VISVESVARAYA TECHNOLOGICAL UNIVERSITY
BELAGAVI-590018

"A MINI PROJECT REPORT"
(Subject Code:21CSL55)
ON

"Online Shopping Management"

Submitted in partial fulfillment for the requirements for the Award of Degree of

BACHELOR OF ENGINEERING
IN
COMPUTER SCIENCE AND ENGINEERING
BY

INDERBIR SINGH [**1EP21CS032**]
ASHIRBAD SAI [**1EP21CS012**]
NITIN KRIPLANI [**1EP21CS064**]
ANSHUMAN KUMAR GAURAV [**1EP21CS011**]

UNDER THE GUIDANCE OF
Prof. Mrs Madhushree
Assistant Professor
DEPT. Of CSE, EPCET

Department of Computer Science and Engineering
EAST POINT COLLEGE OF ENGINEERING AND TECHNOLOGY
Bidarahalli, Bengaluru – 560 049
2023-2024

EAST POINT COLLEGE OF ENGINEERING AND TECHNOLOGY

(Affiliated to Visvesvaraya Technological University, Belgavi)

Bangalore-560049

EAST POINT COLLEGE OF ENGINEERING & TECHNOLOGY

DEPARTMENT OF COMPUTER SCIENCE & ENGINEERING

CERTIFICATE

This is to certify that the **Project Work** entitled **"Online Shopping Management "** is a bonafied work carried out by Inderbir Singh **bearing USN 1EP21CS032, Ashirbad Sai bearing USN 1EP21CS012, Nitin Kriplani bearing USN 1EP21CS064, Anshuman Kumar Gaurav bearing USN 1EP21CS011** in partial fulfillment for the award of **Bachelor of Engineering in Computer Science and Engineering** under **Visvesvaraya Technological University, Belgaum** during the year **2023-2024**. It is certified that all the corrections/suggestions indicated in the Internal Assessment have been incorporated in the report and submitted in the department library. This seminar report has been approved as it satisfies the academic requirements in respect of Mini Project Work prescribed for the award of the said degree.

GUIDE	**HOD**	**PRINCIPAL**
Prof. Mrs Madhshree	**Dr. Manimozi Iyer**	**Dr. Mrityunjaye V Latte**
Assistant Professor	Head of the Department	Principal

Examiners

Name of the Examiners **Signature with date**

1.Prof. Pavan Kumar

ACKNOWLEDGEMENT

Any achievement, be it scholastic or otherwise does not depend solely on the individual efforts but on the guidance, encouragement and cooperation of intellectuals, elders and friends. We would like to take this opportunity to thank them all.

First and foremost we would like to express our sincere regards and thanks to **Mr. Promod Gowda** and **Mr. Rajiv Gowda,** CEOs East Point Group of Institutions, Bangalore, for providing necessary infrastructure and creating good environment.

We express our gratitude to **Dr. Mrityunjaye V Latte,** Principal, EPCET who has always been a great source of inspiration.

We express our sincere regards and thanks to Dr. Manimozi Iyer **,** Professor and Head, Department of Computer Science and Engineering, EPCET, Bangalore, for his encouragement and support.

We are grateful to acknowledge the guidance and encouragement given to us by Prof. Swathi Pandey , Assistant Professors, Department of Computer Science and Engineering, EPCET, Bangalore, who has rendered a valuable assistance.

We also extend our thanks to all the faculties of the **Department of Computer Science and Engineering, EPCET**, Bangalore, who have encouraged us throughout the course of the Mini Project Work.

Last, but not the least, we would like to thank our family and friends for their inputs to improve the Mimi Project works.

Name: Ashirbad Sai	**USN: 1EP21CS012**
Name: Inderbir Singh	**USN: 1EP21CS032**
Name: Nitin Kriplani	**USN: 1EP21CS064**
Name: Anshuman Kumar Gaurav	**USN: 1EP21CS011**

ABSTRACT

The main aim of the project is the management of the database of *ONLINE SHOPPING MANAGEMENT*.

The Online Shopping Management DBMS mini project aims to develop a streamlined database system for efficient handling of online shopping activities. It includes user registration, product catalog management, order processing, inventory tracking, payment integration, and reporting features. Utilizing SQL for database operations, the system ensures data integrity and reliability. Designed to enhance user experience and facilitate effective management, it provides a scalable solution for online shopping platforms.

The MYSQL database is used as a platform along with PHP and WAMP Server support. Application and the GUI are developed in HTML5, CSS3 using PHP and WAMP Server.

Overall this Art Gallery Management System is used to manage most art-related activities like exhibitions, gallery management, art stocks etc. in the gallery.

CHAPTER 1

INTRODUCTION

• INTRODUCTION TO SQL

SQL which is an abbreviation for **Structured Query Language** is a language to request data from a database, to add, update, or remove data within a database, or to manipulate the metadata of the database.

Sometimes SQL is characterized as *non-procedural* because procedural languages generally require the details of the operations to be specified, such as opening and closing tables, loading and searching indexes, or flushing buffers and writing data to file systems. Therefore, SQL is designed at a higher conceptual level of operation than procedural languages.

Commonly used statements are grouped into the following categories

Data Query Language (DQL)

SELECT-Used to retrieve certain records from one or more tables.

Data Manipulation Language (DML)

INSERT - Used to create a record

UPDATE - Used to change certain records.

DELETE - Used to delete certain records.

Data Definition Language (DDL)

CREATE - Used to create a new table, a view of a table, or other object in database.

ALTER - Used to modify an existing database object, such as a table.

DROP - Used to delete an entire table, a view of a table or other object in the database.

Data Control Language (DCL)

GRANT - Used to give a privilege to someone

REVOKE - Used to take back privileges granted to someone.

1.2 INTRODUCTION TO FRONT END SOFTWARE

The "front end languages" live in the browser. After you type in an address in the address bar at the top and hit Enter, your browser will receive an at least an HTML file from the web server.

Each of these languages performs a separate but very important function but the work harmoniously together to determine how the web page is STRUCTURED(HTML), how it LOOKS(CSS), and how its FUNCTIONS (JavaScript).

Front end web development is NOT design (You won't be playing around in Photoshop or anything), but a *front-end developer* does apply the work of designers to the web page by translating their well-designed layouts into real code. The front-end developer stands between the designer on one end and the back-end developer on the other, translating the design into code and plugging the data from the back-end developer into the right spots.

PHP is a server-side scripting language designed primarily for web development but also used as a general-purpose programming language. Originally created by Rasmus Lerdorf in 1994, the PHP reference implementation is now produced by The PHP Development Team.

PHP code may be embedded into HTML or it can be used in combination with various web template systems, web content management systems and web frameworks. PHP code is usually processed by a PHP interpreter implemented as a module in the web server or as a Common Gateway Interface (CGI) executable. The web server software combines the result of the interpreted and executed PHP code, which may be any type of data, including images, with the generated web page. PHP code may also be executed with a command-line interface (CLI) and it can be used to implement stand-alone graphical applications.

The standard PHP interpreter, powered by the Zend Engine, is free to use software released under the PHP License. PHP code is usually processed by a PHP interpreter implemented as a module in the web server or as Common Gateway Interface(CGI) executable. PHP has been widely ported on web servers on almost every operating system and platform, free of charge.

CHAPTER 2

REQUIREMENTS SPECIFICATION

2.1 SOFTWARE REQUIREMENTS

Operating System	:	64bit WINDOWS Operating System, X64-based processor
Database	:	MYSQL
Scripting Language	:	HTML5, CSS3, PHP
Server	:	WAMP

2.2 HARDWARE REQUIREMENTS

Processor	:	Intel Celeron CPU N3060 @1.60GHz or Above
RAM	:	4.00 GB or Above
Hard Disk	:	1 TB
Compact Disk	:	CD-ROM, CD-R, CD-RW
Input device	:	Keyboard

CHAPTER 3

OBJECTIVE OF THE PROJECT

The main objective of creating an Online shopping database project is

- To manage the details of gallery, exhibition, artwork and artist. It manages all the sales and inventory in the gallery. The purpose of the project is to build and application program to reduce the manual work.

- To tracks all the details about the sales of the artwork, the customer that bought it, etc. It manages the information about the artwork. Provides an information and description of the artworks left, thereby increasing the efficiency of managing the gallery. The organisation can maintain a computerized record of the artwork present in the gallery.

- To helps in the utilization of the resources in an effective manner. It maintains a list of all the customers and the various artwork that they have bought and the money that have invested in each.

- To maintains the record of exhibitions and various sales made during it. The objective of developing such computerized system is to reduce the paper work and safe of time in art gallery database management, thereby increasing the efficiency and decreasing the work load.

- To develop such computerized system is to reduce the paper work and safe of time in art gallery database management, thereby increasing the efficiency and decreasing the work load.

CHAPTER 4

IMPLEMENTATION

4.1 ER DIAGRAM

- An **entity-relationship model (ER Model)** describes inter-related things of interest in a specific domain of knowledge. An ER model is composed of entity types (which classify the things of interest) and specifies relationships that can exist between instances of those entity types.

