# E-commerce platform

Project submitted to the

SRM University – AP, Andhra Pradesh

for the partial fulfillment of the requirements to award the degree of

**Bachelor of Technology/Master of Technology**

In

**Computer Science and Engineering**

**School of Engineering and Sciences**

Submitted by

**Sreemayi Billa (AP22110010549)**
**Sriya Balanagu(AP22110010539)**
**Sowjanya Thota (AP22110010543)**
**Indu Meghana Kalluri (AP22110010540)**
**Vineetha Pemmasani (AP22110010580)**



Under the Guidance of

**(Dr. Aurobindo Behera)**

**SRM University–AP**

**Neerukonda, Mangalagiri, Guntur**

**Andhra Pradesh – 522 240**

**[April, 2024]**

# Certificate

This is to certify that the work present in this Project entitled "**E-Commerce Platform**" has been carried out by

**Sreemayi Billa (AP22110010549)**
**Sriya Balanagu(AP22110010539)**
**Sowjanya Thota (AP22110010543)**
**Indu Meghana Kalluri (AP22110010540)**
**Vineetha Pemmasani (AP22110010580)**

under my/our supervision. The work is genuine, original, and suitable for submission to the SRM University – AP for the award of Bachelor of Technology/Master of Technology in **School of Engineering and Sciences**.

**Supervisor**

Prof. / Dr. Aurobindo Behera

Designation,

Affiliation.

# Acknowledgements

I am immensely grateful to all the individuals and resources that contributed significantly to the realization of this e-commerce database management system project.

First and foremost, heartfelt appreciation goes to the dedicated contributors who devoted their time and expertise to shape this project into its final form. Their unwavering commitment and collaborative efforts were indispensable to its success.

The genesis of this project owes much to the innovative ideas and guidance from Professor Aurobindo Behra, whose insights served as a beacon throughout the development journey.

The learning curve of implementing database management systems was greatly smoothed by referring to educational resources provided by our esteemed faculty in charge. These resources, curated by their respective authors, provided invaluable insights and knowledge, deserving recognition for enriching our understanding.

A special acknowledgement is due to Professor Aurobindo Behra for his steadfast support and mentorship. His guidance not only navigated us through technical intricacies but also elevated the project with strategic perspectives, continually pushing its boundaries through attentive monitoring and insightful advice.

Lastly, but certainly not least, I extend my heartfelt gratitude to friends and family for their unwavering encouragement. Their steadfast support served as a constant source of motivation, propelling the project forward and transforming this achievement into a collective success.

Finally, I would like to extend my sincere gratitude to everyone who contributed, whether directly or indirectly, to the success of this project, and for their contribution to the success and development of this concept.

# Table of Contents

# Abstract

E-commerce databases that facilitate uninterrupted online transactions, manage huge product catalogs and encourage dynamic customer involvement are the pillars of online retail operations. This abstract presents an advanced e-commerce database system implemented to cater for the evolving needs of digital retail.

Tracking, updating and managing order statuses are all part of the order processing process. Data is used in CRM to empower customer experiences, administer accounts, and create loyalty. It is a strategic approach towards inventory management that tracks, forecasts and replenishes inventories in real time so as to minimize stock-outs and maximize sales.

To secure sensitive payment information and guarantee safe transactions, robust encryption techniques as well as authentication mechanisms should be put into place.

DBMS Implementation: The e-commerce database system utilizes a reliable and scalable DBMS platform based on industry-standard practices and technologies for maximum performance.Relational database modeling methods establish logical relationships between data entities while normalizing the structure of the database.

Moreover, the e-commerce database systems have become a key component for online operations as well as other benefits such as:

- Enhancing overall efficiency by properly allocating the resources and simplifying processes that will bring down expenses. Therefore, seamless personalized shopping experience can be enabled through effective ordering process, targeted marketing campaigns and responsive customer care.

- Data Security: This includes safeguarding customer's private information and financial data from hacking and data leakage in the internet.

- Scalability: To adapt with changing business requirements and increase transaction loads, infrastructures should also grow proportionately.

To conclude, electronic commerce databases are irreplaceable in the digital transformation of retail making companies unique. It helps in carrying out efficient activities bettering security while providing excellent user experiences which cater for sustainable expansion.

# Abbreviations

DBMS            Database management system

MySQL             My Structured Query Language

CRM            Customer Relationship Management

ER             Entity Relationship

EER              Enhanced Entity Relationship

# List of Figures

# 1.    Introduction

Mini world and Description: The proliferation of e-commerce has fundamentally changed the way businesses operate and consumers shop in a constantly changing commercial environment. The E-commerce Management DBMS is our project which appreciates the value of moving with the changing trends to digital, providing all-inclusive solutions for both businesses and customers.

Because of this, we are planning to come up with a database for small cloth sellers who would like to sell their products online.

Our project at its core seeks to facilitate smooth interaction between sellers and buyers within the web markets.

Through user-centricity, our project is accessible, easy to use and highly functional as it could be. All interactions right from when a new user signs up on the website till they check out have been designed in such a way that they can enhance the overall shopping experience.

