Sri Sathya Sai Institute of Higher Learning

(Nandigiri Campus)

Subject: Database Management System

Title: Clothing Store Management System

Name: G. Yaswanth

Regd no: 234214

Class: BS. Computer Science

Clothing Store Management System

Problem Statement

The **Clothing Store Management System** is a database that helps a retail clothing business organize and manage its daily operations. It ensures that customers, products, orders, and payments are handled efficiently, making the business run smoothly.

Main Purpose

The system is designed to:

- Store and manage customer details.
- Keep track of clothing products available in the store.
- Record and process customer orders.
- Manage payments and transactions.
- Provide easy access to important business information.

Key Features

1. Customer Management

- Stores customer details like name, phone number, and email.
- Helps track customer purchase history.
- Makes it easy to offer personalized discounts or promotions.

2. Product Management

- Stores information about clothing items, including:
  - Product name
  - Category (shirts, pants, dresses, etc.)
  - Sizes and colours available
  - Brand and price
- Helps the store manage stock levels to avoid running out of popular products.

3. Order Management

- Keeps track of customer orders, including:
  - What products are ordered
  - Quantity of each item
  - Order date and delivery status (pending, shipped, delivered, etc.)

- Makes it easier to find customer orders and update their status.

4. Payment Management

- Records details of customer payments, including:
  - Payment amount
  - Payment method (cash, card, online payment)
  - Payment date
- Ensures secure and accurate transaction records.

5. Order Details Tracking

- Connects each order with the products purchased.
- Helps in calculating the total bill amount.
- Provides order summaries for easy reference.

Entities and Attributes

Entities:

1. **Customer**
2. **Product**
3. **Order**
4. **Order_Details**
5. **Payment**

Their Attributes:

1. Customer

The **Customer** entity stores details of people who purchase products from the clothing store.

- **Customer_ID**
- **Name**
- **Email**
- **Phone**
- **Address**

Keys & Relationships:

- Primary Key: **Customer_ID**
- Relationship: A Customer can place multiple Orders (One-to-Many).

2. Product

The **Product** entity stores information about all clothing items available in the store.

- **Product_ID**
- **Product_Name**

- **Category**
- **Size**
- **Price**
- **Stock_Quantity**

Keys & Relationships:

- **Primary Key: Product_ID**
- **Relationships:** A **Product** can be part of multiple **Orders** through **Order_Details** (**Many-to-Many**).

3. Order

The **Order** entity records purchases made by customers.

- **Order_ID**
- **Customer_ID**
- **Order_Date**
- **Total_Amount**
- **Order_Status**

Keys & Relationships:

- **Primary Key: Order_ID**
- **Foreign Key: Customer_ID** (linked to **Customer**)
- **Relationships:**
  - A **Customer** can place multiple **Orders** (**One-to-Many**).
  - An **Order** can contain multiple **Products** through **Order_Details** (**One-to-Many**).
  - Each **Order** has one **Payment** (**One-to-One**).

4. Order_Details

The **Order_Details** entity connects orders and products since an order can contain multiple products.

- **Order_Detail_ID**
- **Order_ID**
- **Product_ID**
- **Quantity**

Keys & Relationships:

- **Primary Key: Order_Detail_ID**
- **Foreign Keys:**
  - **Order_ID** (linked to **Order**)
  - **Product_ID** (linked to **Product**)
- **Relationships:**
  - Each **Order** can contain multiple **Products** (**One-to-Many**).
  - Each **Product** can be part of multiple **Orders** (**Many-to-Many**).

5. Payment

The **Payment** entity stores payment details for customer orders.

- **Payment_ID**
- **Order_ID**
- **Payment_Date**
- **Payment_Status**

Keys & Relationships:

- **Primary Key: Payment_ID**
- **Foreign Key: Order_ID** (linked to **Order**)
- **Relationship:** Each **Order** has a single **Payment** (**One-to-One**).

Schema Diagrams

Entities:

1. **Customer**
2. **Product**
3. **Order**
4. **Order_Details**
5. **Payment**

Schema Diagrams of Entities:

1. Customer

Field Name	Data Type	Constraints
Customer_ID	INT	Primary Key (PK)
Name	VARCHAR	Not Null
Phone	VARCHAR	Not Null
Address	VARCHAR	Not Null

2. Product

Field Name	Data Type	Constraints
Product_ID	INT	Primary Key (PK)
Product_Name	VARCHAR	Not Null
Category	VARCHAR	Not Null
Size	VARCHAR	Not Null
Colour	VARCHAR	Not Null
Price	FLOAT	Not Null
Stock_Quantity	INT	Not Null

3. Order

Field Name	Data Type	Constraints
Order_ID	INT	Primary Key (PK)
Customer_ID	INT	Foreign Key (FK)
Order_Date	VARCHAR	Not Null
Total_Amount	FLOAT	Not Null
Order_Status	VARCHAR	Not Null

4. Order_Details

Field Name	Data Type	Constraints
Order_Detail_ID	INT	Primary Key (PK)
Order_ID	INT	Foreign Key (FK)
Product_ID	INT	Foreign Key (FK)
Quantity	INT	Not Null