- An entity may be defined as a thing capable of an independent existence that can be uniquely identified. An entity is an abstraction from the complexities of a domain.

- Attributes are drawn as ovals and are connected with a line to exactly one entity or relationship set.

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

- Cardinality constraints are expressed as follows:

  - A double line indicates a participation constraint, totality or subjectivity: all entities in the entity set must participate in at least one relationship in the relationship set.

  - An arrow from entity set to relationship set indicates a key constraint, i.e. injectivity: each entity of the entity set can participate in at most one relationship in the relationship set.

  - A thick line indicates both, i.e. bijectivity: each entity in the entity set is involved in exactly one relationship.

  - An underlined name of an attribute indicates that it is a key: two different entities or relationships with this attribute always have different values for this attribute.

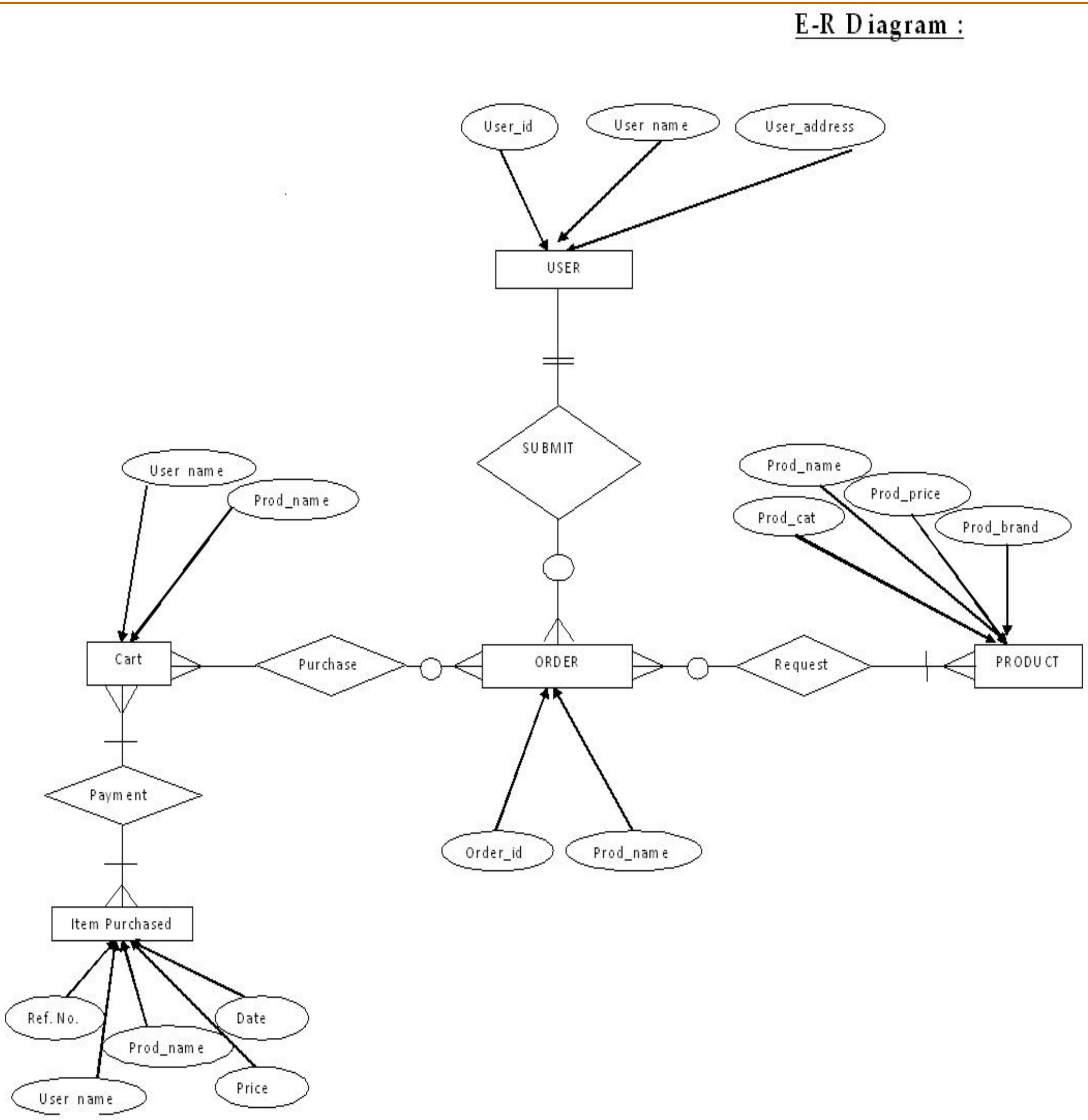


FIGURE 4.1: ER DIAGRAM of ART GALLERY DATABASE

4.2 MAPPING OF ER DIAGRAM TO RELATIONS

STEP 1: Mapping of Regular Entities

For each regular entity type E in the ER schema, create relation R that includes all simple attributes of E.

USER

UID	UNAME	ADDRESS

CART

UID	UNAME	PRODUCTNAME

PRODUCT

PID	PNAME	PRICE	PBRAND	PCAT

CUSTOMER

CUSTID	**PID**	FNAME1	LNAME1	ADDRESS	PHONE	DOB
	FK					

STEP 2 : Mapping of Weak Entity Types

ITEM_PURCHASED

ITEMNO	**CUSTID**	PNAME	DATE	REF_NO	PRICE
	FK				

STEP 3: Mapping of 1:1 Relationship

Identify the relation S that represents the participating entity type at the 1-side of the relationship type.

Include as foreign key in S the primary key of the relations T that represents the other entity type participating in R.

For each binary 1:1 relationship type R in ER schema, identify the relations S and T that correspond to the entity types participating in R if any.

There are **no** 1:1 relationship.

STEP 4 : Mapping of 1:N Relationship

CART

UID	STARTDATE	ENDDATE	**PID**
			FK

CUSTOMER

CUSTID	FNAME	LNAME	BIRTHPLACE	STYLE	**PID**	**REf_ID**	**CUSTID**
					FK	**FK**	**FK**

ITEM_PURCHASED

CUSTID	**PID**	DATE	PRICE	ADDRESS	CART_NO	**REF_ID**
	FK					**FK**

STEP 5 : Mapping of M:N Relationship

Create a new relation S to represent R.

Include as foreign key attributes in S the primary key of the relations that represents the participating entity types their combination will form the primary key of S.

Also, include any simple attributes of the M:N relationship type as attributes of S.

STEP 6: Mapping of Multi-Valued Attributes

For each multivalued attributes 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 of relationship type that has A as an attribute.

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

CONTACTS

CUSTID	PHONE

STEP 7: Mapping of N-Ary Relationship Types

For each n-ary relationship type R, where n>2 create a new relationship S to represent R. λ include as foreign key attributes in S the primary keys of the relations that represent the participating entity types.

λ also includes any simple attributes of the n-ary relationship type (or simple components of composite attributes) as attributes of S.

There are **no** n-ary relationship types.

4.3 SCHEMA DIAGRAM

USER

USERID	USER_NAME	ADDRESS

PRODUCT

PID	PROD_NAME	PRICE	**BRAND**

CUSTOMER

CUSTID	**PID**	FNAME1	LNAME1	ADDRESS	DOB	**CARTID**

ORDER

PID	DATE	AMOUNT	CUST_ID	REF_NO			**ORD_ID**

ITEM_PURCHASED

PID	**CUSTID**	NAME	USERNAME	DATE	PRICE	**REFNO**	**GID**

CART

USERNAME	PRODUCTNAME

FIGURE 4.3: SCHEMA DIAGRAM

4.3 NORMALIZE THE RELATIONS

Database normalization, or simply normalization, is the process of organizing the columns(attributes) and tables(relations) of a relational database to reduce data redundancy and improve data integrity. Normalization involves arranging attributes in relations based on dependencies between attributes.

- ## First Normal Form

As per First normal form, no two rows of data must contain repeating group of information. Each set of columns must have a unique value, such that multiple columns cannot be used to fetch the same row. Each table should be organized into rows, and each row should have a primary key that will distinguishes it as unique.

Example:

USER

USERID	UNAME	ADDRESS

All the tables in the database are normalized to 1NF as all the attributes are atomic.

- ## Second Normal Form (2NF)

A table is in 2NF if it is in 1NF and if all non-key attributes are fully functionally dependent on all of the key.

