Database Design and Implementation of a Scalable E- Commerce System

Database Management Systems Lab (0612-304)

presenting to:

Md Ayon Mia Lecturer

Department of CSE

Dhaka International University

Team Members

1.Abdullah Ibrahim(12)

2.Shihab Uddin Soumik(35)

3.Tasnuba Kawsar(40)

4.Md Nure Lokman(43)

Agenda

1.Introduction

2.Objectives

3.Database Design

4.Implementation

5.Future Improvement

6.Query

7.Conclusion

Goals

- Design an efficient database
- Ensure data consistency and accuracy
- Improve system performance
- Enhance data security and integrity

Project Overview

1. Design a normalized, scalable relational database for e-commerce.
2. Ensure data integrity and consistency across operations.
3. Support customer, vendor, product, and order management.
4. Implement secure login, reviews, and payment tracking.
5. Enable future scalability with modular relationships.

Entities and Relationships

1. Entity: Users & Address

Attributes:

A.user_id (PK) – Unique, NOT NULL B.user_name – NOT NULL

C.email – Unique, NOT NULL

D.address – Unique, NOT NULL

2. Entity: Products, Product Images

Attributes:

- product_code (PK) – Unique, NOT NULL
- product-name,NOT NULL
- Product_quantity
- category_name – NOT NULL
- price – NOT NULL, positive

3. Entity: Cart & Wishlist (Weak)

Attributes:

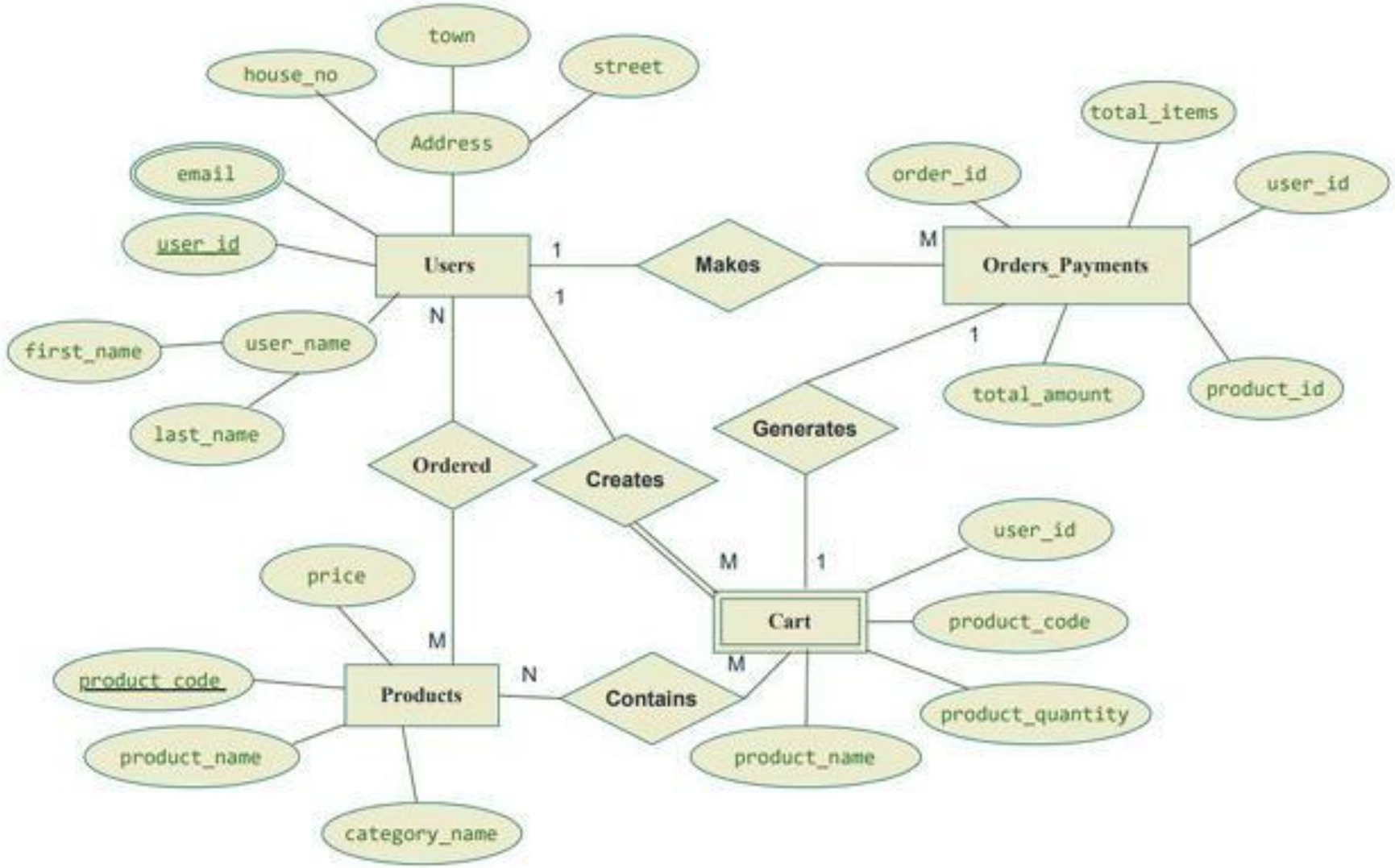
- user_id – NOT NULL
- product_code – NOT NULL
- product_name – NOT NULL
- product_quantity – NOT NULL, >1