Key Features:

- User Registration: New users can easily sign up on the website by creating personal accounts that unlock access to different features and service options available on it.

- Order History: Consumers can see their order history so that they can review their past purchases before making informed decisions.

- Sale Management: Admins have power to launch sales with discounts on all products. This engages customers, boosting sales.

- Product Filtering: Shoppers can filter goods by category, brand, size, price. Tailors their shopping time.

- Seller Interaction: Sellers easily control listings and accounts. Including unregistering or halting specific merchandise.

- User Profile Management: Both sellers, customers can adjust personal data and preferences. Keeping info current.

- Analytics and Reporting: Admins access insightful analytics, reporting. Know bought products, dates, items sold, profits. Aiding data-driven choices.

Our project makes good use of a relational database schema with SQL. This ensures the system works well, can handle more users, and works fast. By using a database well, we want to build a strong and reliable platform. This platform will help businesses grow and succeed while making things easy and satisfying for customers.

Essentially, the E-commerce DBMS project tries to connect traditional commerce with the digital world. By using new technologies, we hope to grow a lively online marketplace. This marketplace will benefit sellers, customers, and admins equally.

# 2. Methodology

The methodology used for the development of the e-commerce DBMS project is composed of five main phases. These phases are indispensable so that everything linked to the project can be properly planned, implemented, and maintained.

## 2.1. Requirement Phase:

- In this entry stage, detailed work was undertaken to find out and classify the requirements of the online shop from different stakeholders, who belonged to the customers, sellers and administrators.
- How the prerequisites were determined is through interviews, surveys, and study to apprehend the assorted needs and choices of the target users.
- Functional and non-functional requirements were outlined and described including product registration and sales reporting, keeping in mind safety protocols.

### 2.1.1. Functional Requirements

**User Registration:**

- The users account registration process should be very easy, allowing them to provide their essential details for account creation.

**View Cart Details:**

- Customers will have an immediate overview of products in their cart including quantity, price, and incentives applied such as discounts.

**Order History:**

- Customers are required to see the history of their orders so that they can track the details of the previous purchase. These include the order date, product's details, and order status.

**Admin-initiated Sales:**

- Administrators must have a tool to create sales with the help of reducing prices from products, and activate offers like special promotion to attract customers.

**Product Filtering:**

- Through this option, customers will have the opportunity to manage their search based on the criteria they desire be it category, price range, brand or any other product details, making the search easy for them.

**Cart Management:**
- Customers must have the ability to add or remove products from their shopping carts as they see fit, thus giving them the flexibility and the ability to control what they purchase.

**Seller Management:**
- Sellers should be able to deregister or remove their products from the platform at any time that fit the needs of management of their seller accounts.

**User Profile Updates:**
- It is important to give them access to the tools for tracking information related to their profile and account variances which will provide a constant monitoring of their personal data.

**Admin Reporting:**
- Managers should have the ability to view what products were purchased on specific dates, the quantity of products currently on stock, as well as the number of products sold within particular periods of time, which allows for quality performance analysis and decision-making.

**Cart Total Display:**
- You should be able to view the subtotal price of products currently in your cart, so to speak, the transparency provides a firm foundation for making purchasing decisions.

**Customer Management:**
- The capability of admins to access the details of customers who haven't made any purchase is yet a crucial factor for targeted communication and marketing campaigns, these allow admins to attract and maintain customers.

**Profit Tracking:**
- The sub administrators must be able to see total profits earned from the website which will then be used as a tool for prudent financial management and for evaluating the capacity of the business.

### 2.2. Design Phase:

- The course of design then moved ahead to incorporate the identified requirement by considering the conceptualization of the DBMS for e-commerce.
- The table schema design was highly necessary, consisting of the Cart, Seller, Customer, Product, Seller_Phone_num, Cart_item, and Payment table sheets that entailed the corresponding relationships, constraints, and connectivity.
- Furthermore, the architecture diagram depicting the interactions between customer, seller, and administrator modules was developed or drawn and accessibility was made simple and convenient.

### 2.3. Implementation Phase:

- Given the design specifications the deployment phase has started with the implementation of a complete e-commerce DBMS.
- Backend development was made using database management systems MySQL
- Frontend development is the field of the internet which includes designing user-friendly interfaces with the aid of web developing technologies such as HTML, CSS, JavaScript, and frameworks including React and Angular.
- On the safe side, the security measures like authentication, authorization and data encryption were ensured on the system to secure sensitive information.

### 2.4. Verification Phase:

- In particular, it consisted of thorough realization testing to confirm the system's correctness, efficiency and stability.
- The unit testing process was done so as to confirm that the individual components were correct, but integration testing was done to be sure that the elements from the backend and the frontend behaved but also worked correctly together.
- System tests were executed in order to see whether the current system is compliant with the initially mentioned requirements and users acceptance tests (UAT) which allowed the demonstrated end users to give their feedback and later help to get system improvement.

**2.5. Maintenance Phase:**

- On the maintenance side, I had to monitor and ensure that the operational performance of the e-commerce DBMS system was up to mark and also making any possible updates that were needed.
- The mission's effectiveness relied heavily on continuous observation of the system's performance and promptly resolving any emergencies or bugs that came through.
- And, we also made constant improvements and added features based on users' needs and requirements that changed. However, we always monitored and worked reactively to issues, as well as proactively towards ensuring the stability and reliability of the system.

