INTERNATIONAL INSTITUTE OF
INFORMATION TECHNOLOGY
HYDERABAD

SANJI’S BARATIE
A "ONE PIECE" INSPIRED MINI WORLD

DATA AND APPLICATIONS - HOMEWORK 1

TEAM 16

Ketaki Shetye, 2022114013
Maitreya Chitale, 2022114011
Prabhav Shetty, 2022111015
Sujal Deoda, 2022115001
Vinit Mehta, 2022111001

Contents

1 Introduction

Step into "Sanji's Baratie" a mini world that celebrates the culinary genius of Sanji from "One Piece." Explore a meticulously crafted database that captures the essence of his restaurant on an enchanting island. Experience the adventure, camaraderie, and unforgettable flavors of the "One Piece" universe through exquisite dishes, loyal customers, and dedicated staff. Delve into a tapestry of flavors and embark on an epicurean journey that blends fine dining with the spirit of "One Piece." Bon appétit!

2 Objective

The database aims to fulfil the following purposes:

1. **Data Retrieval:** It would allow the user to retrieve particular data of interest from very large collection of data, all of which is not needed for a particular task.
2. **Data Integrity:** Ensures that data is accurate, consistent, and reliable.
3. **Data Security:** Databases offer security features to protect sensitive information. Access control mechanisms ensure that only authorized users can view or modify data, and encryption can be used to protect data at rest and in transit.
4. **Data Analysis:** User can use the all the data stored to study patterns to derive insights, generate reports and make decisions for the future of the restaurant.
5. **Data Maintenance:** The user can easily update, insert or delete data from the database without much hassle to manage integrity and consistency (database software takes care of it).
6. **Application Integration:** Databases are often integrated with various software applications, allowing these applications to store and retrieve data seamlessly.

3 Users

The users of the "Sanji's Baratie" database are:

1. **Sanji (Restaurant Owner and Chef):** Sanji himself would use the database to manage the restaurant's menu, track customer preferences, monitor orders, and oversee the performance of his staff. He could use it to refine his culinary offerings and enhance the restaurant's overall experience.
2. **Restaurant Staff:** The waitstaff and kitchen staff would use the database to fulfill customer orders, update the menu as needed, and coordinate their roles within the restaurant.
3. **Customers:** Frequent patrons or visitors to Sanji's restaurant could use the database to view the menu, place orders, and provide feedback or reviews on their dining experiences.
4. **Restaurant Managers:** If there are managers or supervisors overseeing the restaurant's operations, they could use the database to monitor staff performance, track sales, and make managerial decisions. They can also use this data to check what all raw materials vegetables and spices would be required and place order accordingly. They can also keep track of staff's rating and based on the rating can increment or decrement some staff persons salary or recruit/fire staff.
5. **Potential Investors or Partners:** If Sanji's restaurant were seeking investments or partnerships, potential stakeholders might use the database to assess the restaurant's performance and prospects.

4 Applicatons

This database stores all the information about the restaurant related to its staff, reviews, orders, bills, feedbacks, grocery, salary etc. which all can be used to run the restaurant efficiently and ensure good service to the customers. Some software can extract out the reviews information from the database and show the overall rating of the restaurant to the customers so that they can take their decision based on this. The managers and head chefs can take a peek on the storage of vegetables, spices and all and decide when do they want to place some order. The manager can decide whether give some increment/decrement or recruit/fire some staff (based on their customer feedback). The information about the revenue being generated can be inferred.

5 Database Requirements

5.1 Assumptions

1. Purchase order is placed for one Ingredient at a time with vendor (that is bill for purchase of each Ingredient from vendor (maybe same) are separate).
2. All the attributes are NOT NULL unless specified specifically.
3. Assuming that each customer places just one order. If the same customer places another order than it is treated as a separate customer.
4. Each customer can take up/reserve only one table and not multiple tables.

