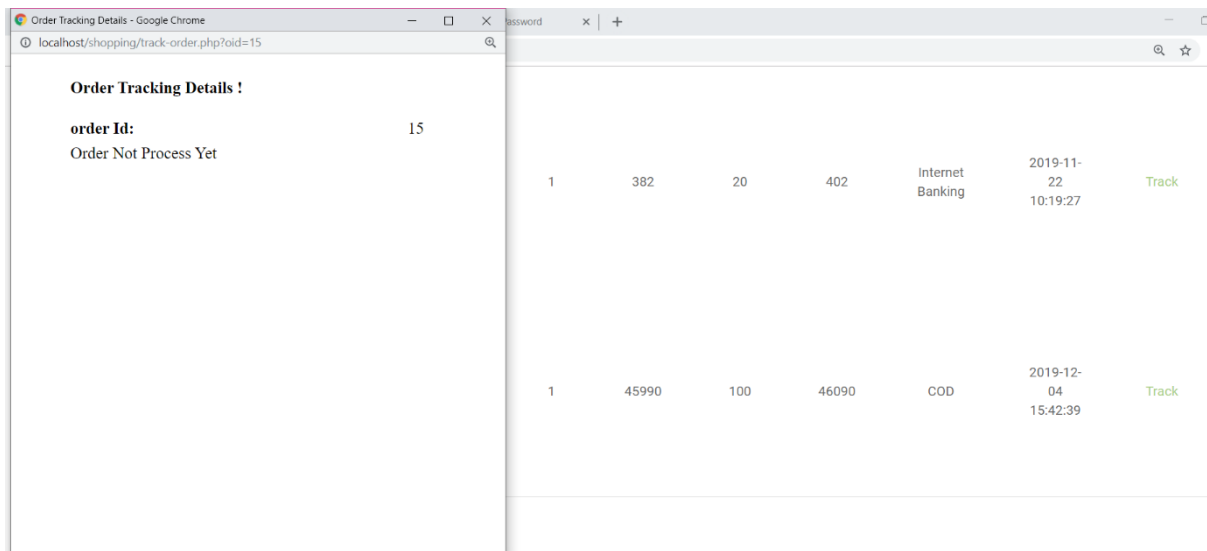


Fig 4.11: Track.

## 4.12 Update Track Information

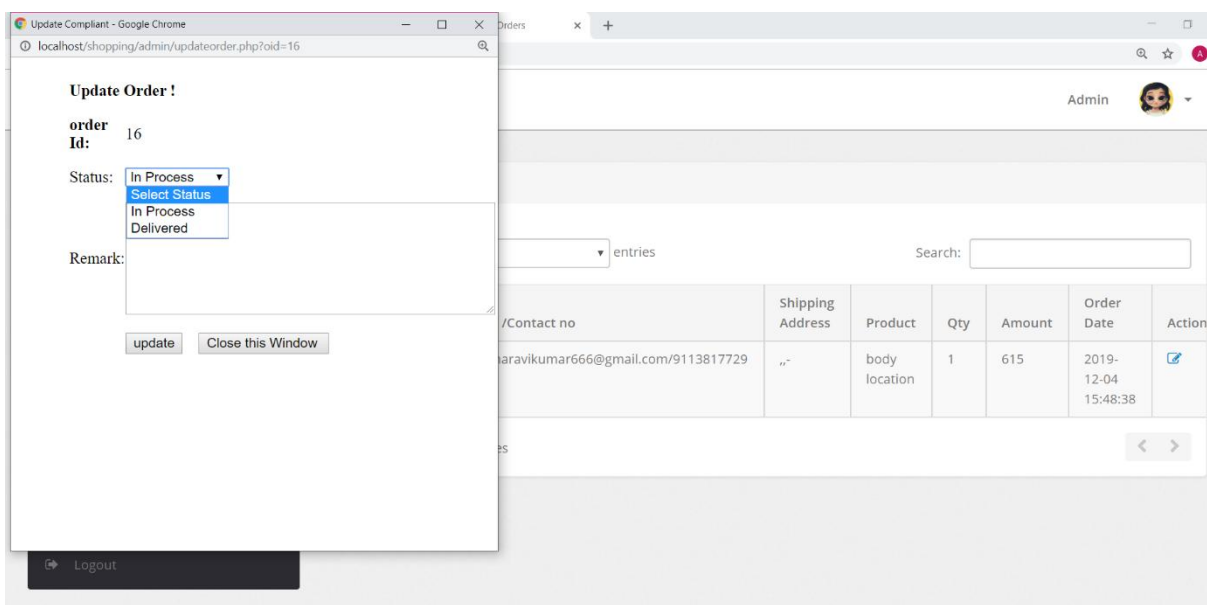


Fig 4.12: Update track information.

## **CHAPTER 5**

### **CONCLUSION**

In this project we have provided an easy and quick method to order products between seller and customer to view the transactions in a single page.

In future online bookings could have filtering options to filter by price and categories of products and subcategories of products. Another addition would be a convenient way for the admin to view all the user details.

## **BIBLIOGRAPHY**

### **Books:**

- Fundamentals of Database Systems, Ramez Elmasri and Shamkant B. Navathe, 7th Edition, 2017, Pearson.
- Database management systems, Ramakrishnan, and Gehrke, 3rd Edition, 2014, McGraw Hill.
- Silberschatz Korth and Sudharshan, Database System Concepts, 6th Edition, McGrawHill, 2013.

### **List of Websites:**

- <https://en.wikipedia.org/wiki/Database/>
- <https://www.w3schools.com/php/>