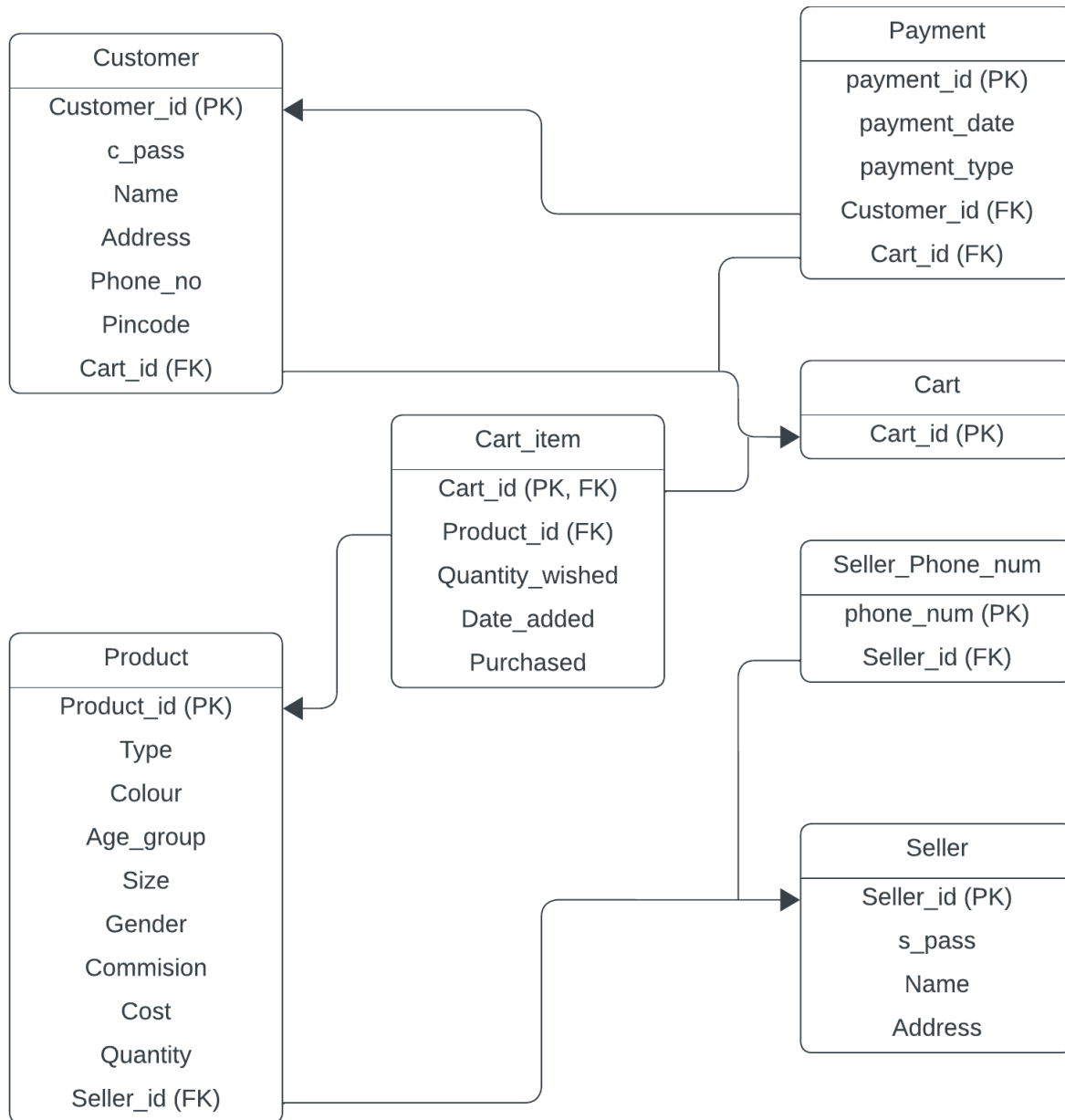# 3. Discussion

## 3.1. Relational Database Schema



fig. 1. Relational database schema

Through this schema I can control, handle and evolve all the needs that are required by the customers, the sellers, the product, carts, cart items or payments within the e-commerce platform. The constraints of ForeignKey prevent data redundancy, and so speed up querying and data manipulation. You may go deep and diversify that framework according to specific needs or services which you are going to offer to your clients.

## 3.2. Relation Descriptions

**Relation Cart:**

The Cart table is where the ID of the cart to every customer is stored. The cart ID is specialized for every single shopping and serves as the primary key of the specific table.

Cart_id (VARCHAR(7), NOT NULL, PRIMARY KEY): unique identifier for each cart

**Relation Customer:**

Customer table stores parentheses match all the information of every customer. A customer is assigned a unique customer ID which acts as the primary key for the customers table data. In the view of the table, there may be the customer's password, name, address, pincode, and phone number. The cart ID has the linking role of a foreign key which connects the customer's identity to their cart as a well.

