

Subject Name: **DBMS**

Subject Code: **24CSE0209**

Cluster: **iBETA**

Department: **CSE**



Project Report File

Submitted By:

S. No.	Name	Roll No.	Contact
1	Ayush Chauhan	2410990961	8448522434
2	Anmol	2410990924	9729022329
3	Anmol Puri	2410990927	7018022823
4	Ashmit Singh Gogia	2410990953	8809376336

Content Management System

1. Introduction

In today's competitive business landscape, efficient management of inventory and streamlined billing processes are crucial for the success of small and medium-sized enterprises (SMEs). Many small businesses still rely on manual inventory tracking and invoicing methods, which are prone to errors, time-consuming, and can lead to significant financial losses due to stockouts, overstocking, or inaccurate billing. The proposed Content Management System aims to provide a robust, reliable, and user-friendly solution to address these challenges. It will serve as a centralized platform for managing product inventory, processing sales, generating invoices, and tracking financial transactions. This system is designed to empower small business owners with the tools they need to optimize their operations, improve accuracy, and enhance customer satisfaction. It incorporates modern database principles to ensure data integrity, consistency, security, and scalability.

2. Problem Overview

Small businesses often face several interconnected challenges in managing their inventory and billing, including:

1. **Manual Processing:** Relying on spreadsheets or paper-based records for inventory and billing leads to slow data retrieval and updates, making it difficult to respond quickly to market demands or customer inquiries.
2. **Data Redundancy and Inconsistency:** Poorly structured records can result in duplicate entries for products, customers, or transactions, leading to inaccurate inventory counts and billing discrepancies.
3. **Difficulty in Tracking:** It is challenging to track product movement, sales performance, and customer purchasing history without a centralized system, making it hard to identify popular products or loyal customers.
4. **Lack of Centralized Access:** Business owners with multiple employees or sales points may struggle with a unified view of their inventory and sales data, leading to miscommunication and operational inefficiencies.
5. **Security Risks:** Manual or unsecured digital records are vulnerable to loss, theft, or unauthorized access, jeopardizing sensitive business and customer data.
6. **Limited Reporting and Analytics:** Without a proper system, generating insightful reports on sales trends, inventory levels, or financial performance is difficult, hindering informed decision-making.

A well-structured Inventory and Billing Management System eliminates these problems

by providing a centralized, relational database with properly defined entities, relationships, and constraints.

3. Scope of the Project

The scope of this project is to design a relational database that will:

1. Store all details of products, customers, suppliers, orders, invoices, and payments.
2. Represent the complex relationships between these entities in a logical and optimized structure.
3. Enable fast retrieval and updating of records through structured queries for inventory, sales, and billing.
4. Support future scalability by accommodating growth in the number of products, customers, and transactions.
5. Ensure data security and access control to protect sensitive business and customer information.

4. Objectives

The primary objectives of the Inventory and Billing Management System are:

1. **Centralized Data Management:** Develop a relational database model that integrates all inventory, sales, and billing data into a unified system, eliminating duplication and redundancy.
2. **Efficient Data Retrieval:** Optimize the database design to allow fast and accurate retrieval of information for products, customers, orders, and invoices.
3. **Relationship Mapping:** Implement proper handling of one-to-many and many-to-many relationships, such as:
 - a. Customers placing multiple orders.
 - b. Orders containing multiple products.
 - c. Products supplied by multiple suppliers.
4. **Data Integrity and Consistency:** Enforce Primary Keys, Foreign Keys, and constraints to maintain accuracy and validity of data across all modules (inventory, sales, billing).
5. **Security and Access Control:** Protect sensitive business information through proper database security measures and role-based access for different users (e.g., administrator, sales staff).
6. **Scalability and Maintainability:** Ensure the database can handle future growth in product lines, customer base, and transaction volume, and can be easily updated without disrupting operations.
7. **Support for Reporting and Analytics:** Provide the ability to generate reports

for:

- a. Current inventory levels and low stock alerts.
 - b. Sales performance by product, customer, or period.
 - c. Outstanding invoices and payment tracking.
 - d. Supplier performance and purchasing trends.
8. **Minimizing Redundancy:** Apply normalization techniques to reduce data duplication, optimize storage, and maintain consistency across the inventory and billing data.