Example:

CUSTOMER

CUSTID	**CARTID**	FNAME1	LNAME1	ADDRESS	DOB	**PID**

FD1

FD1

CUSTID	FNAME1	LNAME1	DOB

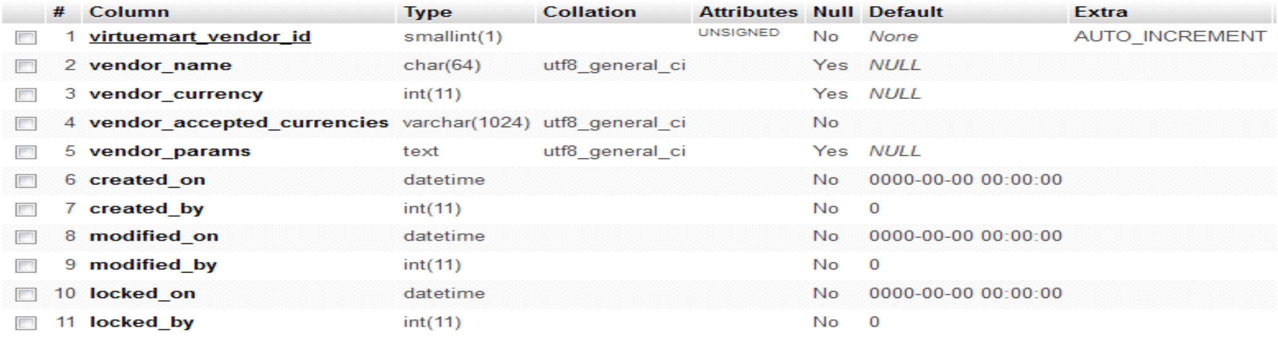
- ## Third Normal Form(3NF):

A table is in 3NF if it is in 2NF and if it has no transitive dependency. X->Y, Y->Z, X>Z

According to CODD’s definition a relation schema R is in 3NF. It satisfies 2NF and no non-prime attribute of R is transitively dependent on the primary key. All tables of database satisfies upto 3NF.

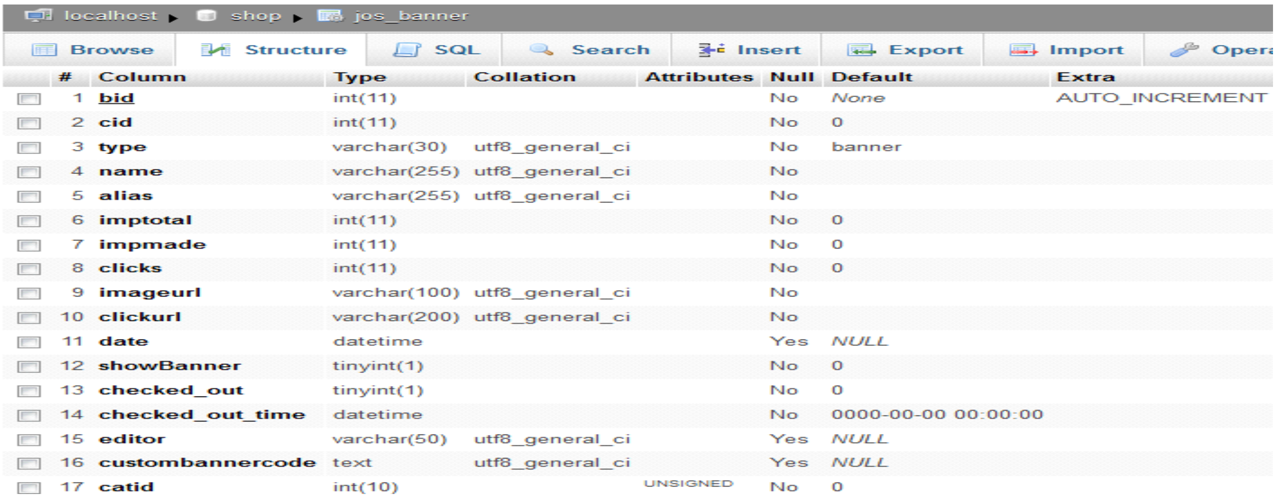
4.5 CREATION OF TABLES

1. CREATING VENDORS TABLE



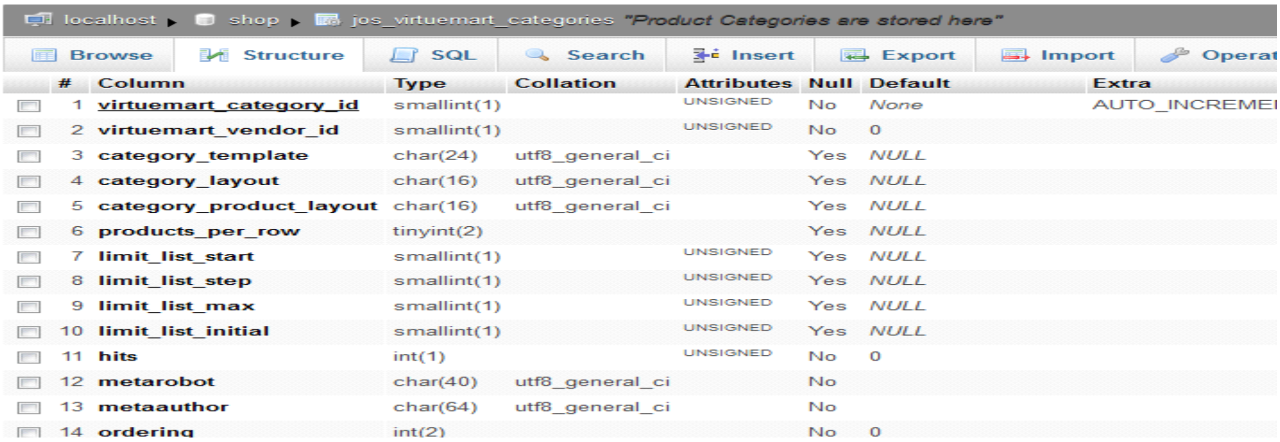
	#	Column	Type	Collation	Attributes	Null	Default	Extra
□	1	virtuemart_vendor_id	smallint(1)		UNSIGNED	No	*None*	AUTO_INCREMENT
□	2	vendor_name	char(64)	utf8_general_ci		Yes	*NULL*	
□	3	vendor_currency	int(11)			Yes	*NULL*	
□	4	vendor_accepted_currencies	varchar(1024)	utf8_general_ci		No		
□	5	vendor_params	text	utf8_general_ci		Yes	*NULL*	
□	6	created_on	datetime			No	0000-00-00 00:00:00	
□	7	created_by	int(11)			No	0	
□	8	modified_on	datetime			No	0000-00-00 00:00:00	
□	9	modified_by	int(11)			No	0	
□	10	locked_on	datetime			No	0000-00-00 00:00:00	
□	11	locked_by	int(11)			No	0	

2. CREATE BANNER TABLE



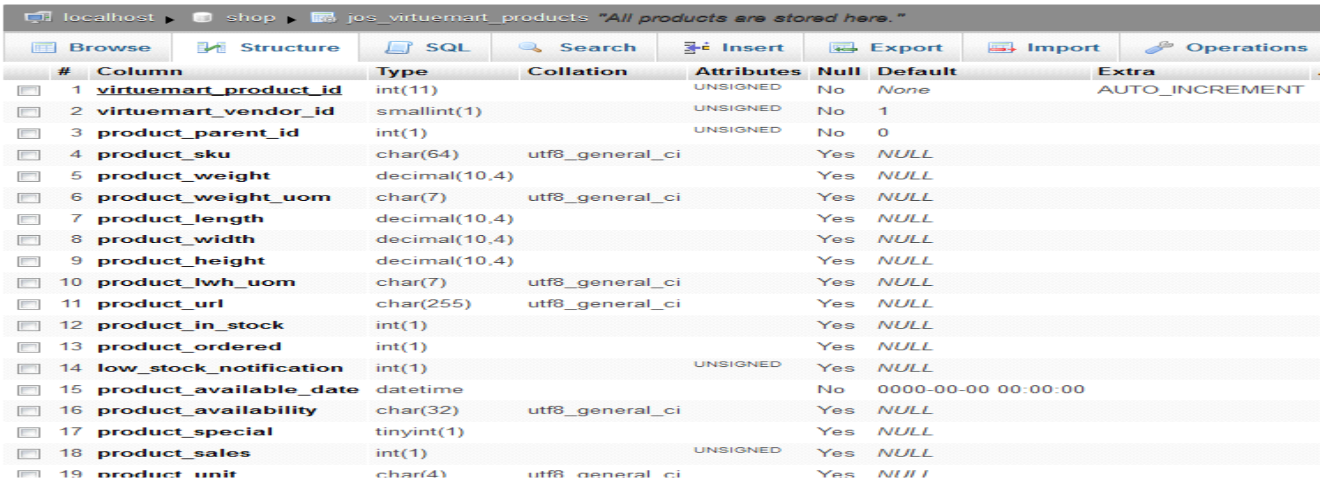
	#	Column	Type	Collation	Attributes	Null	Default	Extra
□	1	bid	int(11)			No	*None*	AUTO_INCREMENT
□	2	cid	int(11)			No	0	
□	3	type	varchar(30)	utf8_general_ci		No	banner	
□	4	name	varchar(255)	utf8_general_ci		No		
□	5	alias	varchar(255)	utf8_general_ci		No		
□	6	imptotal	int(11)			No	0	
□	7	impmade	int(11)			No	0	
□	8	clicks	int(11)			No	0	
□	9	imageurl	varchar(100)	utf8_general_ci		No		
□	10	clickurl	varchar(200)	utf8_general_ci		No		
□	11	date	datetime			Yes	*NULL*	
□	12	showBanner	tinyint(1)			No	0	
□	13	checked_out	tinyint(1)			No	0	
□	14	checked_out_time	datetime			No	0000-00-00 00:00:00	
□	15	editor	varchar(50)	utf8_general_ci		Yes	*NULL*	
□	16	custombannercode	text	utf8_general_ci		Yes	*NULL*	
□	17	catid	int(10)		UNSIGNED	No	0	