Customer_id (VARCHAR(6), NOT NULL, PRIMARY KEY): unique identifier for each customer

c_pass (VARCHAR(10), NOT NULL): customer password

Name (VARCHAR(20), NOT NULL): customer name

Address (VARCHAR(20), NOT NULL): customer address

Pincode (NUMBER(6), NOT NULL): customer pincode

Phone_number_s (NUMBER(10), NOT NULL): customer phone number

Cart_id (VARCHAR(7), NOT NULL): foreign key referencing the Cart table

**Relation Seller:**

SellerData that holds the seller data is stored in the Seller table. Seller ID of each is the seller's only key that is used as a primary key for the table. In this table, the password, name and address-all of the seller- are also available.

Seller_id (VARCHAR(6), NOT NULL, PRIMARY KEY): unique identifier for each seller

s_pass (VARCHAR(10), NOT NULL): seller password

Name (VARCHAR(20), NOT NULL): seller name

Address (VARCHAR(10), NOT NULL): seller address

**Relation Seller_Phone_num:**

The Seller_Phone_num table is used to store phone numbers for each person who is selling. This represents the relationship between two phone numbers and the same seller. Thus, the table has a primary key jointly composed by the combination of the phone numbers and the seller ID. Consequently, the seller ID too acts as a foreign key to join these two- phone numbers and the seller.

Phone_num (NUMBER(10), NOT NULL, PRIMARY KEY): phone number for the seller

Seller_id (VARCHAR(6), NOT NULL, PRIMARY KEY, FOREIGN KEY): foreign key referencing the Seller table

**Relation Payment :**

The table Expresses the payment is for the relevant customer. This an individual payment appears in a table, with a unique payment ID serving as the primary key and a hidden data field. The table also includes the corresponding date, payment type, customer ID, order ID and the total amount to be paid. The customer_id and cart_id are employed here as the relation keys, which enable you to connect the payment to the customer.

payment_id (VARCHAR(7), NOT NULL, PRIMARY KEY): unique identifier for each payment

payment_date (DATE, NOT NULL): date of the payment

Payment_type (VARCHAR(10), NOT NULL): type of payment (e.g. credit card, debit card, etc.)

Customer_id (VARCHAR(6), NOT NULL, FOREIGN KEY): foreign key referencing the Customer table

Cart_id (VARCHAR(7), NOT NULL, FOREIGN KEY): foreign key referencing the Cart table

total_amount (NUMBER(6)): total amount of the payment

**Relation Product:**

The Product table stores data in this manner so that each product has its specific characteristics. Every item of category will have a separate product ID which serves the purpose to be unique for the table to be the primary key for it. The table will also include the product's type, color, size, gender, commission, cost and quantity. The seller ID is a record's ID to link a product to a seller.

Product_id (VARCHAR(7), NOT NULL, PRIMARY KEY): unique identifier for each product

Type (VARCHAR(7), NOT NULL): type of the product

Color (VARCHAR(15), NOT NULL): color of the product

P_Size (VARCHAR(2), NOT NULL): size of the product

Gender (CHAR(1), NOT NULL): gender of the product (e.g. men, women, unisex)

Commission (NUMBER(2), NOT NULL): commission for the product

Cost (NUMBER(5),NOT NULL): cost of the product

Quantity (NUMBER(2), NOT NULL): quantity of the product

Seller_id (VARCHAR(6), FOREIGN KEY): foreign key referencing the Seller table

**Relation Cart_item:**

The Cart_item table, in turn, stores cart information about each of the items which a customer has added to their cart. In the database design, each item has a specific cart ID and product ID, combining them both to be the primary key for the table. The table comprises the item quantity, date of item added to the cart and other columns which show if the item has been purchased or not. Foreign keys are created by using the same cartID and productID as the links that connect the item to the cart as well as the product.

Quantity_wished (NUMBER(1), NOT NULL): quantity of the item wished

Date_Added (DATE, NOT NULL): date the item was added to the cart

Cart_id (VARCHAR(7), NOT NULL, FOREIGN KEY): foreign key referencing the Cart table

Product_id (VARCHAR(7), NOT NULL, FOREIGN KEY): foreign key referencing the Product table

purchased (VARCHAR(3), DEFAULT 'NO'): indicates whether the item has been purchased or not.