5. Significance of the Project

The proposed Inventory and Billing Management System will significantly improve small business operations by:

1. **Reducing Operational Delays:** Automating inventory updates and invoice generation will save time and allow businesses to process transactions faster.
2. **Improving Accuracy and Reliability:** Eliminating manual errors in stock counts and billing will lead to more accurate financial records and reduced discrepancies.
3. **Enhancing Customer Satisfaction:** Faster order processing, accurate billing, and improved product availability will lead to a better customer experience.
4. **Strengthening Security:** Centralized and secure data storage will protect sensitive business and customer information from loss or unauthorized access.
5. **Enabling Informed Decision-Making:** Accurate and timely reports on sales, inventory, and finances will provide business owners with the insights needed to make strategic decisions.

Main Entities and Attributes:

1. Product

- Product_ID (PK)
- Name
- Description
- Price
- Stock_Quantity
- Supplier_ID (FK)

2. Customer

- Customer_ID (PK)
- Name
- Address
- Phone
- Email

3. Supplier

- Supplier_ID (PK)
- Name
- Contact_Person
- Phone
- Email

4. Order

- Order_ID (PK)
- Order_Date
- Total_Amount
- Customer_ID (FK)

5. Order_Item

- Order_ID (FK)
- Product_ID (FK)
- Quantity
- Unit_Price

6. Invoice

- Invoice_ID (PK)
- Invoice_Date
- Due_Date

- Total_Amount
- Payment_Status (Paid/Unpaid/Partially Paid)
- Order_ID (FK)

7. Payment

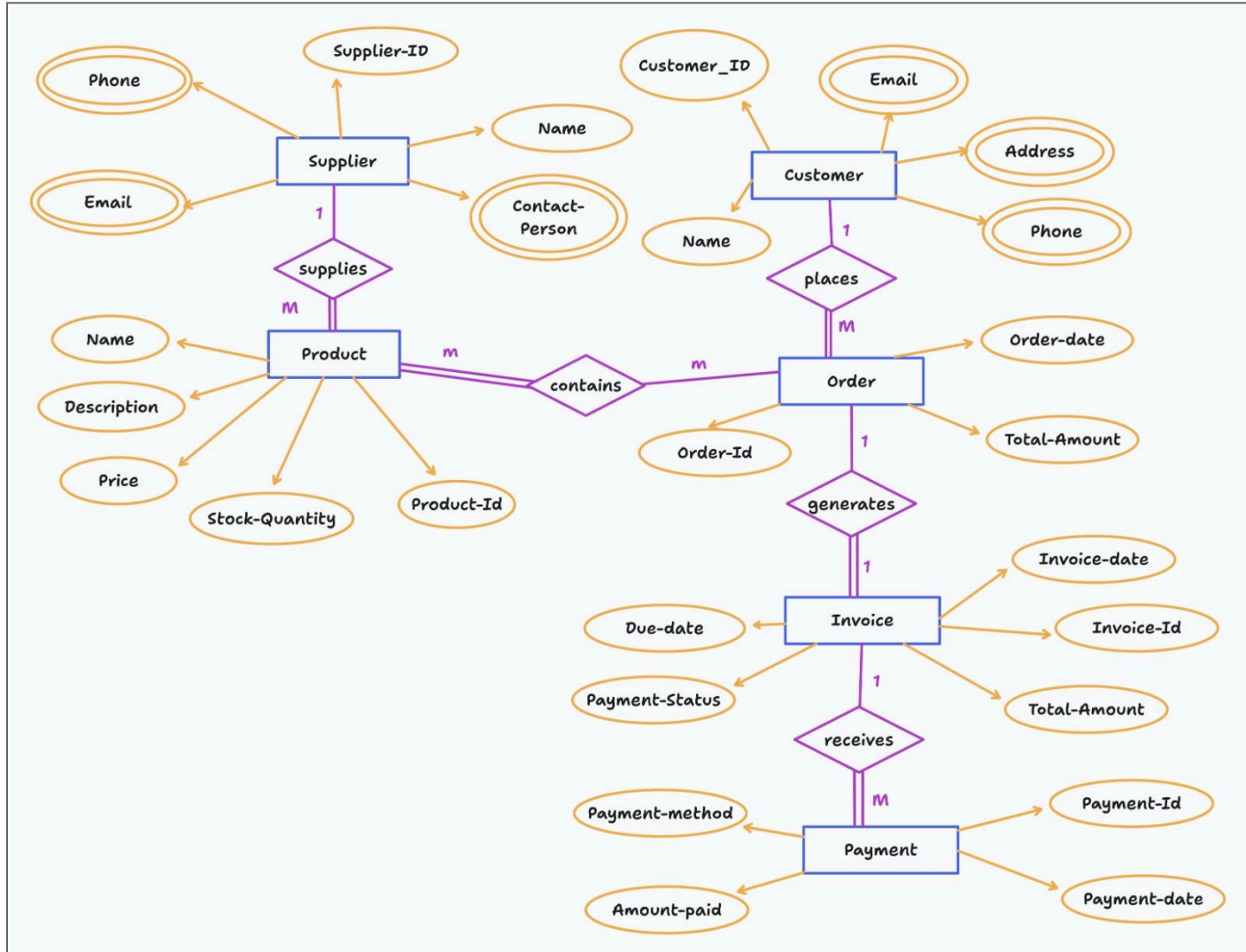
- Payment_ID (PK)
- Payment_Date
- Amount_Paid
- Payment_Method
- Invoice_ID (FK)