3. CREATE CATAGORY TABLE


localhost ▸ shop ▸ jos_virtuemart_categories "Product Categories are stored here"
Browse
Structure
SQL
Search
Insert
Export
Import
Operat

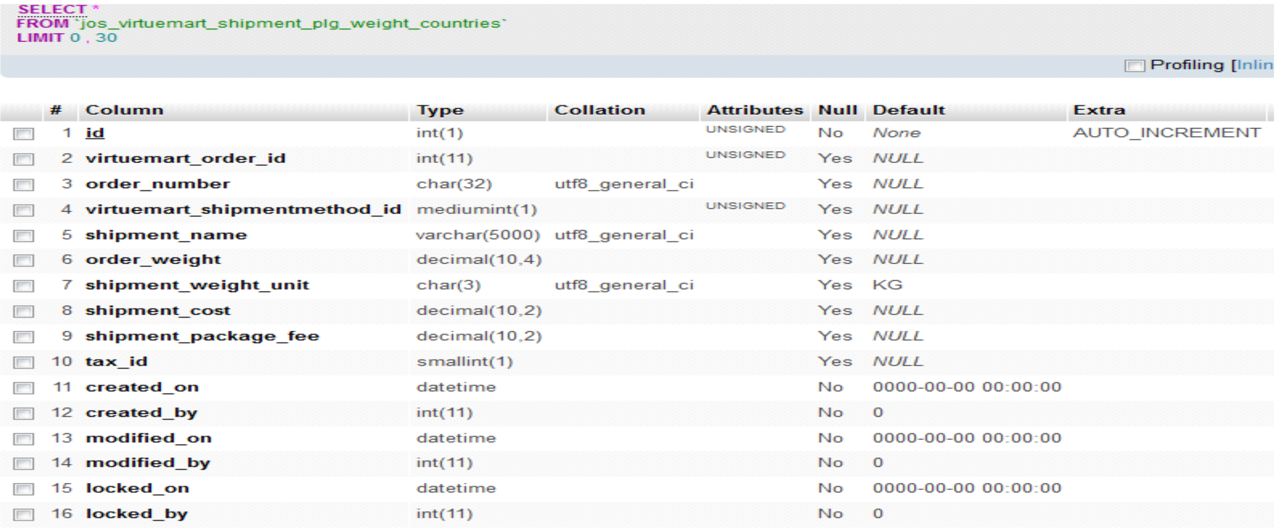
	#	Column	Type	Collation	Attributes	Null	Default	Extra
□	1	virtuemart_category_id	smallint(1)		UNSIGNED	No	*None*	AUTO_INCREMEI
□	2	virtuemart_vendor_id	smallint(1)		UNSIGNED	No	0	
□	3	category_template	char(24)	utf8_general_ci		Yes	*NULL*	
□	4	category_layout	char(16)	utf8_general_ci		Yes	*NULL*	
□	5	category_product_layout	char(16)	utf8_general_ci		Yes	*NULL*	
□	6	products_per_row	tinyint(2)			Yes	*NULL*	
□	7	limit_list_start	smallint(1)		UNSIGNED	Yes	*NULL*	
□	8	limit_list_step	smallint(1)		UNSIGNED	Yes	*NULL*	
□	9	limit_list_max	smallint(1)		UNSIGNED	Yes	*NULL*	
□	10	limit_list_initial	smallint(1)		UNSIGNED	Yes	*NULL*	
□	11	hits	int(1)		UNSIGNED	No	0	
□	12	metarobot	char(40)	utf8_general_ci		No		
□	13	metaauthor	char(64)	utf8_general_ci		No		
□	14	ordering	int(2)			No	0	

4. CREATE CUSTOMER TABLE


localhost ▸ shop ▸ jos_virtuemart_products "All products are stored here."
Browse
Structure
SQL
Search
Insert
Export
Import
Operations

	#	Column	Type	Collation	Attributes	Null	Default	Extra
□	1	virtuemart_product_id	int(11)		UNSIGNED	No	None	AUTO_INCREMENT
□	2	virtuemart_vendor_id	smallint(1)		UNSIGNED	No	1	
□	3	product_parent_id	int(1)		UNSIGNED	No	0	
□	4	product_sku	char(64)	utf8_general_ci		Yes	NULL	
□	5	product_weight	decimal(10,4)			Yes	NULL	
□	6	product_weight_uom	char(7)	utf8_general_ci		Yes	NULL	
□	7	product_length	decimal(10,4)			Yes	NULL	
□	8	product_width	decimal(10,4)			Yes	NULL	
□	9	product_height	decimal(10,4)			Yes	NULL	
□	10	product_lwh_uom	char(7)	utf8_general_ci		Yes	NULL	
□	11	product_url	char(255)	utf8_general_ci		Yes	NULL	
□	12	product_in_stock	int(1)			Yes	NULL	
□	13	product_ordered	int(1)			Yes	NULL	
□	14	low_stock_notification	int(1)		UNSIGNED	Yes	NULL	
□	15	product_available_date	datetime			No	0000-00-00 00:00:00	
□	16	product_availability	char(32)	utf8_general_ci		Yes	NULL	
□	17	product_special	tinyint(1)			Yes	NULL	
□	18	product_sales	int(1)		UNSIGNED	Yes	NULL	
□	19	product_unit	char(4)	utf8_general_ci		Yes	NULL	

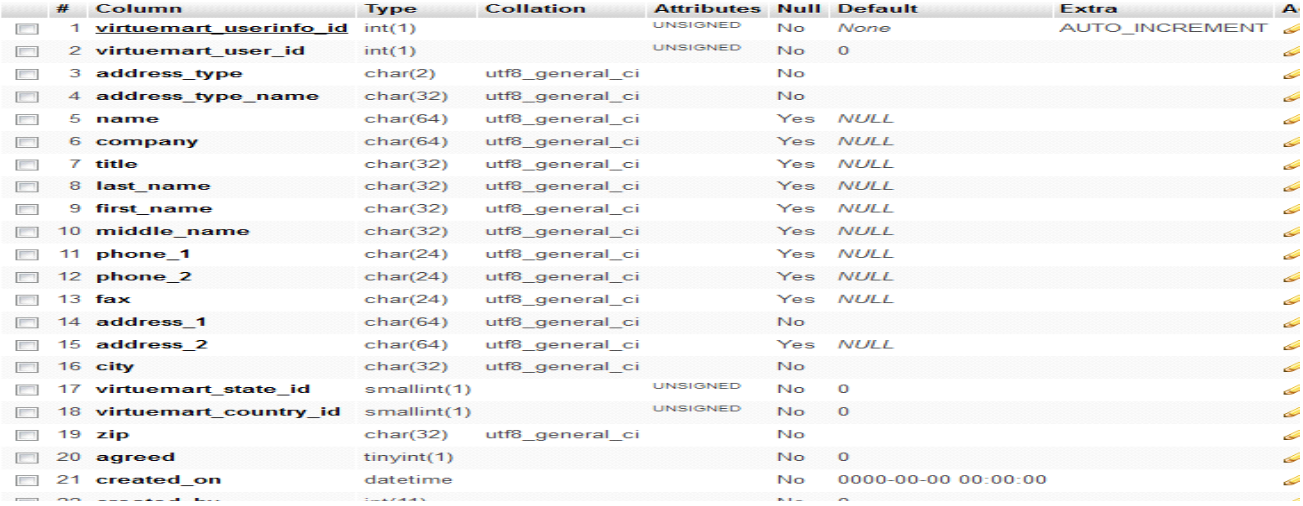
5. CREATE SHIPMENT TABLE



```
SELECT *
FROM `jos_virtuemart_shipment_plg_weight_countries`
LIMIT 0 , 30
```