## 3.3. Entity Relation Diagram

The e-commerce project ER diagram is an important element that is needed for database creation and design. It gives a structured graphical view of the database architecture, which simplifies the understanding of different entities and their relations as well as allows to control the quality and completeness of the system data. Using the ER diagram structure as the basis, we shall be able to implement and manage the database in such a way that the e-commerce platform works according to our plan.
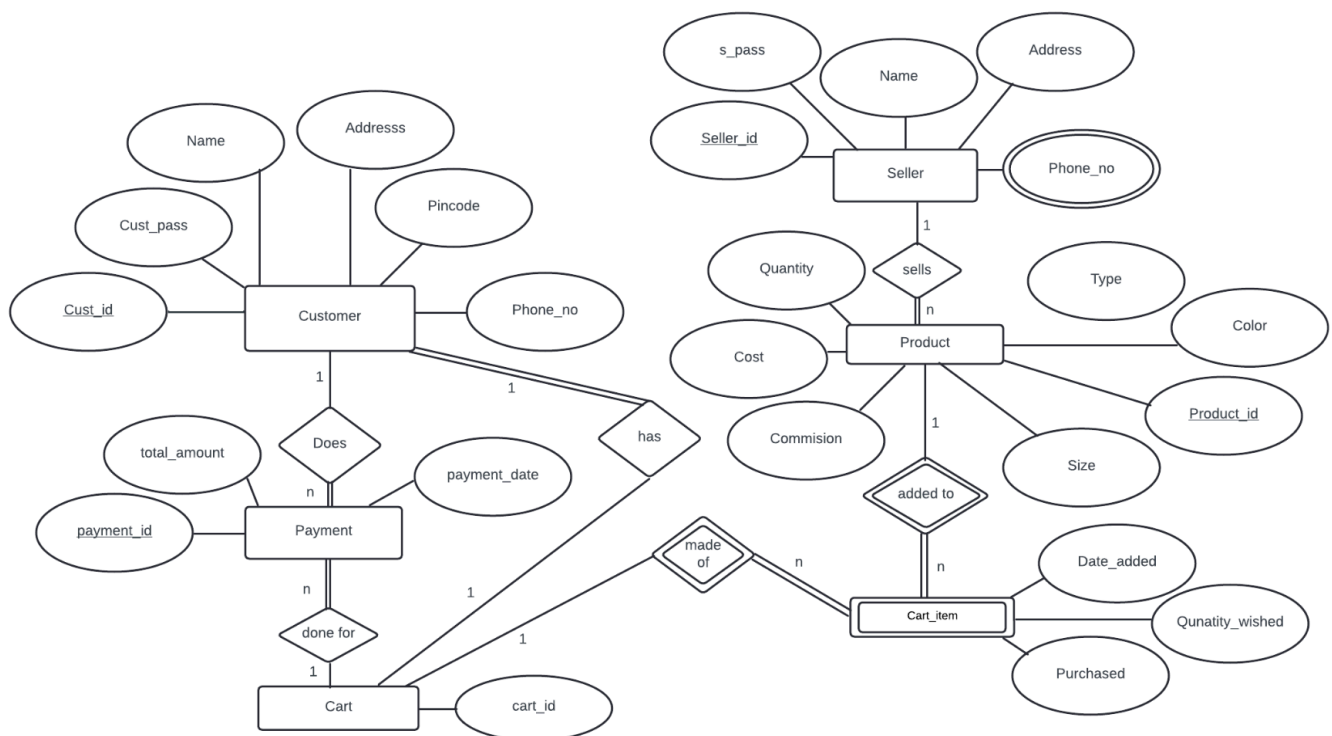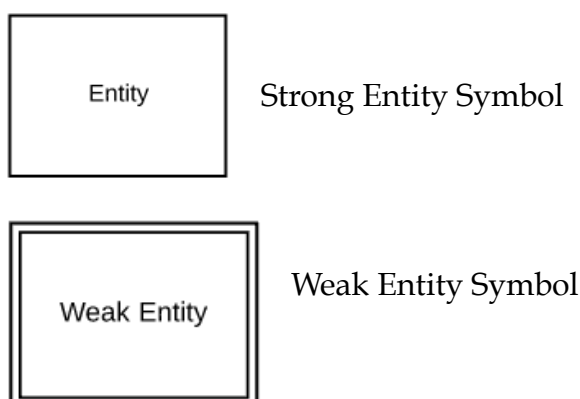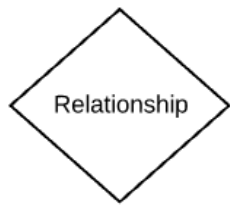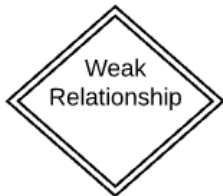


fig 2. ER Diagram of E-commerce Management system

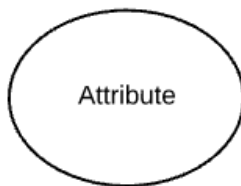fig 3. ER Diagram Symbols and their meanings:



