RESTAURANT MANAGEMENT SYSTEM(RMS)

Submitted by-

Pratyaksh Chauhan (055031)

Himanshu Aggarwal (055013)

Project Overview

The Restaurant Management System (RMS) is designed to streamline restaurant operations by managing orders, reservations, customer details, staff records, and inventory efficiently. Utilizing a relational database in MySQL, the system ensures seamless data storage, retrieval, and management, improving overall service quality and customer satisfaction.

Purpose of the Report

This report provides a comprehensive analysis of the Restaurant Management System, detailing its objectives, structure, relationships, and managerial implications. It serves as a reference for restaurant managers, database administrators, and IT professionals to understand the system's design and functionality.

Objectives

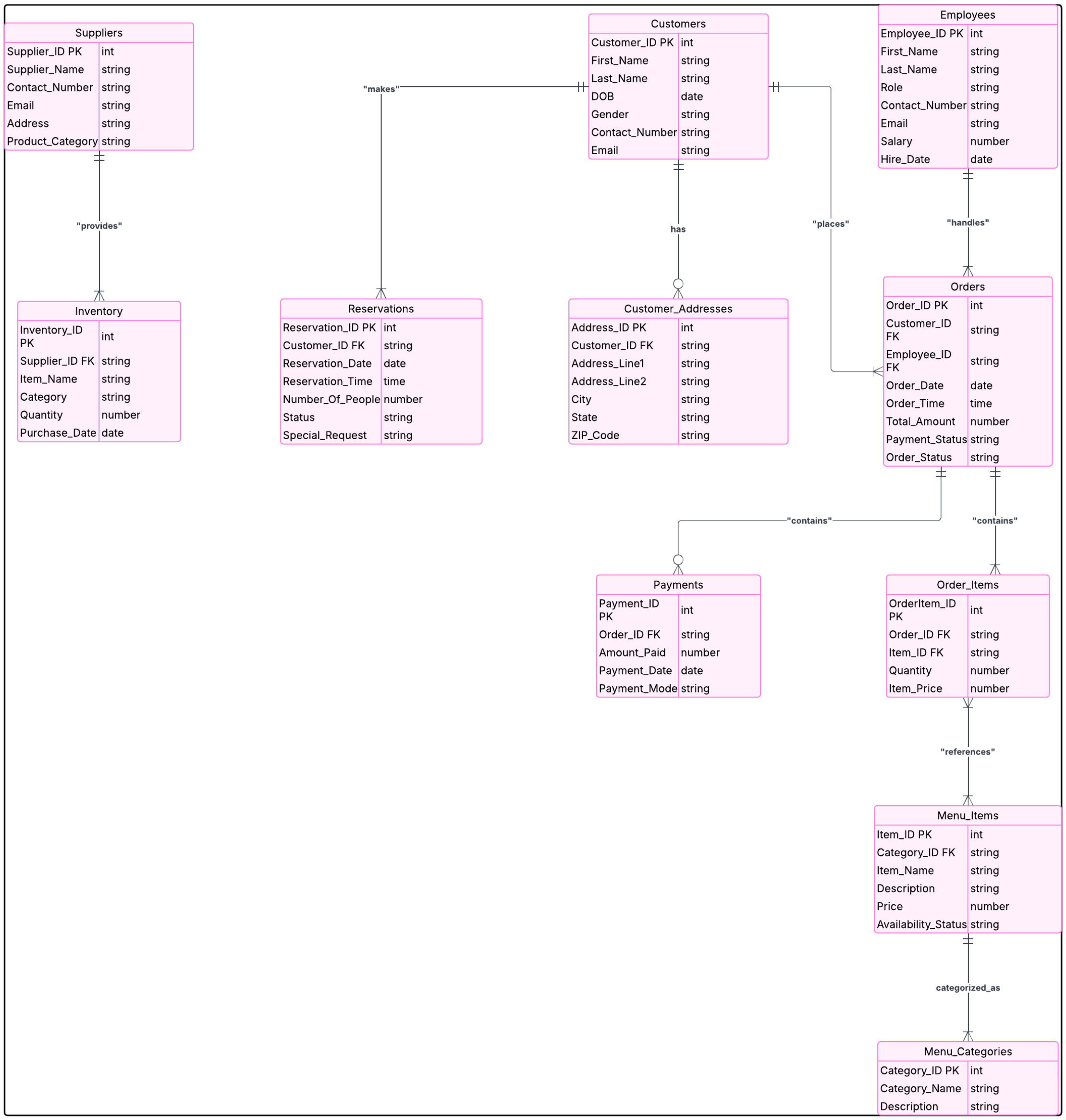
The primary objectives of the Restaurant Management System are:

- To centralize restaurant data for efficient management of operations.
- To improve customer experience through fast order processing and seamless reservation handling.
- To facilitate structured inventory management and minimize food wastage.
- To enhance staff productivity and track employee records.
- To ensure secure access control, protecting sensitive data and maintaining privacy.
- To integrate real-time analytics for better decision-making and operational efficiency.

Database Description

The database consists of 11 relational tables, each serving a specific function. The structure follows normalization principles to reduce redundancy and maintain data consistency.

ERD Diagram



Suppliers	
Supplier_ID PK	int
Supplier_Name	string
Contact_Number	string
Email	string
Address	string
Product_Category	string

Customers	
Customer_ID PK	int
First_Name	string
Last_Name	string
DOB	date
Gender	string
Contact_Number	string
Email	string

Employees	
Employee_ID PK	int
First_Name	string
Last_Name	string
Role	string
Contact_Number	string
Email	string
Salary	number
Hire_Date	date

Inventory	
Inventory_ID PK	int
Supplier_ID FK	string
Item_Name	string
Category	string
Quantity	number
Purchase_Date	date

Reservations	
Reservation_ID PK	int
Customer_ID FK	string
Reservation_Date	date
Reservation_Time	time
Number_Of_People	number
Status	string
Special_Request	string

Customer_Addresses	
Address_ID PK	int
Customer_ID FK	string
Address_Line1	string
Address_Line2	string
City	string
State	string
ZIP_Code	string

Orders	
Order_ID PK	int
Customer_ID FK	string
Employee_ID FK	string
Order_Date	date
Order_Time	time
Total_Amount	number
Payment_Status	string
Order_Status	string

Payments	
Payment_ID PK	int
Order_ID FK	string
Amount_Paid	number
Payment_Date	date
Payment_Mode	string

Order_Items	
OrderItem_ID PK	int
Order_ID FK	string
Item_ID FK	string
Quantity	number
Item_Price	number

Menu_Items	
Item_ID PK	int
Category_ID FK	string
Item_Name	string
Description	string
Price	number
Availability_Status	string

Menu_Categories	
Category_ID PK	int
Category_Name	string
Description	string

Detailed Features of the Tables

Key Features

- **Customer Management:** Stores customer details, including name, contact, and reservation history.
- **Menu and Categories:** Organizes menu items under specific categories.
- **Order Processing:** Tracks customer orders and order status in real-time.
- **Staff Management:** Maintains employee details, roles, and work schedules.
- **Inventory Management:** Tracks stock levels to optimize purchases and reduce waste.

- **Reservations:** Allows customers to book tables in advance.
- **Billing and Payments:** Manages order payments and billing records.
- **Security and Scalability:** Implements role-based access control (RBAC) and supports future system expansions.

Table Descriptions & Attributes

1. **Customers**
   - **Stores:** Personal details of customers.
   - **Primary Key:** customer_id (INT, PRIMARY KEY)
   - **Attributes:** first_name, last_name, contact_number (unique), email, reservation_history.
2. **Customer Address**
   - **Stores:** Customer address details.
   - **Primary Key:** address_id (INT, PRIMARY KEY)
   - **Attributes:** customer_id (FOREIGN KEY), street, city, state, zip_code.
3. **Menu Categories**
   - **Stores:** Categories for menu items.
   - **Primary Key:** menu_category_id (INT, PRIMARY KEY)
   - **Attributes:** category_name, description.
4. **Menu Items**
   - **Stores:** Details of menu items.
   - **Primary Key:** menu_id (INT, PRIMARY KEY)
   - **Attributes:** menu_name, menu_category_id (FOREIGN KEY), price, description.
5. **Orders**
   - **Tracks:** Customer orders.
   - **Primary Key:** order_id (INT, PRIMARY KEY)
   - **Attributes:** customer_id (FOREIGN KEY), order_date, status.
6. **Order Items**
   - **Stores:** Individual items in an order.
   - **Primary Key:** order_item_id (INT, PRIMARY KEY)
   - **Attributes:** order_id (FOREIGN KEY), menu_id (FOREIGN KEY), quantity, total_price.
7. **Reservations**
   - **Tracks:** Customer table bookings.
   - **Primary Key:** reservation_id (INT, PRIMARY KEY)
   - **Attributes:** customer_id (FOREIGN KEY), table_number, reservation_date, time_slot.
8. **Employees**
   - **Stores:** Employee records.
   - **Primary Key:** employee_id (INT, PRIMARY KEY)
   - **Attributes:** first_name, last_name, job_position, salary, contact_number.
9. **Inventory**
   - **Tracks:** Stock levels.
   - **Primary Key:** inventory_id (INT, PRIMARY KEY)