5. Payment

Field Name	Data Type	Constraints
Payment_ID	INT	Primary Key (PK)
Order_ID	INT	Foreign Key (FK)
Payment_Date	VARCHAR	Not Null
Payment_Status	VARCHAR	Not Null

ER Diagram for each Entities

1. Customer

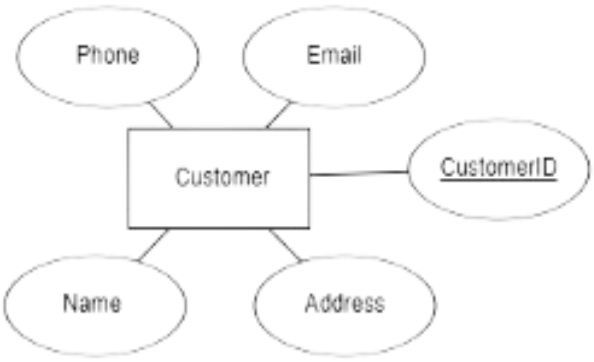
The **Customer** entity stores details of people who purchase products from the clothing store.

- **Customer_ID**
- **Name**
- **Phone**
- **Email**
- **Address**

Keys:

- Primary Key: **Customer_ID**

ER Diagram:



2. Product

The **Product** entity stores information about all clothing items available in the store.

- **Product_ID**
- **Product_Name**
- **Category**
- **Size**
- **Colour**
- **Price**
- **Stock_Quantity**

Keys:

- **Primary Key: Product_ID**

ER Diagram:

3. Order

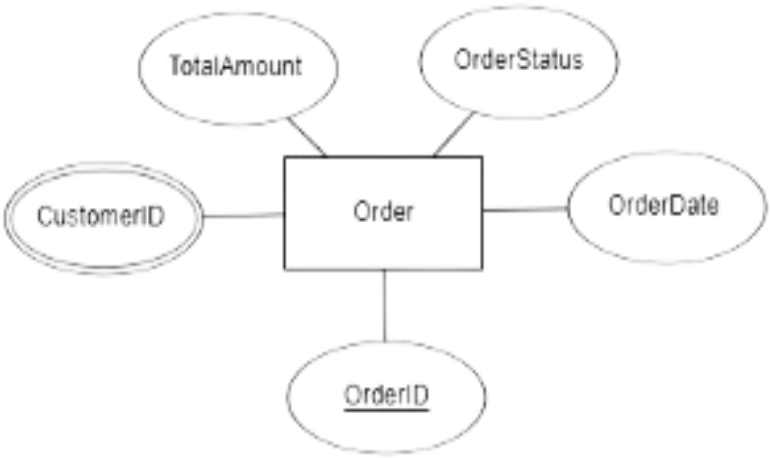
The **Order** entity records purchases made by customers.

- **Order_ID**
- **Customer_ID**
- **Order_Date**
- **Total_Amount**
- **Order_Status**

Keys:

- **Primary Key: Order_ID**
- **Foreign Key: Customer_ID** (linked to **Customer**)

ER Diagram:



4. Order Details

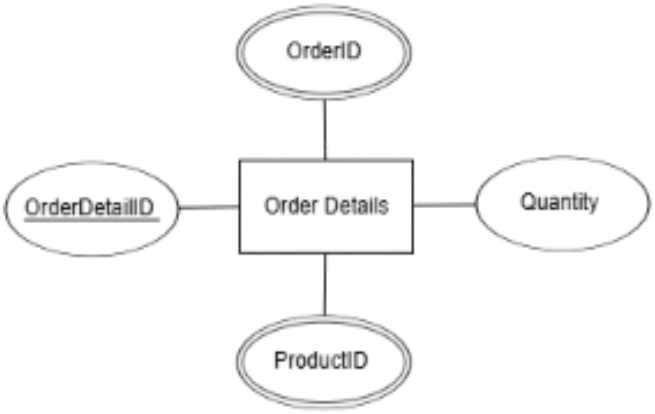
The **Order Details** entity connects orders and products since an order can contain multiple products.

- **Order_Detail_ID**
- **Order_ID**
- **Product_ID**
- **Quantity**

Keys:

- **Primary Key: Order_Detail_ID**
- **Foreign Keys:**
  - **Order_ID** (linked to **Order**)
  - **Product_ID** (linked to **Product**)

ER Diagram:



5. Payment

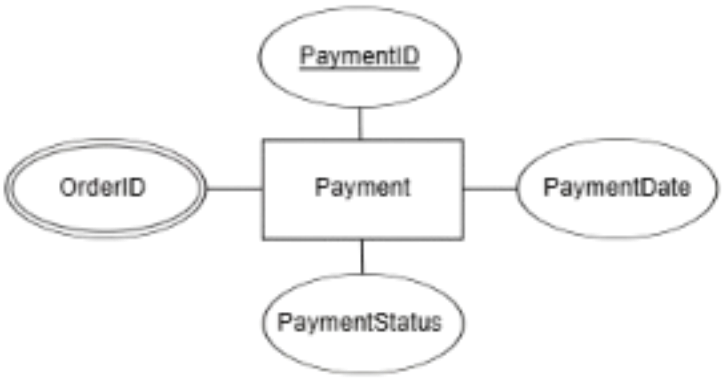
The **Payment** entity stores payment details for customer orders.