5.2 Strong Entity Types

Note: Attributes in red color serves as primary key for that entity type.
Red Colored attributes : Primary Key
Green Colored attributes : Foreign Key

1. Employee
   Stores the details of all the employees (cooks, waiters, cleaning staff and managers) working in the restaurant.
   a. Employee Id
      - Data Type : Integer
   b. Name (Composite Attribute : First Name + Middle Name + Last Name)
      - Data Type : Varchar(String)
   c. Age (Derivable)
      - Data Type : Integer
      - Must be >= 18
      - Age = Curr Date - DOB
   d. DOB
      - Data Type : Date
   e. Gender
      - Data Type : ENUM
   f. Address (Complex) (Composite Attribute : Apt no + Street no + area + city + state)
      - Data Type : Varchar(String)
   g. Contact No
      - Data Type : Integer
      - 10 digits
   h. Role
      - Data Type : ENUM
   i. Start Date
      - Data Type : Date
   j. End Date

        - Data Type : Date
        - NULLABLE

    k. Salary
        - Data Type : Integer
        - Currency : Beli

2. Customer

   Stores the details of all the customers that visit the restaurant.

    a. Customer Id
        - Data Type : Integer

    b. Name (Composite Attribute : First Name + Middle Name + Last Name)
        - Data Type : Varchar(String)

    c. Contact No
        - Data Type : Integer
        - 10 digits

    d. Favourite Dishes (Multivalued)
        - Data Type : Varchar(String of comma separated values)

3. Vendor

   Stores the details of all the vendors from which the restaurant purchases the ingredients.

    a. Vendor Id
        - Data Type : Integer

    b. Name (Composite Attribute : First Name + Middle Name + Last Name)
        - Data Type : Varchar(String)

    c. Contact No
        - Data Type : Integer
        - 10 digits

    d. Address (Complex) (Composite Attribute : Apt no + Street no + area + city + state)
        - Data Type : Varchar(String)

4. Order

   Stores information about individual orders.

    a. Order Id
        - Data Type : Integer

    b. Date and time
        - Data Type : Date-Time

    c. Customer Id
        - Data Type : Integer

    d. Dishes (Multivalued)
        - Data Type : Varchar(String of comma separated valued)

    e. Table No
        - Data Type : Integer

5. Dishes

   Stores information about the dishes being served, ingredients required in the dishes, cuisine of the dishes and the cook which prepares the dish.

    a. Dish Id
        - Data Type : Integer

    b. Name
        - Data Type : Varchar(String)

    c. Ingredient Id
        - Data Type : Integer

d. Employee Id (Id of cook)
   - Data Type : Integer

e. Cuisine
   - Data Type : ENUM

f. Price
   - Data Type : Decimal
   - Upto 2 decimal places
   - Currency : Beli

g. Discount
   - Data Type : Decimal
   - Upto 2 decimal places
   - Unit : %

6. Inventory
   Stores information about availability of ingredients and in what quantities.
   a. Ingredient Id
      - Data Type : Integer
   b. Name
      - Data Type : Varchar(String)
   c. Vendor Id
      - Data Type : Integer
   d. Quantity
      - Data Type : Integer
      - Unit : Kg/pcs (based on item)
   e. Price
      - Data Type : Decimal
      - Upto 2 places precision
      - Currency : Beli

7. Bills
   Stores information about bills of all orders placed by customers.
   a. Bill Id
      - Data Type : Integer
   b. Order Id
      - Data Type : Integer
   c. Customer Id
      - Data Type : Integer
   d. Total Amount
      - Data Type : Decimal
      - Upto 2 places of precision
   e. Date and time
      - Data Type : Date-Time
   f. Table No
      - Data Type : Integer
   g. Mode Of Payment
      - Data Type : ENUM

8. Tables
   Stores Information about tables.
   a. Table No

        - Data Type : Integer

    b. Capacity
        - Data Type : Integer

    c. Used
        - Data Type : ENUM

    d. Reserved
        - Data Type : ENUM

9. Restock Orders

   Stores information about the orders placed with vendors for the purchase of ingredients.