4.Entity: Orders & Payments

Attributes:

- order_id (PK) – Unique
- user_id – NOT NULL
- product_id (FK → Products.product_id) – NOT NULL
- total_amount – NOT NULL, positive
- total_items – NOT NULL, positive

ERDiagram


town
house_no
street
Address
email
user_id
Users
1
1
N
Makes
M
Orders_Payments
order_id
total_items
user_id
first_name
user_name
last_name
total_amount
product_id
1
Generates
Ordered
Creates
price
M
1
user_id
Cart
product_code
M
product_code
Products
N
Contains
product_quantity
product_name
product_name
category_name

Mapping

1.users

user_id	F_name	L_name	town	street	House_no

2.user-Email

user_id	u_Email

3.Address

House_no	town	street

4.Order payment

order_id	user_id	Total_ amount	total_item	Product_id

5.Makes

User_id	order_id

6.Ordered

User_id	order_id

7.Cart

user_id	Product_code

SQL query and output table

1.Show all customers

```
84
85      -- 1.Show all customers
86  •   SELECT * FROM customers;
87
88      -- 2.Show all products with their categories
89  •   SELECT p.product_name, p.price, c.category_name
90      FROM products p
```

Result Grid | Filter Rows: | Edit: | Export/Import

customer_id	name	email	city
1	Abdullah Ibrahim	abdullah@gmail.com	Dhaka
2	Tajmina Tabbasum	tajmina@gmail.com	Chittagong
3	Rafi Khan	rafi@gmail.com	Sylhet
NULL	NULL	NULL	NULL

Result Grid | Form Editor

2. Show all products with their categories

```
88      -- 2.Show all products with their categories
89  •   SELECT p.product_name, p.price, c.category_name
90      FROM products p
91      JOIN categories c ON p.category_id = c.category_id;
92
93      -- 3.Display all orders
```

Result Grid | Filter Rows: | Export: | Wrap Cell Content:

product_name	price	category_name
Smartphone	35000.00	Electronics
Laptop	85000.00	Electronics
T-Shirt	800.00	Clothing
Microwave Oven	12000.00	Home Appliances
Novel Book	500.00	Books

Result Grid | Form Editor

SQLquery and output table

3. Show order details with customer name

```
96    -- 4.Show order details with customer name
97    SELECT o.order_id, c.name AS customer_name, o.order_date, o.total_
98    FROM orders o
99    JOIN customers c ON o.customer_id = c.customer_id;
100
101   -- 5.Find products that are low in stock
102   SELECT product_name, stock FROM products WHERE stock < 20;
```

Result Grid | Filter Rows: | Export: | Wrap Cell Content:

order_id	customer_name	order_date	total_amount
1	Abdullah Ibrahim	2025-11-01	35500.00
2	Tajmina Tabbasum	2025-11-02	86000.00
3	Rafi Khan	2025-11-03	12500.00

Result Grid | Form Editor

4.Find products that are low in stock

```
100
101   -- 5.Find products that are low in stock
102   SELECT product_name, stock FROM products WHERE stock < 20;
103
104   -- 6.Show order items with product names
105   SELECT oi.order_id, p.product_name, oi.quantity, oi.subtotal
```

Result Grid | Filter Rows: | Export: | Wrap Cell Content:

product_name	stock
Laptop	10
Microwave Oven	15

Result Grid | Form Editor | Field Types

Future work

1.Design a System.

2.Partitioning Large Tables for faster query performance.

3.Database Auditing & Logging for all user actions.

4.Dynamic Reporting Views for business intelligence dashboards.

Conclusion

This SQL-based project shows the main database structure needed for a working e-commerce system. With a well organized schema, rules to keep data correct and automation using triggers and stored procedures. It ensures data is safe, consistent and easy to manage. It provides a solid foundation for connecting the front-end and can be expanded to handle more advanced features for real-world use.

Thank You