- **Payment_ID**
- **Order_ID**
- **Payment_Date**
- **Payment_Status**

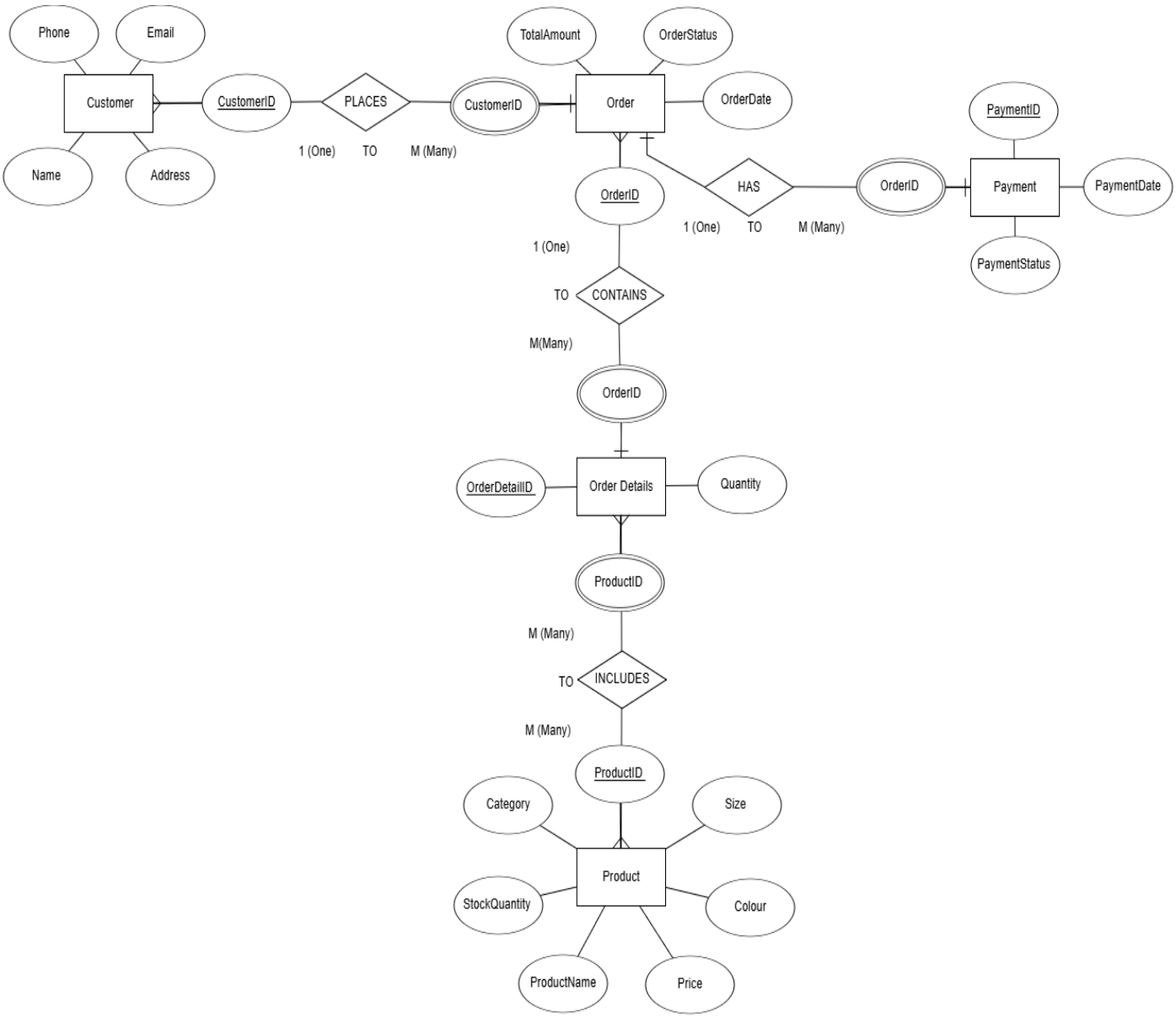
Keys:

- **Primary Key: Payment_ID**
- **Foreign Key: Order_ID** (linked to **Order**)

ER Diagram:



ER Diagram:


Phone
Email
Customer
CustomerID
PLACES
1 (One)
TO
M (Many)
CustomerID
TotalAmount
OrderStatus
Order
OrderDate
Name
Address
OrderID
HAS
1 (One)
TO
M (Many)
OrderID
PaymentID
Payment
PaymentDate
PaymentStatus
1 (One)
TO
CONTAINS
M(Many)
OrderID
OrderDetailID
Order Details
Quantity
ProductID
M (Many)
TO
INCLUDES
M (Many)
ProductID
Category
Size
Product
StockQuantity
Colour
ProductName
Price