

DATA AND APPLICATION

Pranjali Bishnoi (2021101038)

Mohammad Zaid (2021101004)

Vaibhav Agrawal (2021101019)

OVERVIEW

Description of the mini-world

The mini world for our project is Restaurant Management System.

All information regarding the functioning of the restaurant is stored in the database. The database model primarily consists of restaurant, storing information about them, their menu, their orders, billing, inventory management and all other related activity.

Purpose of the database

The database is made for Restaurant Management, making it easier for restaurant managers and owners to get control over the orders and all of the billing information. The database can also help the restaurant staffs to track the orders from one place, making it easier to handle multiple orders. Based on the database, the restaurant can also give smart recommendation to the customers, based on their past orders.

Users

The users of this database included restaurant managing staffs and their owners who want to get complete information about the happening of the restaurants and to track the orders of the customers.

It can also be used by the customers who want to see what all options are available at a restaurant, all at one place.

Database Requirements

Entities

Strong Entity

1. Restaurant

- Restaurant Name : VARCHAR(60) , NOT NULL
- Restaurant ID : INTEGER, PRIMARY KEY , NOT NULL
- Location : VARCHAR(60) , NOT NULL
- Contact Number : INTEGER , NOT NULL
- Restaurant Hours : VARCHAR(60)

2. Customer

- Customer Name : VARCHAR(60) , NOT NULL
- Customer Id : INTEGER, PRIMARY KEY , NOT NULL
- Contact Number : INTEGER , NOT NULL
- Orders : VARCHAR(1000)
- Restaurants Visited : VARCHAR(1000)

3. Menu

- Item Name : VARCHAR(60) , NOT NULL
- Item Id : INTEGER, Primary Key , NOT NULL
- Restaurant Id : Foreign Key , Integer , NOT NULL
- Price : FLOAT , NOT NULL
- Discount : FLOAT , NOT NULL
- Availability Status : {Available, Not Available}

4. Order (Super Class)

- Order Id : INTEGER, PRIMARY KEY , NOT NULL
- Restaurant Id : FOREIGN KEY , INTEGER , NOT NULL
- Ordered By : VARCHAR(60) , NOT NULL
- Items : VARCHAR(60), NOT NULL
- Order Type : { Delivery, Dine-In}
- Total Price : FLOAT, NOT NULL
- Order Time : VARCHAR(60)
- Customer Name : VARCHAR(60) , NOT NULL

6. Staffs (Super Class)

- Name : VARCHAR(60), NOT NULL
- Employee Id : INTEGER, PRIMARY KEY , NOT NULL
- Restaurant Id : INTEGER, FOREIGN KEY , NOT NULL
- Contact Number : INTEGER
- Date of Birth : VARCHAR(60) , NOT NULL
- Age : INTEGER

Weak Entity

Billing

- Order Id : INTEGER, FOREIGN KEY , NOT NULL
- Total Bill : FLOAT, PRIMARY KEY
- Payment Method : { Cash , Online Transaction }
- Payment Status : { Paid , Pending }

Expenditure

- Restaurant Id : INTEGER , FOREIGN KEY , NOT NULL
- Total Sell : FLOAT, NOT NULL
- Total Expenses : FLOAT , NOT NULL
- Total Profit : FLOAT , NOT NULL
- Month : VARCHAR(60) , NOT NULL

Constraints

- One Bill will be generated corresponding to each Order Id
- Each Staff can work for only one Restaurant
- One customer may order from multiple restaurants
- Discount < Price
- Payment Method can be either cash or online transaction.
- Item Id is unique over all the Restaurants

Relationships

The relationship which exists are :

- Binary Relationship OFFERS between Restaurants and Menu

One Restaurant can have multiple Menu items but any Menu item entry will belong to only one Restaurant

Cardinality : N : 1

Min-Max Ratio : Restaurants(0,N),Menu(1,1)

- Binary Relationship BILLED between Order and Billing

One Order will generate exactly, one bill, similarly any one bill be mapped to exactly one order

Cardinality : 1:1

Min-Max Ratio : Order(1,1),Billing(1,1)

- Binary Relationship WORKS_IN between Staffs and Restaurant

One Staff can work for only one Restaurant, and one Restaurant should have at least 4 staffs

Min-Max Ratio : Restaurant(4,N),Staffs(1,1)

- Binary Relationship between EXPENSE Restaurant and Expenditure

One Restaurant can generate multiple Expenditure entries , one for each month, but one Expenditure entry will be mapped to exactly one Restaurant

Cardinality Ratio : N:1

Min-Max Ratio : Restaurant(0,N) , Expenditure(1,1)

n>3 Relationship

- Customer requests Order from a Restaurant which is served by a Staff

Customer —> Order —> Restaurant —> Staff (Not an actual representation but a visualisation)

Subclass

- Order Entity is further divided into two subclasses :
 - Dine-in order
 - Delivery order
- Further the Staffs Entity can be further divided into following subclasses :

- Managers
- Waiter
- Cook
- Cleaning Services

Attribute Type

- Multivalued Attribute :
 - Orders attribute of Customer can be more than one
 - Restaurants Visited of Customer can be more than one
- Derived Attribute :
 - Total Profit attribute of Expenditure can be derived from Total Sell and Total Expenses
 - Age attribute of Staffs can be derived from Date of Birth
- Composite Attribute :
 - Location attribute of Restaurant can be broken down into District, City and State
 - Customer Name attribute of Customer and Name attribute of Staffs can be broken down into First Name, Middle Name and Last Name.

Functional Requirements

- Insert :
 - Insertion of new Restaurant entries
- Constraints :
- Contact Number should consist of 10 digits
 - Restaurant Name should be non empty
- Insertion of new Menu entries

Constraints :

- Item Name should be non empty
 - Price should be a positive quantity
 - Discount should be a positive quantity
 - Discount should be less than Price
- New Order entry generated

Constraints :

- Order Type can be either Delivery or Dine-in
 - Item Name and Item Id should be non empty and should refer to an entry in the Menu Table
- New Staffs entry generated

Constraints :

- Date of Birth and Age should be in coordination
 - Name should be non empty
 - Payment Method can be either cash or online transaction
 - Payment Status can be either paid or pending
- Delete :
 - Deletion of staff members in case of resignation
 - Deletion of a Menu entry in case it is discontinued from the Restaurant
 - Update :
 - Updating Restaurant Hours attribute of a Restaurant entry case it changes
 - Updating Restaurants Visited attribute of a Customer entry in case the customer visits a new restaurant
 - Updating Price and Discount attribute of a Menu entry in case it changes
 - Retrieval :
 - Selection

- List of all staff members for a given Restaurant
- Projection
 - Getting all Restaurants in which a given Item Name is available and Price is lesser than a given quantity.

Example : Search for Restaurants having Burger in less than 50 INR
- Aggregate
 - Maximum profitable month of a Restaurant in a given year
 - Total Profit of a Restaurant in a given Year
- Search
 - Get Order details by Customer Name and Restaurant Name
- Reports generated
 - Monthly financial report of a given Restaurant
 - Top dishes report of a given Restaurant for a month