Relationships:

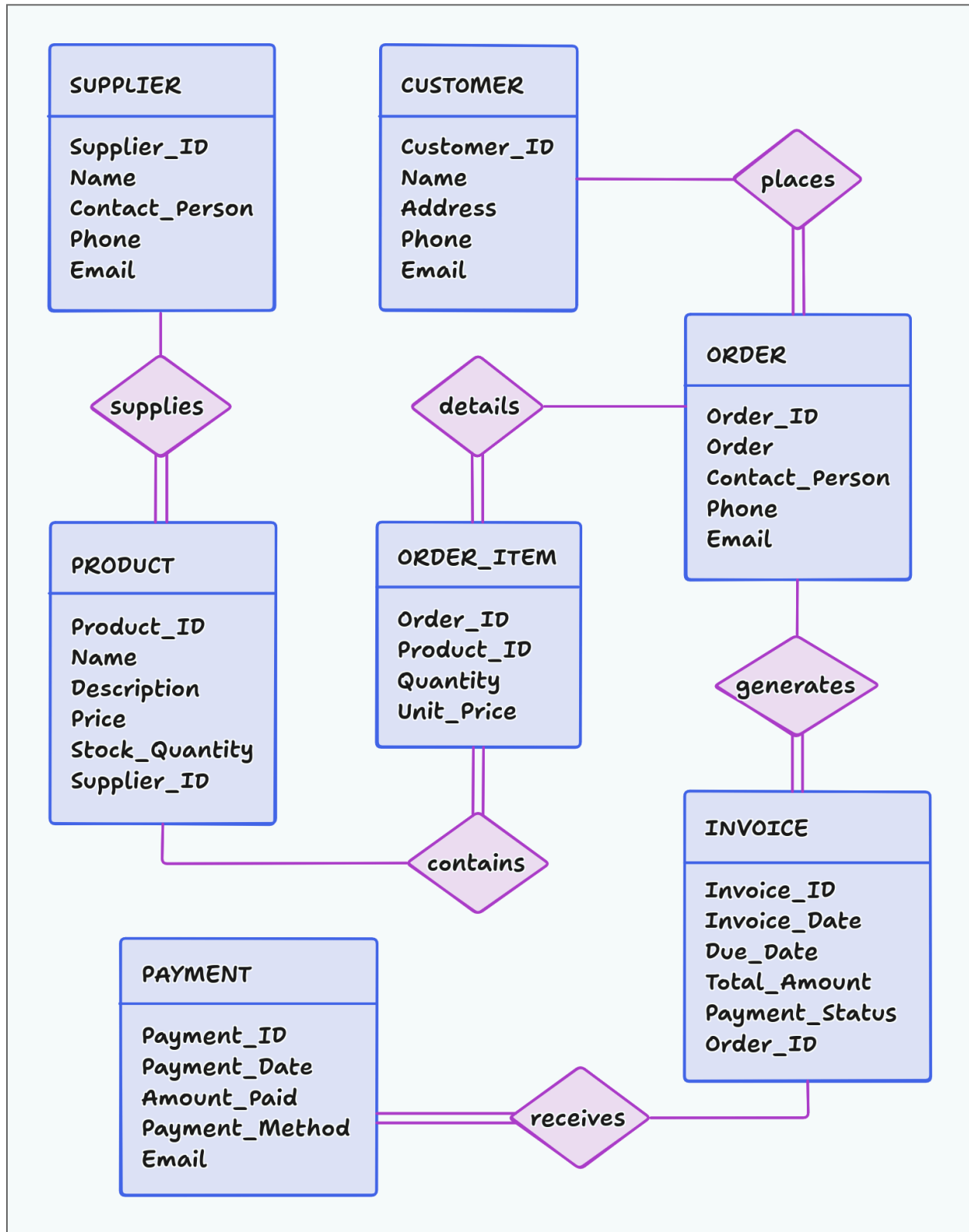
- **Customer–Order:** One-to-Many
- **Order–Order_Item:** One-to-Many
- **Product–Order_Item:** One-to-Many
- **Supplier–Product:** One-to-Many
- **Order–Invoice:** One-to-One (An invoice is generated for an order)
- **Invoice–Payment:** One-to-Many (An invoice can have multiple payments if partially paid)