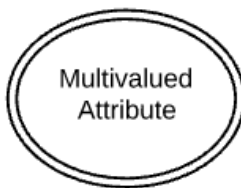Strong Entity Symbol



Weak Entity Symbol

Relationship


Weak relationship


Attribute


Multivalued attribute


Partial participation


Total participation

## 3.4. Enhanced Entity Relation Diagram(EER)

EER diagrams provide a visual representation of the relationships among the tables in your model.
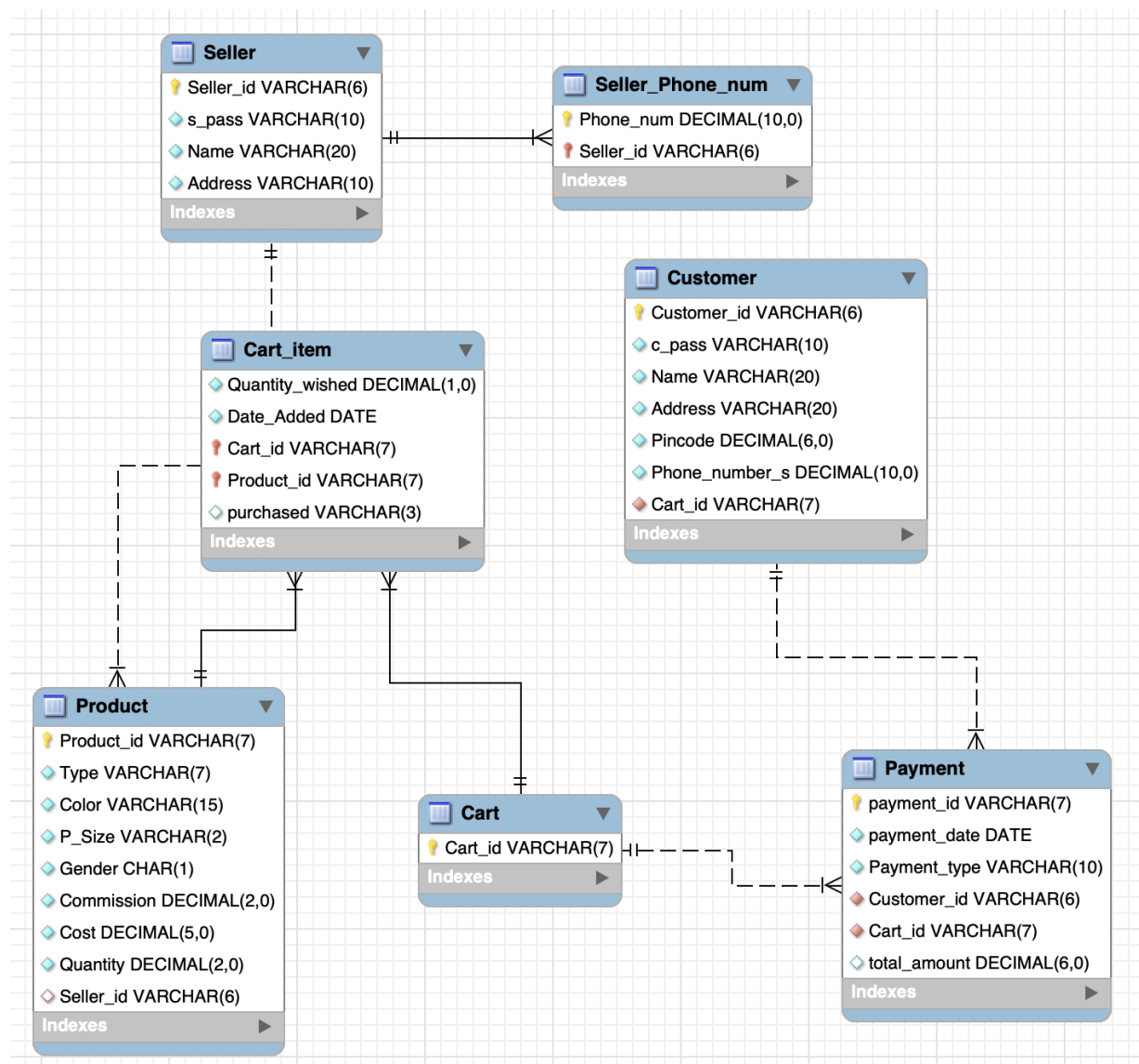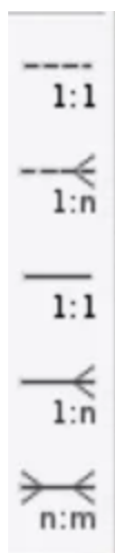


fig 4. EER diagram of E-commerce Management system

fig 5. EER Symbols and their meaning:

**1.One-to-One (1:1) Relationship**

**2. One-to-Many (1:N) Relationship**

**3. Many-to-One (N:1) Relationship**

**4. Many-to-Many (N:N) Relationship**

## 3.5. Queries

### 3.5.1. Creating Relations:

CREATE TABLE Cart
(
    Cart_id VARCHAR(7) NOT NULL,
    PRIMARY KEY(Cart_id)
);


CREATE TABLE Customer
(
    Customer_id VARCHAR(6) NOT NULL,
    c_pass VARCHAR(10) NOT NULL,
    Name VARCHAR(20) NOT NULL,
    Address VARCHAR(20) NOT NULL,
    Pincode NUMERIC(6) NOT NULL,
    Phone_number_s NUMERIC(10) NOT NULL,
    PRIMARY KEY (Customer_id),
    Cart_id VARCHAR(7) NOT NULL,
    FOREIGN KEY(Cart_id) REFERENCES cart(Cart_id)
);