- **Attributes:** item_name, quantity_available, last_restocked.

10. **Suppliers**
    - **Stores:** Supplier details.
    - **Primary Key:** supplier_id (INT, PRIMARY KEY)
    - **Attributes:** supplier_name, contact_person, contact_number, email, supplied_items.
11. **Payments**
    - **Stores:** Payment transactions.
    - **Primary Key:** payment_id (INT, PRIMARY KEY)
    - **Attributes:** order_id (FOREIGN KEY), payment_method, amount_paid, payment_status.

Relationships in Detail

- **Customers → Orders (1:M)**
  - A customer can place multiple orders, but each order belongs to only one customer.
  - **Foreign Key:** customer_id in ORDERS references CUSTOMERS.customer_id.
- **Orders → Order Items (1:M)**
  - An order can have multiple menu items, but each order detail entry is linked to one order.
  - **Foreign Key:** order_id in ORDER_ITEMS references ORDERS.order_id.
- **Menu Items → Order Items (1:M)**
  - A menu item can appear in multiple orders, but each order detail entry refers to a single menu item.
  - **Foreign Key:** menu_id in ORDER_ITEMS references MENU_ITEMS.menu_id.
- **Menu Categories → Menu Items (1:M)**
  - A category can have multiple menu items, but each menu item belongs to one category.
  - **Foreign Key:** menu_category_id in MENU_ITEMS references MENU_CATEGORIES.menu_category_id.
- **Customers → Reservations (1:M)**
  - A customer can make multiple reservations, but each reservation belongs to only one customer.
  - **Foreign Key:** customer_id in RESERVATIONS references CUSTOMERS.customer_id.
- **Employees → Orders (1:M)**
  - An employee can process multiple orders, but each order is handled by a single employee.
  - **Foreign Key:** employee_id in ORDERS references EMPLOYEES.employee_id.
- **Suppliers → Inventory (1:M)**
  - A supplier can provide multiple inventory items, but each inventory item comes from a single supplier.
  - **Foreign Key:** supplier_id in INVENTORY references SUPPLIERS.supplier_id.
- **Orders → Payments (1:1)**
  - Each order has one payment transaction associated with it.

    - **Foreign Key:** order_id in PAYMENTS references ORDERS.order_id.

Managerial Implications

Impact on Restaurant Administration

- **Improved Efficiency:** Automated tracking of orders, reservations, and inventory reduces manual workload.
- **Better Menu Organization:** Categorization ensures optimal menu structuring and faster retrieval.
- **Compliance with Health and Safety Standards:** Proper inventory tracking reduces food wastage and ensures compliance.
- **Seamless Integration:** The database can integrate with POS systems and mobile ordering apps.

Impact on Customers

- **Personalized Dining Experience:** Order history enables better service and customized recommendations.
- **Faster Service:** Automated order processing reduces wait times.
- **Reservation Convenience:** Online booking enhances customer satisfaction.

Impact on Staff and Management

- **Optimized Workforce Management:** Staff schedules and roles can be managed efficiently.
- **Data-Driven Decision Making:** Insights from customer preferences and sales trends assist in menu optimization.
- **Enhanced Financial Management:** Real-time billing and expense tracking improve financial accuracy.

Conclusion

The Restaurant Management System is a scalable and secure solution designed to optimize restaurant operations. With real-time analytics, seamless integration, and enhanced security, it ensures improved service delivery, efficient resource management, and enhanced customer satisfaction.