ER MODEL & DIAGRAM

1. ER Diagram



2. ER Model



ER to Relational Model (Schema)

- **Product**(Product_ID [PK], Name, Description, Price, Stock_Quantity, Supplier_ID [FK])
- **Customer**(Customer_ID [PK], Name, Address, Phone, Email)
- **Supplier**(Supplier_ID [PK], Name, Contact_Person, Phone, Email)
- **Order**(Order_ID [PK], Order_Date, Total_Amount, Customer_ID [FK])
- **Order_Item**(Order_ID [FK], Product_ID [FK], Quantity, Unit_Price) – Composite PK (Order_ID, Product_ID)
- **Invoice**(Invoice_ID [PK], Invoice_Date, Due_Date, Total_Amount, Payment_Status, Order_ID [FK])
- **Payment**(Payment_ID [PK], Payment_Date, Amount_Paid, Payment_Method, Invoice_ID [FK])

Relational Algebra (RA)

Q1. Get names of all products currently in stock.

Ans. $\pi_{\text{Name}}(\sigma_{\text{Stock_Quantity} > 0}(\text{Product}))$

Q2. Find all products with a stock quantity less than 10 (low stock).

Ans. $\pi_{\text{Name}, \text{Stock_Quantity}}(\sigma_{\text{Stock_Quantity} < 10}(\text{Product}))$

Q3. Get details of all unpaid invoices.

Ans. $\sigma_{\text{Payment_Status} = \text{'Unpaid'}}(\text{Invoice})$

Q4. Find all orders placed on '2023-10-27'.

Ans. $\sigma_{\text{Order_Date} = \text{'2023-10-27'}}(\text{Order})$

Q5. List the names and contact persons of all suppliers.

Ans. $\pi_{\text{Name}, \text{Contact_Person}}(\text{Supplier})$

Q6. List all products and their supplier's name.

Ans. $\pi_{\text{Product.Name, Supplier.Name}}(\text{Product} \bowtie \text{Supplier})$

Q7. Find all orders placed by the customer named 'John Doe'.

Ans. $\pi_{\text{Order_ID, Order_Date, Total_Amount}}(\sigma_{\text{Customer.Name} = \text{'John Doe'}}(\text{Order} \bowtie \text{Customer}))$

Q8. Get the name and address of customers for Order_ID 101.

Ans. $\pi_{\text{Name, Address}}(\sigma_{\text{Order_ID} = 101}(\text{Customer} \bowtie \text{Order}))$

Q9. List all payments made for Invoice_ID 5001.

Ans. $\sigma_{\text{Invoice_ID} = 5001}(\text{Payment})$

Q10. Find the invoice details for Order_ID 205.

Ans. $\sigma_{\text{Order_ID} = 205}(\text{Invoice})$

Q11. List the products (name and price) included in Order_ID 301.

Ans. $\pi_{\text{Name, Price}}(\sigma_{\text{Order_ID} = 301}((\text{Product} \bowtie \text{Order_Item})))$

Q12. Get the name of the supplier for the product named 'Laptop'.