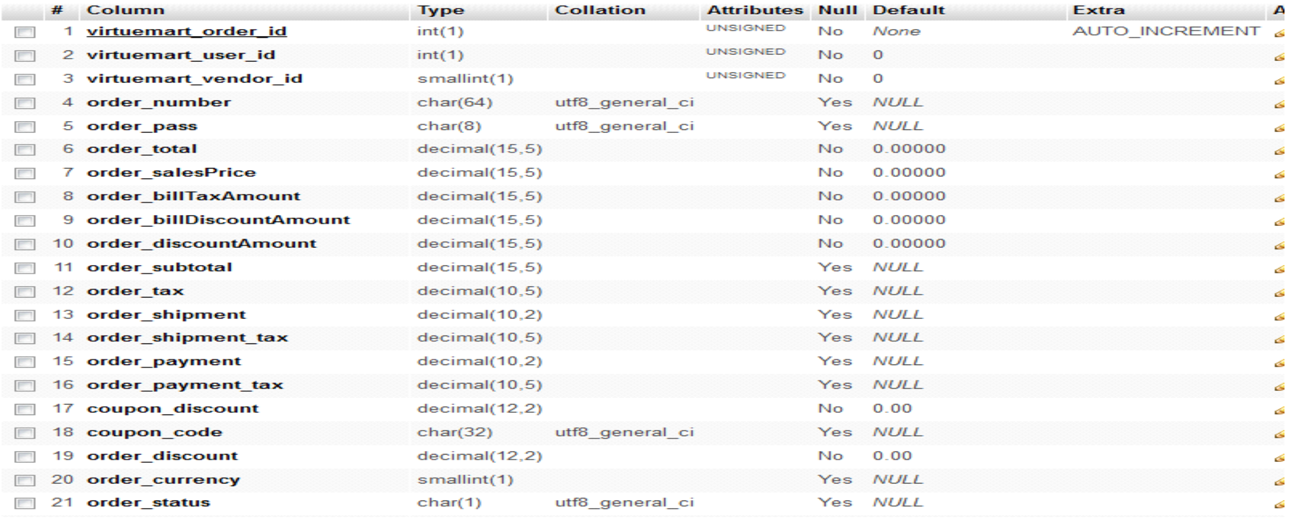
	#	Column	Type	Collation	Attributes	Null	Default	Extra
□	1	id	int(1)		UNSIGNED	No	None	AUTO_INCREMENT
□	2	virtuemart_order_id	int(11)		UNSIGNED	Yes	NULL	
□	3	order_number	char(32)	utf8_general_ci		Yes	NULL	
□	4	virtuemart_shipmentmethod_id	mediumint(1)		UNSIGNED	Yes	NULL	
□	5	shipment_name	varchar(5000)	utf8_general_ci		Yes	NULL	
□	6	order_weight	decimal(10,4)			Yes	NULL	
□	7	shipment_weight_unit	char(3)	utf8_general_ci		Yes	KG	
□	8	shipment_cost	decimal(10,2)			Yes	NULL	
□	9	shipment_package_fee	decimal(10,2)			Yes	NULL	
□	10	tax_id	smallint(1)			Yes	NULL	
□	11	created_on	datetime			No	0000-00-00 00:00:00	
□	12	created_by	int(11)			No	0	
□	13	modified_on	datetime			No	0000-00-00 00:00:00	
□	14	modified_by	int(11)			No	0	
□	15	locked_on	datetime			No	0000-00-00 00:00:00	
□	16	locked_by	int(11)			No	0	

6. CREATE USER INFORMATION TABLE



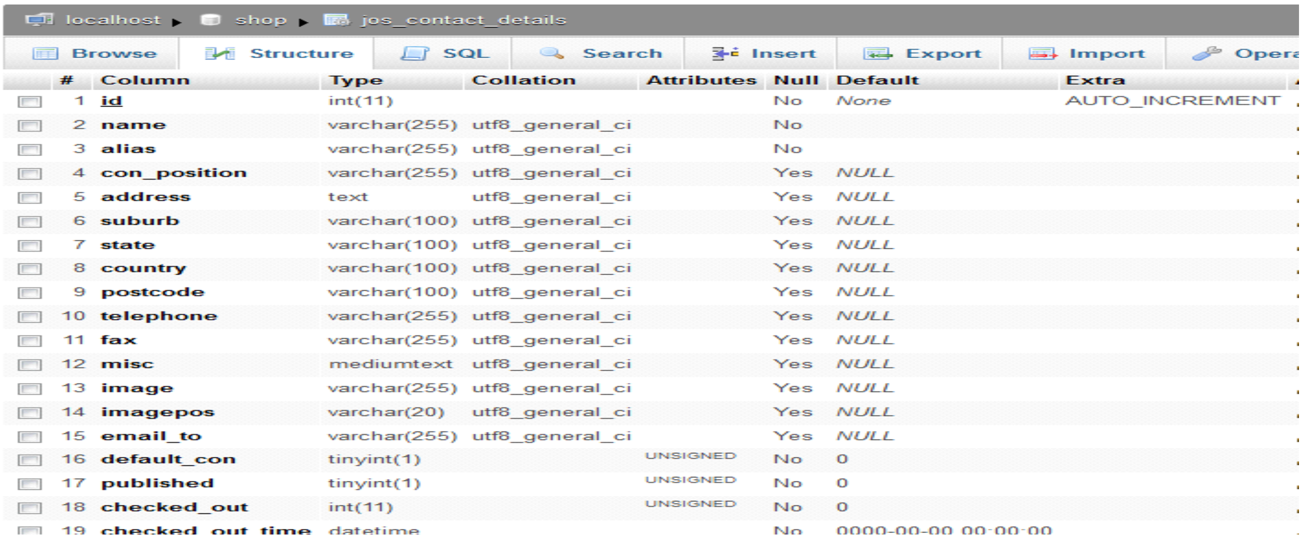
	#	Column	Type	Collation	Attributes	Null	Default	Extra	A
□	1	virtuemart_userinfo_id	int(1)		UNSIGNED	No	None	AUTO_INCREMENT	✎
□	2	virtuemart_user_id	int(1)		UNSIGNED	No	0		✎
□	3	address_type	char(2)	utf8_general_ci		No			✎
□	4	address_type_name	char(32)	utf8_general_ci		No			✎
□	5	name	char(64)	utf8_general_ci		Yes	NULL		✎
□	6	company	char(64)	utf8_general_ci		Yes	NULL		✎
□	7	title	char(32)	utf8_general_ci		Yes	NULL		✎
□	8	last_name	char(32)	utf8_general_ci		Yes	NULL		✎
□	9	first_name	char(32)	utf8_general_ci		Yes	NULL		✎
□	10	middle_name	char(32)	utf8_general_ci		Yes	NULL		✎
□	11	phone_1	char(24)	utf8_general_ci		Yes	NULL		✎
□	12	phone_2	char(24)	utf8_general_ci		Yes	NULL		✎
□	13	fax	char(24)	utf8_general_ci		Yes	NULL		✎
□	14	address_1	char(64)	utf8_general_ci		No			✎
□	15	address_2	char(64)	utf8_general_ci		Yes	NULL		✎
□	16	city	char(32)	utf8_general_ci		No			✎
□	17	virtuemart_state_id	smallint(1)		UNSIGNED	No	0		✎
□	18	virtuemart_country_id	smallint(1)		UNSIGNED	No	0		✎
□	19	zip	char(32)	utf8_general_ci		No			✎
□	20	agreed	tinyint(1)			No	0		✎
□	21	created_on	datetime			No	0000-00-00 00:00:00		✎
□	22	created_by	int(11)			No	0		✎