CREATE TABLE Seller
(
    Seller_id VARCHAR(6) NOT NULL,
    s_pass VARCHAR(10) NOT NULL,
    Name VARCHAR(20) NOT NULL,
    Address VARCHAR(10) NOT NULL,

```
    PRIMARY KEY (Seller_id)
);


CREATE TABLE Seller_Phone_num
(
    Phone_num NUMERIC(10) NOT NULL,
    Seller_id VARCHAR(6) NOT NULL,
    PRIMARY KEY (Phone_num, Seller_id),
    FOREIGN KEY (Seller_id) REFERENCES Seller(Seller_id)
    ON DELETE CASCADE
);


CREATE TABLE Payment
(
    payment_id VARCHAR(7) NOT NULL,
    payment_date DATE NOT NULL,
    Payment_type VARCHAR(10) NOT NULL,
    Customer_id VARCHAR(6) NOT NULL,
    Cart_id VARCHAR(7) NOT NULL,
    PRIMARY KEY (payment_id),
    FOREIGN KEY (Customer_id) REFERENCES Customer(Customer_id),
    FOREIGN KEY (Cart_id) REFERENCES Cart(Cart_id),
    total_amount numeric(6)
);


CREATE TABLE Product
(
    Product_id VARCHAR(7) NOT NULL,
    Type VARCHAR(7) NOT NULL,
```

```
    Color VARCHAR(15) NOT NULL,

    P_Size VARCHAR(2) NOT NULL,

    Gender CHAR(1) NOT NULL,

    Commission NUMERIC(2) NOT NULL,

    Cost NUMERIC(5) NOT NULL,

    Quantity NUMERIC(2) NOT NULL,

    Seller_id VARCHAR(6),

    PRIMARY KEY (Product_id),

    FOREIGN KEY (Seller_id) REFERENCES Seller(Seller_id)

    ON DELETE SET NULL

 );


 CREATE TABLE Cart_item
(

 Quantity_wished NUMERIC(1) NOT NULL,

 Date_Added DATE NOT NULL,

 Cart_id VARCHAR(7) NOT NULL,

 Product_id VARCHAR(7) NOT NULL,

 FOREIGN KEY (Cart_id) REFERENCES Cart(Cart_id),

 FOREIGN KEY (Product_id) REFERENCES Product(Product_id),

 PRIMARY KEY (Cart_id, Product_id)

);


 alter table Cart_item add purchased varchar(3) default 'NO';
```

### 3.5.2. Populating Relations:

insert into Cart values('crt1012');

insert into Customer values('cid101','XYZM4567','Aarav','H-234','560078',9876543210, 'crt1012');

insert into Seller values('sid101','67890','Priya','Mumbai');

insert into Product values('pid1002','saree','blue','S','F',15,1500,15,'sid101');

insert into Seller_Phone_num values('9876543210','sid101');

insert into Cart_item values(2,'1999-10-11','crt1012','pid1002','N');

insert into Payment values('pmt1002','1999-10-11','online','cid101','crt1012',NULL);


insert into Cart values('crt1013');

insert into Customer values('cid102','PQRS7890','Amit','G-890','411027',9898765432, 'crt1013');

insert into Seller values('sid102','54321','Neha','Pune');

insert into Product values('pid1003','kurta','green','L','M',10,1200,20,'sid102');

insert into Seller_Phone_num values('9898765432','sid102');

insert into Cart_item values(1,'1999-10-12','crt1013','pid1003','N');

insert into Payment values('pmt1003','1999-10-12','online','cid102','crt1013',NULL);


insert into Cart values('crt1014');

insert into Customer values('cid103','LMNO9876','Ananya','A-456','700016',9988776655, 'crt1014');

insert into Seller values('sid103','43210','Rahul','Kolkata');

insert into Product values('pid1004','shirt','yellow','XL','M',12,800,10,'sid103');

insert into Seller_Phone_num values('9988776655','sid103');

insert into Cart_item values(3,'1999-10-13','crt1014','pid1004','N');

insert into Payment values('pmt1004','1999-10-13','online','cid103','crt1014',NULL);


insert into Cart values('crt1015');

insert into Customer values('cid104','STUV2345','Vidya','B-678','600040',9876543210, 'crt1015');

insert into Seller values('sid104','98765','Arun','Chennai');

insert into Product values('pid1005','dress','pink','M','F',10,2000,25,'sid104');

insert into Seller_Phone_num values('9876543210','sid104');

insert into Cart_item values(2,'1999-10-14','crt1015','pid1005','N');

insert into Payment values('pmt1005','1999-10-14','online','cid104','crt1015',NULL);

insert into Cart values('crt1016');

insert into Customer values('cid105','UVWX4567','Sakshi','E-123','110001',9876543210,
'crt1016');

insert into Seller values('sid105','87654','Amitabh','Delhi');

insert into Product values('pid1006','t-shirt','black','L','M',10,600,30,'sid105');

insert into Seller_Phone_num values('9876543210','sid105');

insert into Cart_item values(1,'1999-10-15','crt1016','pid1006','N');

insert into Payment values('pmt1006','1999-10-15','online','cid105','crt1016',NULL);

insert into Cart values('crt1017');

insert into Customer values('cid106','WXYZ3456','Riya','F-789','411057',9876543210,
'crt1017');

insert into Seller values('sid106','76543','Aditya','Pune');

insert into Product values('pid1007','shoes','brown','8','M',15,2500,10,'sid106');