Ans. $\pi_{\text{Supplier.Name}}(\sigma_{\text{Product.Name} = \text{'Laptop'}}(\text{Product} \bowtie \text{Supplier}))$

Q13. List customer names who have ordered products from 'Supplier A'.

Ans. $\pi_{\text{Customer.Name}}(\sigma_{\text{Supplier.Name} = \text{'Supplier A'}}(\text{Customer} \bowtie \text{Order} \bowtie \text{Order_Item} \bowtie \text{Product} \bowtie \text{Supplier}))$

Q14. Find the names and email addresses of customers who have ordered 'Keyboard'.

Ans. $\pi_{\text{Name, Email}}(\sigma_{\text{Product.Name} = \text{'Keyboard'}}(\text{Customer} \bowtie \text{Order} \bowtie \text{Order_Item} \bowtie \text{Product}))$

Q15. Get the order date and customer name for all unpaid invoices.

Ans. $\pi_{\text{Order_Date, Name}}(\sigma_{\text{Payment_Status} = \text{'Unpaid'}}(\text{Invoice} \bowtie \text{Order} \bowtie \text{Customer}))$

Q16. List all products (names) that have been paid for by 'Credit Card'.

Ans. $\pi_{\text{Product.Name}}(\sigma_{\text{Payment_Method} = \text{'Credit Card'}}(\text{Payment} \bowtie \text{Invoice} \bowtie \text{Order} \bowtie \text{Order_Item} \bowtie \text{Product}))$

Q17. Find the contact person of the supplier for products ordered by 'Jane Smith'.

Ans. $\pi_{\text{Contact_Person}}(\sigma_{\text{Customer.Name} = \text{'Jane Smith'}}(\text{Customer} \bowtie \text{Order} \bowtie \text{Order_Item} \bowtie \text{Product} \bowtie \text{Supplier}))$

Q18. List the invoice due dates for orders containing products with a price greater than \$500.

Ans. $\pi_{\text{Due_Date}}(\sigma_{\text{Price} > 500}(\text{Product} \bowtie \text{Order_Item} \bowtie \text{Order} \bowtie \text{Invoice}))$

Q19. Get the payment dates for all orders placed by 'John Doe'.

Ans. $\pi_{\text{Payment_Date}}(\sigma_{\text{Name} = \text{'John Doe'}}(\text{Customer} \bowtie \text{Order} \bowtie \text{Invoice} \bowtie \text{Payment}))$

Q20. Find the names of suppliers whose products are on invoices that are 'Partially Paid'.

Ans. $\pi_{\text{Supplier.Name}}(\sigma_{\text{Payment_Status} = \text{'Partially Paid'}}(\text{Supplier} \bowtie \text{Product} \bowtie \text{Order_Item} \bowtie \text{Order} \bowtie \text{Invoice}))$

Q21. List the phone numbers of customers who have received an invoice with a total amount over \$200.

Ans. $\pi_{\text{Phone}}(\sigma_{\text{Total_Amount} > 200}(\text{Customer} \bowtie \text{Order} \bowtie \text{Invoice}))$

Q22. Get the description of products present in orders that have been fully paid.

Ans. $\pi_{\text{Description}}(\sigma_{\text{Payment_Status} = \text{'Paid'}}(\text{Product} \bowtie \text{Order_Item} \bowtie \text{Order} \bowtie \text{Invoice}))$

Q23. Find the addresses of customers who have made payments via 'Bank Transfer'.

Ans. $\pi_{\text{Address}}(\sigma_{\text{Payment_Method} = \text{'Bank Transfer'}}(\text{Customer} \bowtie \text{Order} \bowtie \text{Invoice} \bowtie \text{Payment}))$

Q24. List all order items (Product Name, Quantity, Unit Price) for orders placed by customers in 'New York' (assuming address contains city).

Ans. $\pi_{\text{Product.Name}, \text{Quantity}, \text{Unit_Price}}(\sigma_{\text{Address LIKE \%New York\%}}(\text{Customer} \bowtie \text{Order} \bowtie \text{Order_Item} \bowtie \text{Product}))$

Q25. Get the names of products for which payments are past their due date (Invoice_Date > Due_Date).

Ans. $\pi_{\text{Product.Name}}(\sigma_{\text{Invoice_Date} > \text{Due_Date}}(\text{Product} \bowtie \text{Order_Item} \bowtie \text{Order} \bowtie \text{Invoice}))$