7. ORDER TABLE



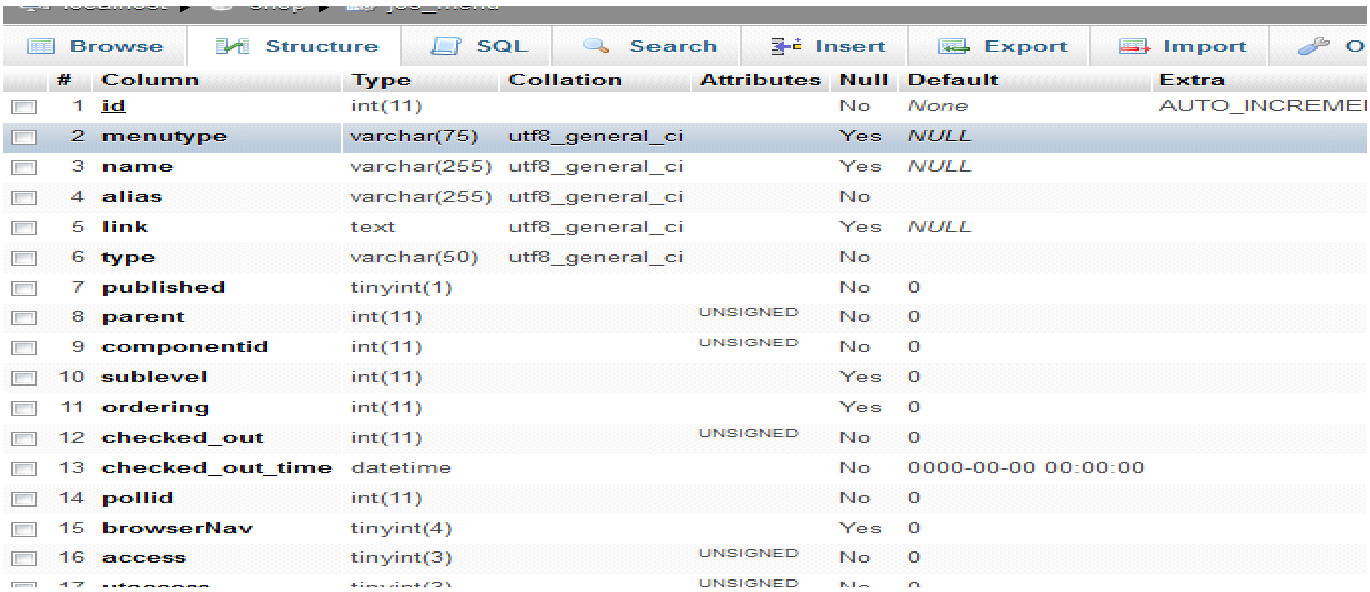
	#	Column	Type	Collation	Attributes	Null	Default	Extra	A
□	1	virtuemart_order_id	int(1)		UNSIGNED	No	*None*	AUTO_INCREMENT	
□	2	virtuemart_user_id	int(1)		UNSIGNED	No	0		
□	3	virtuemart_vendor_id	smallint(1)		UNSIGNED	No	0		
□	4	order_number	char(64)	utf8_general_ci		Yes	*NULL*		
□	5	order_pass	char(8)	utf8_general_ci		Yes	*NULL*		
□	6	order_total	decimal(15,5)			No	0.00000		
□	7	order_salesPrice	decimal(15,5)			No	0.00000		
□	8	order_billTaxAmount	decimal(15,5)			No	0.00000		
□	9	order_billDiscountAmount	decimal(15,5)			No	0.00000		
□	10	order_discountAmount	decimal(15,5)			No	0.00000		
□	11	order_subtotal	decimal(15,5)			Yes	*NULL*		
□	12	order_tax	decimal(10,5)			Yes	*NULL*		
□	13	order_shipment	decimal(10,2)			Yes	*NULL*		
□	14	order_shipment_tax	decimal(10,5)			Yes	*NULL*		
□	15	order_payment	decimal(10,2)			Yes	*NULL*		
□	16	order_payment_tax	decimal(10,5)			Yes	*NULL*		
□	17	coupon_discount	decimal(12,2)			No	0.00		
□	18	coupon_code	char(32)	utf8_general_ci		Yes	*NULL*		
□	19	order_discount	decimal(12,2)			No	0.00		
□	20	order_currency	smallint(1)			Yes	*NULL*		
□	21	order_status	char(1)	utf8_general_ci		Yes	*NULL*		

8. CONTACT DETAILS TABLE



	#	Column	Type	Collation	Attributes	Null	Default	Extra
□	1	id	int(11)			No	*None*	AUTO_INCREMENT
□	2	name	varchar(255)	utf8_general_ci		No		
□	3	alias	varchar(255)	utf8_general_ci		No		
□	4	con_position	varchar(255)	utf8_general_ci		Yes	*NULL*	
□	5	address	text	utf8_general_ci		Yes	*NULL*	
□	6	suburb	varchar(100)	utf8_general_ci		Yes	*NULL*	
□	7	state	varchar(100)	utf8_general_ci		Yes	*NULL*	
□	8	country	varchar(100)	utf8_general_ci		Yes	*NULL*	
□	9	postcode	varchar(100)	utf8_general_ci		Yes	*NULL*	
□	10	telephone	varchar(255)	utf8_general_ci		Yes	*NULL*	
□	11	fax	varchar(255)	utf8_general_ci		Yes	*NULL*	
□	12	misc	mediumtext	utf8_general_ci		Yes	*NULL*	
□	13	image	varchar(255)	utf8_general_ci		Yes	*NULL*	
□	14	imagepos	varchar(20)	utf8_general_ci		Yes	*NULL*	
□	15	email_to	varchar(255)	utf8_general_ci		Yes	*NULL*	
□	16	default_con	tinyint(1)		UNSIGNED	No	0	
□	17	published	tinyint(1)		UNSIGNED	No	0	
□	18	checked_out	int(11)		UNSIGNED	No	0	
□	19	checked_out_time	datetime			No	0000-00-00 00:00:00	

9. MENU TABLE



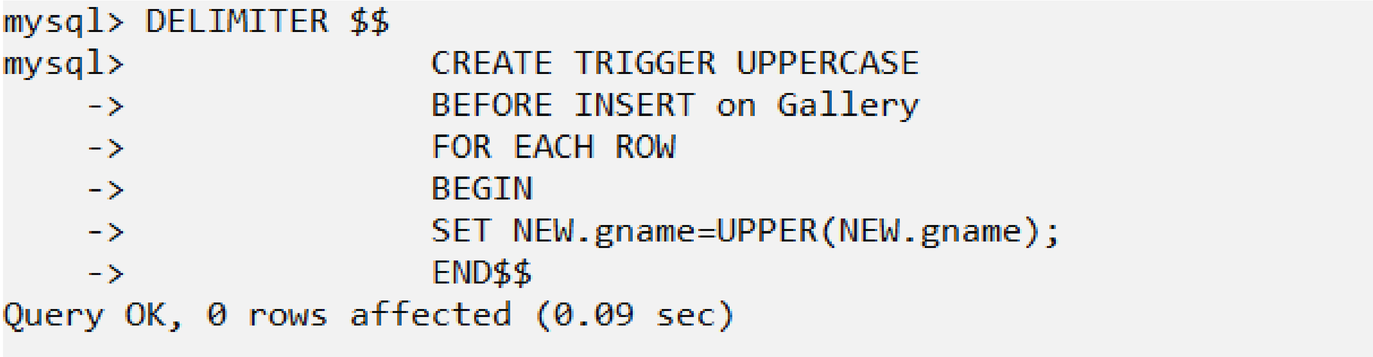
	#	Column	Type	Collation	Attributes	Null	Default	Extra
□	1	id	int(11)			No	None	AUTO_INCREMEI
□	2	menutype	varchar(75)	utf8_general_ci		Yes	NULL	
□	3	name	varchar(255)	utf8_general_ci		Yes	NULL	
□	4	alias	varchar(255)	utf8_general_ci		No		
□	5	link	text	utf8_general_ci		Yes	NULL	
□	6	type	varchar(50)	utf8_general_ci		No		
□	7	published	tinyint(1)			No	0	
□	8	parent	int(11)		UNSIGNED	No	0	
□	9	componentid	int(11)		UNSIGNED	No	0	
□	10	sublevel	int(11)			Yes	0	
□	11	ordering	int(11)			Yes	0	
□	12	checked_out	int(11)		UNSIGNED	No	0	
□	13	checked_out_time	datetime			No	0000-00-00 00:00:00	
□	14	pollid	int(11)			No	0	
□	15	browserNav	tinyint(4)			Yes	0	
□	16	access	tinyint(3)		UNSIGNED	No	0	
□	17	utaccess	tinyint(3)		UNSIGNED	No	0	

4.7 CREATION OF TRIGGERS

The trigger is made such that when a new record is inserted into a Gallery table, it automatically changes the lowercase name into uppercase in the backend.

TRIGGER ON GALLERY TABLE TO CHANGING NAME TO UPPERCASE

```
DELIMITER $$

CREATE TRIGGER UPPERCASE

BEFORE INSERT on Gallery

FOR EACH ROW

BEGIN

SET NEW.gname=UPPER(NEW.gname);

END$$
```



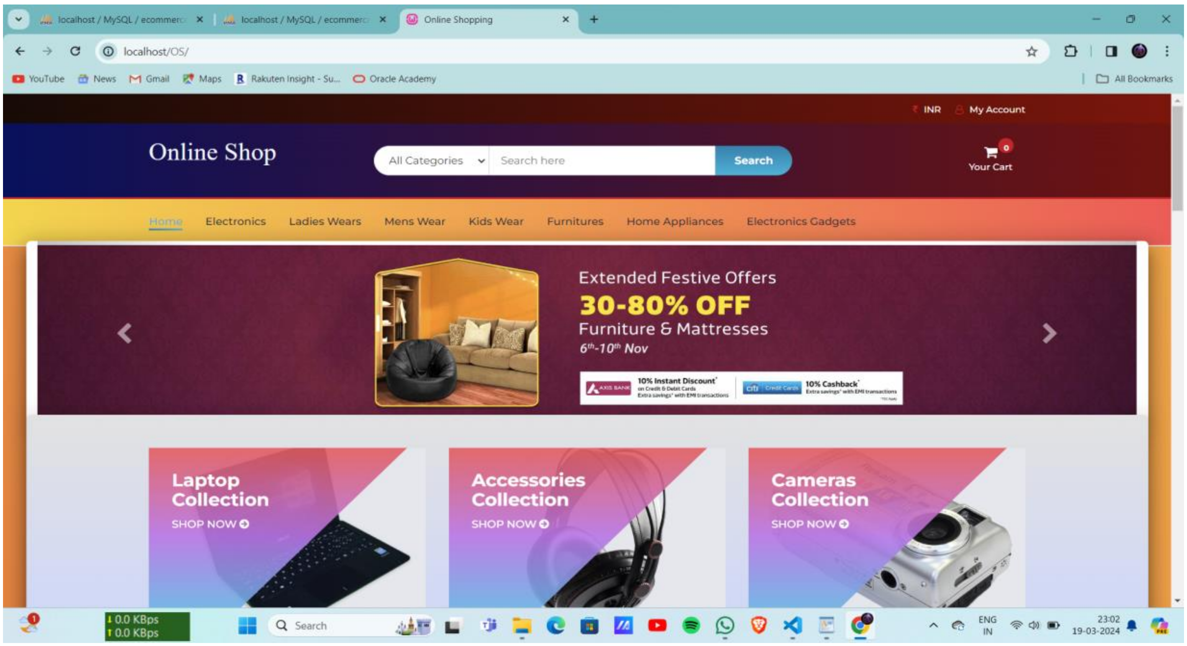
```
mysql> DELIMITER $$
mysql>                  CREATE TRIGGER UPPERCASE
    ->                  BEFORE INSERT on Gallery
    ->                  FOR EACH ROW
    ->                  BEGIN
    ->                  SET NEW.gname=UPPER(NEW.gname);
    ->                  END$$
Query OK, 0 rows affected (0.09 sec)
```