insert into Seller_Phone_num values('9876543210','sid106');

insert into Cart_item values(2,'1999-10-16','crt1017','pid1007','N');

insert into Payment values('pmt1007','1999-10-16','online','cid106','crt1017',NULL);

insert into Cart values('crt1018');

insert into Customer values('cid107','ABCD5678','Sneha','G-345','500081',9876543210,
'crt1018');

insert into Seller values('sid107','65432','Varun','Hyderabad');

insert into Product values('pid1008','watch','silver','M','M',10,3000,15,'sid107');

insert into Seller_Phone_num values('9876543210','sid107');

insert into Cart_item values(1,'1999-10-17','crt1018','pid1008','N');

insert into Payment values('pmt1008','1999-10-17','online','cid107','crt1018',NULL);

insert into Cart values('crt1019');

insert into Customer values('cid108','EFGH6789','Rohan','C-123','560102',9876543210, 'crt1019');

insert into Seller values('sid108','87654','Akansha','Bangalore');

insert into Product values('pid1009','bag','purple','M','F',10,1000,20,'sid108');

insert into Seller_Phone_num values('9876543210','sid108');

insert into Cart_item values(2,'1999-10-18','crt1019','pid1009','N');

insert into Payment values('pmt1009','1999-10-18','online','cid108','crt1019',NULL);


insert into Cart values('crt1020');

insert into Customer values('cid109','IJKL7890','Shreya','D-567','400022',9876543210, 'crt1020');

insert into Seller values('sid109','98765','Sahil','Mumbai');

insert into Product values('pid1010','earrings','gold','M','F',20,5000,10,'sid109');

insert into Seller_Phone_num values('9876543210','sid109');

insert into Cart_item values(1,'1999-10-19','crt1020','pid1010','N');

insert into Payment values('pmt1010','1999-10-19','online','cid109','crt1020',NULL);

# 4. Concluding Remarks

For our E-commerce database management system project, we have engineered a versatile relational database scheme to help the business in many tasks relating to online purchases.

The database has been laid out with referential integrity constraints that provide not only for the accuracy but also for the consistency of data. Tables are related to each other by foreign key relations which helps to establish data consistency and implement business logic. Aling with this, we define the primary keys appropriately for every table that should be unique key identifiers for each record.

The system comprises the functions of adding products to the cart, payment of the money, and maintenance of seller and customer details. Yet there is still a lot of room for development, for example, in terms of functionality including tracking orders, authentication of users and inventory management.

# 5. Future Work

The following developments will result in the platform-application-e-commerce-platform to be easily navigated by the users and the seller's tools for the management to be easily managed.

- User Authentication and Authorization: The login/logout functionality needs to be secure in more than one way and roles access control should be a part of it.

- Order Tracking and Management:Customers can track their orders in real-time and peer to peer support will be useful in order processing.

- Inventory Management:In addition, planning to execute inventory tracking techniques in order to monitor stock availability and adjust accordingly in terms of ordering and re-stocking.

- Enhanced Product Catalog:Promote the product and add great product ingredients.

- User Reviews and Ratings: Provide an opportunity for users to give feedback, grading products and vendors.

- Promotions and Discounts:Create tools for the promotion of Offers, bumps and discounts.

- Responsive Web Design: Guarantee that the platform is tailored-made for different types of devices and screen sizes.

- Performance Optimization:Bearing in mind to continue to optimize DB and application performance.

- Security Enhancements: Reinforcing security measures would protect data, shield it from unlawful access and achieve the relevant goals.

- Integration with External Systems: Survey the integration availability of the application in the field of shipping, payment infrastructure, and marketing tools.

In general, this schema is strong, as it has offered a structure to manage inventories, products, and user interactions which are convenient ways for customers to browse business websites. This tool has a lot of potential that can be improved with the additional options and options could turn out to be as perfect as it could be for an effective webstore.

# References

1. Database Management Systems by Raghu Ramakrishnan and Johannes Gehrke: https://www.db-book.com/db7/
2. Introduction to Database Systems by C.J. Date: https://www.pearson.com/us/higher-education/program/Date-Introduction-to-Database-Systems-8th-Edition/PGM338912.html
3. SQL Tutorial by W3Schools: https://www.w3schools.com/sql/
4. Database Design for Mere Mortals: A Hands-On Guide to Relational Database Design by Michael J. Hernandez: https://www.informit.com/store/database-design-for-mere-mortals-a-hands-on-guide-to-9780132358040
5. MySQL Documentation: https://dev.mysql.com/doc/
6. https://www.softwareideas.net/erd-relation-arrows
7. https://www.lucidchart.com/pages/ER-diagram-symbols-and-meaning
8. "Fundamentals of Database Systems" by Ramez Elmasri and Shamkant Navathe
9. "Database Systems: Concepts, Design, and Applications" by Shio Kumar Singh
10. "Database Systems: The Complete Book" by Hector Garcia-Molina, Jeffrey D. Ullman, and Jennifer Widom
11. "Database Design and Implementation" by Edward Sciore
12. "Database Management Systems" by Seema Kedar and Sarit Kumar Pattanaik