CHAPTER 5

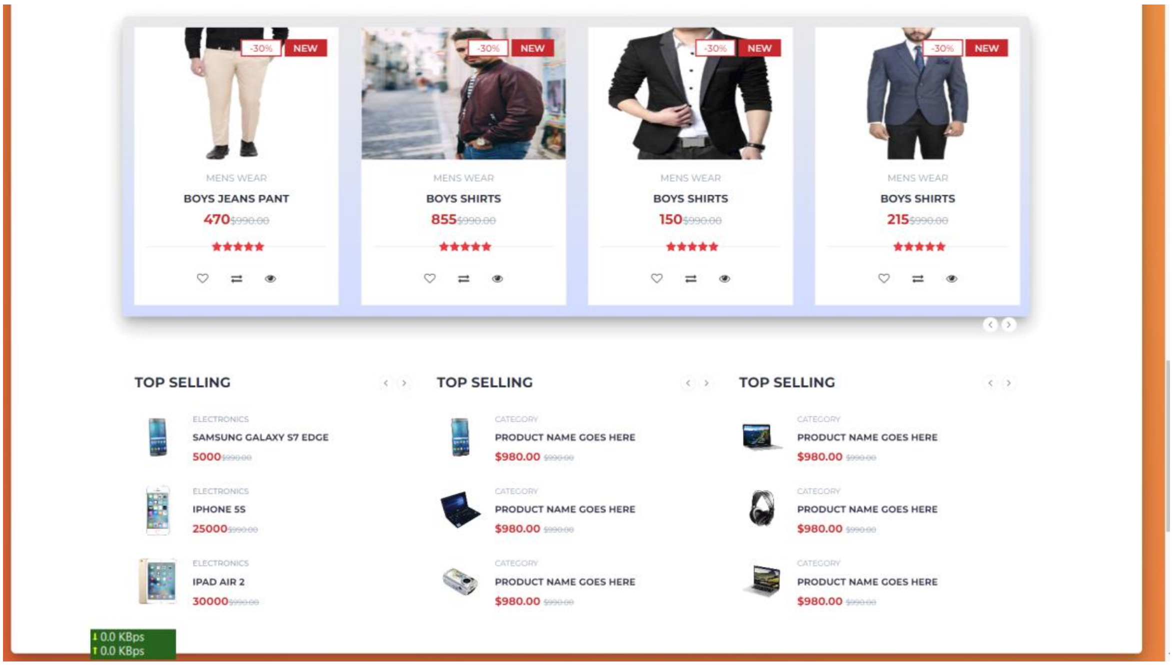
RESULTS

This section describes the screens of “ONLINE SHOP DATABASE”. The snapshots are shown below for each module.

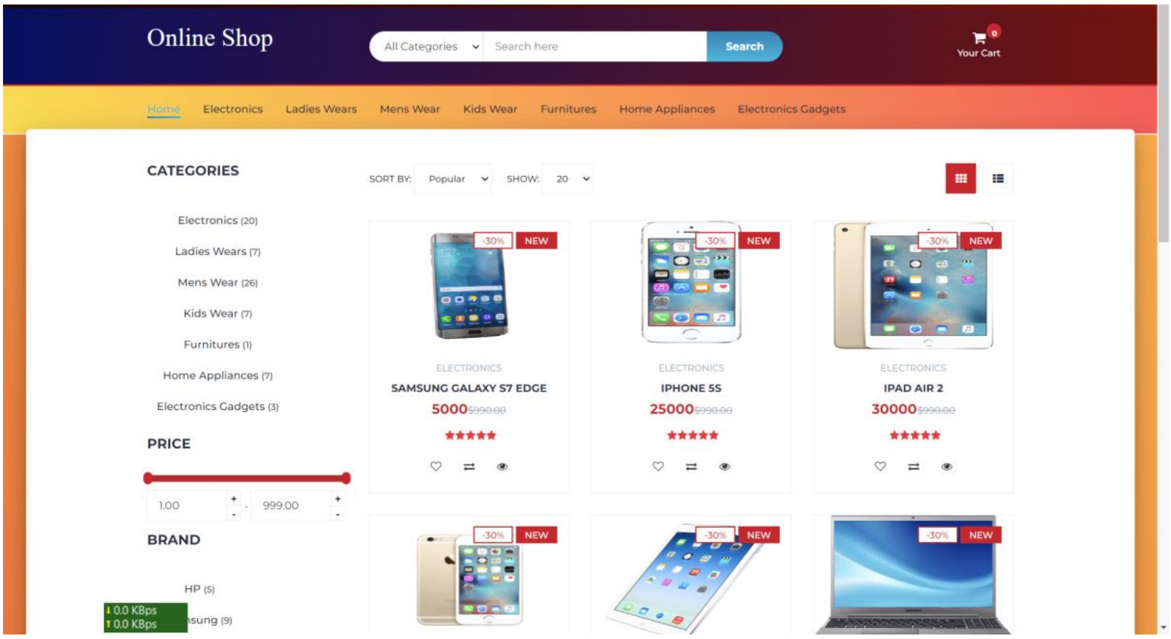
5.1 SNAPSHOTS



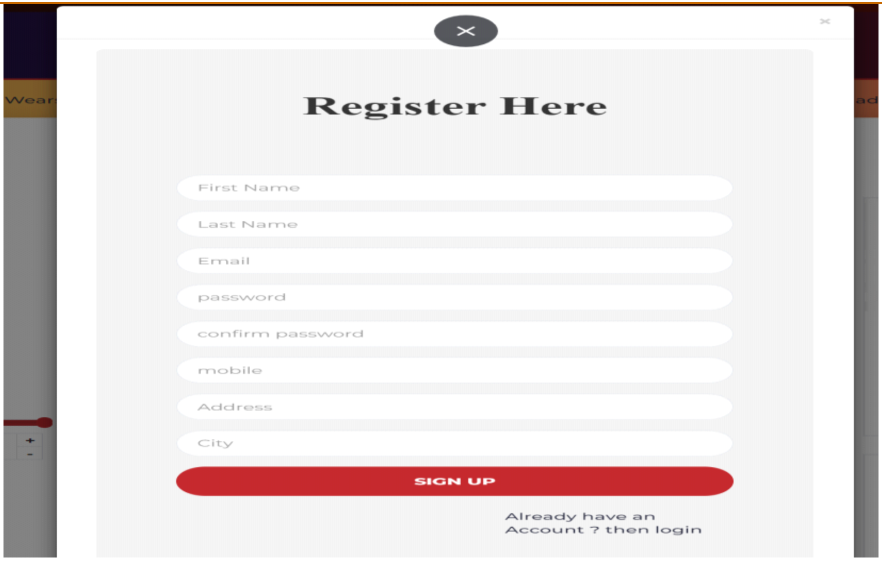
PIC 1: HOME PAGE



PIC 2: MEN'S SECTION



PTC 3: ELECTRONICS SECTION



Register Here

First Name

Last Name

Email

password

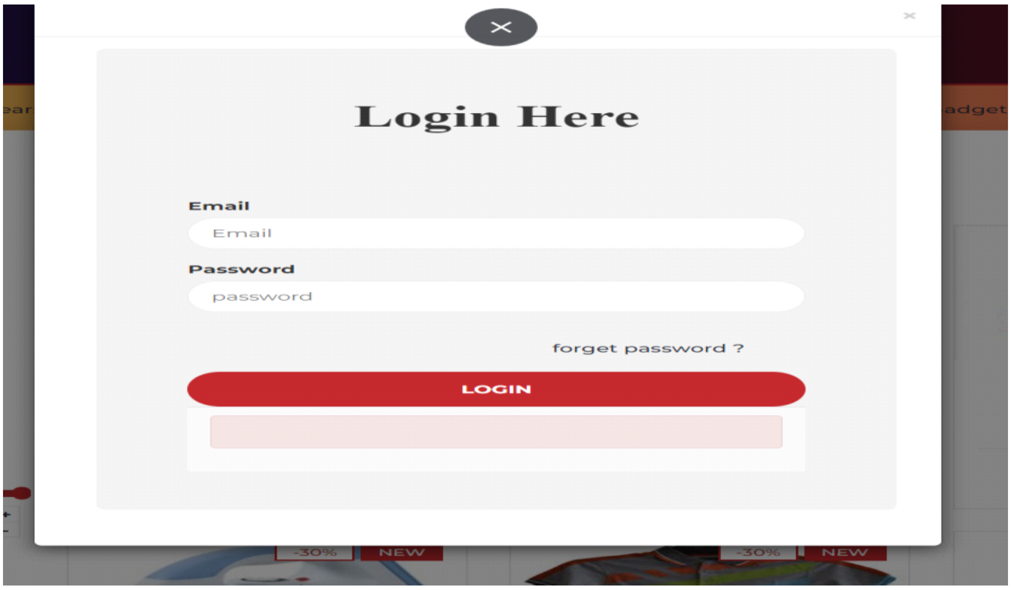
confirm password

mobile

Address

City

PIC 4: REGISTER FORM



Login Here

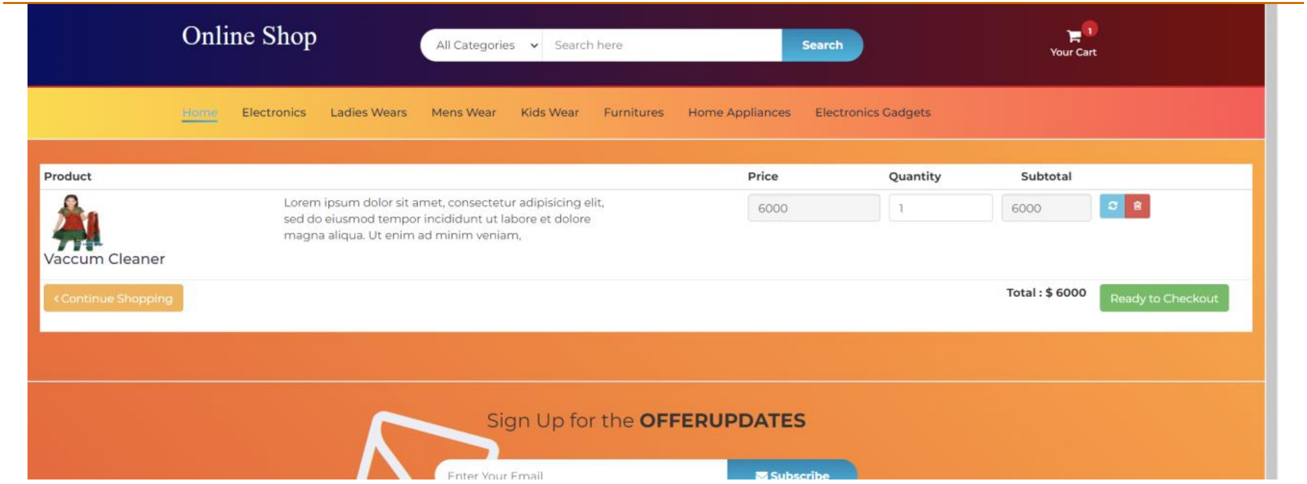
Email

Email

Password

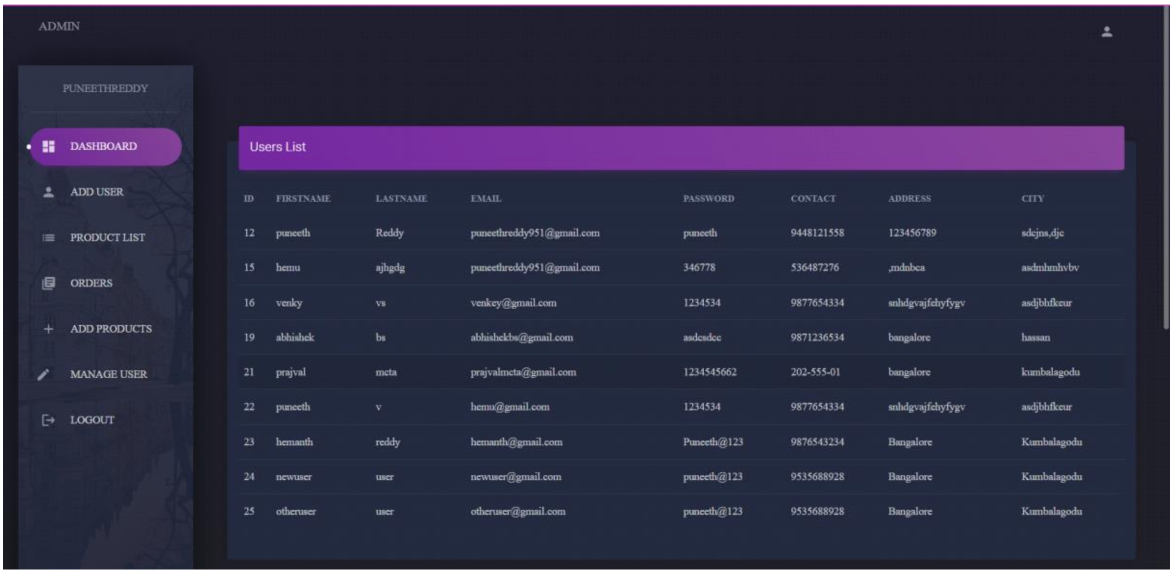
password

PIC 5: LOGIN FORM



Product		Price	Quantity	Subtotal
Vaccum Cleaner	Lorem ipsum dolor sit amet, consectetur adipisicing elit, sed do eiusmod tempor incididunt ut labore et dolore magna aliqua. Ut enim ad minim veniam,	6000	1	6000

PIC 6: CART PAGE



Users List

ID	FIRSTNAME	LASTNAME	EMAIL	PASSWORD	CONTACT	ADDRESS	CITY
12	puneeth	Reddy	puneethreddy951@gmail.com	puneeth	9448121558	123456789	sdcjns,djc
15	hemu	ajhgdg	puneethreddy951@gmail.com	346778	536487276	,mdnbca	asdmhmhvbv
16	venky	vs	venkey@gmail.com	1234534	9877654334	snhdgvajfehyfygv	asdjbhfkeur
19	abhishek	bs	abhishekbs@gmail.com	asdcsdcc	9871236534	bangalore	hassan
21	prajval	meta	prajvalmeta@gmail.com	1234545662	202-555-01	bangalore	kumbalagodu
22	puneeth	v	hemu@gmail.com	1234534	9877654334	snhdgvajfehyfygv	asdjbhfkeur
23	hemanth	reddy	hemanth@gmail.com	Puneeth@123	9876543234	Bangalore	Kumbalagodu
24	newuser	user	newuser@gmail.com	puneeth@123	9535688928	Bangalore	Kumbalagodu
25	otheruser	user	otheruser@gmail.com	puneeth@123	9535688928	Bangalore	Kumbalagodu

PIC 7: ADMIN CONTROL PAGE

CONCLUSION

In conclusion, the development of the online shopping management system was a significant undertaking that aimed to streamline the e-commerce experience. Despite encountering challenges in database design and integration, the project successfully achieved its objectives. Valuable lessons were learned, particularly in terms of system scalability and user interface optimization. Looking ahead, potential enhancements include advanced search functionalities and support for multiple payment gateways. Overall, the project underscores the importance of robust database management systems in facilitating efficient online transactions and contributes to understanding the complexities of e-commerce application development in today's digital

REFERENCES

- Fundamentals of Database System, 7th Edition

  -By Elmasri Ramez and Navathe Shamkanth

- PHP and MySQL Web Development

  -By Luke Welling and Laura Thompson

- HTML & CSS: Design and Build Web Sites

  -By John Duckett

- For Front End Code and CSS styling
  - HYPERLINK "%20%20%20%20%20%20%20%20%20%20%20%20%20%20%20%20https://www.w3schools.com/html" HYPERLINK "%20%20%20%20%20%20%20%20%20%20%20%20%20%20%20%20https://www.w3schools.com/html" HYPERLINK "%20%20%20%20%20%20%20%20%20%20%20%20%20%20%20%20https://www.w3schools.com/html"https://www.w3schools.com/html
  - https://www.stackoverflow.com
  - https://www.tutorialspoint.com

- For MySQL references
  - HYPERLINK "%20%20%20%20%20%20%20%20%20%20%20%20%20%20https://www.youtube.com" HYPERLINK "%20%20%20%20%20%20%20%20%20%20%20%20%20%20https://www.youtube.com" HYPERLINK "%20%20%20%20%20%20%20%20%20%20%20%20%20%20https://www.youtube.com"https://www.youtube.com
  - https://www.udemy.com