    a. Restock Order Id
        - Data Type : Integer

    b. Ingredient Id
        - Data Type : Integer

    c. Vendor Id
        - Data Type : Integer

    d. Quantity
        - Data Type : Integer

    e. Total Cost
        - Data Type : Decimal
        - With 2 digits of precison
        - Currency : Beli

    f. Paid
        - Data Type : ENUM

5.3 Weak Entity Types

1. Feedback

   Stores Information about the feedbacks given by customers for their orders.

    a. Customer Id
        - Data Type : Integer

    b. Rating
        - Data Type : ENUM

    c. Date
        - Data Type : Date

    d. Order Id
        - Data Type : Integer

    e. Review
        - Data Type : Varchar(String)

2. Reservations

   Stores information about the tables reserved by customers.

    a. Customer Id
        - Data Type : Integer

    b. Table No
        - Data Type : Integer

    c. From
        - Data Type : Date-Time

    d. Till
        - Data Type : Date-Time

- NULLABLE

5.4 Relationship Types

5.4.1 Binary Relationships

- Waiter **SERVES** the Order
  - Participating entities : Waiter (Employee), Order
  - Cardinality Ratio : 1:N
- Chef **PREPARES** the Order
  - Participating entities : Chef (Employee), Order
  - Cardinality Ratio : N:M
- Vendor **SELLS** Ingredients
  - Participating entities : Vendor, Ingredient
  - Cardinality Ratio : 1:N
- Manager **MANAGES** Inventory
  - Participating entities : Manager, Inventory
  - Cardinality Ratio : 1:1
- Manager **OVERSEES** Employees
  - Participating entities : Manager, Employee
  - Cardinality Ratio : 1:N
- Cook **REPORTS** to Manager
  - Participating entities : Cook, Manager
  - Cardinality Ratio : N:1
- Waiter **REPORTS** to Manager
  - Participating entities : Waiter, Manager
  - Cardinality Ratio : N:1
- Cleaning Staff **REPORTS** to Manager
  - Participating entities : Cleaning Staff, Manager
  - Cardinality Ratio : N:1
- Customer **RESERVES** Table
  - Participating entities : Customer, Table
  - Cardinality Ratio : 1:1
- Customer **SITS** on a Table
  - Participating entities : Customer, Table
  - Cardinality Ratio : 1:1
- Dish **REQUIRES** Ingredient
  - Participating entities : Dish, Ingredient
  - Cardinality Ratio : N:M
- Customer **RATES** Order
  - Participating entities : Customer, Order
  - Cardinality Ratio : 1:1

5.4.2 Degree $>$ 2 relationship types

- Customer **PAYS** Bill for an Order and **GIVES** Feedback
  - Participating entities : Customer, Bill, Order, Feedback
  - Cardinality Ratio : 1:1:1:1
- Manager **PLACES** Restock Order with Vendor

- Participating entities : Manager, Restock Order, Vendor
- Cardinality Ratio : 1:M:N

6 Functional Requirements

6.1 MODIFICATION

1. Insert
   (a) Adding entry of new customer, along with the order placed by the customer, bill paid and all corresponding details.
   (b) Adding a new dish to the menu.
   (c) Adding new vendor for some new ingredient.
   (d) Recruiting new employees.
2. Delete
   (a) Deleting a dish from menu.
3. Update
   (a) Update the price of some dish/ingredient.
   (b) Update the status of payment for order placed with vendors.
   (c) Updating salary of some employee etc.

6.2 RETRIEVALS

1. Selection
   (a) Select all orders with a particular dish.
   (b) Select all employees with join date before some particular date.
2. Analysis
   (a) Get the average salary of all the employees.
   (b) Decline/Increment in the number of customers.
3. Aggregate
   (a) Get the total earnings for some day/month/year.
   (b) Get the total expenditure on ingredients.

7 Summary

"Sanji's Baratie" is a fictional restaurant database inspired by the culinary world of Sanji based on the “One Piece" series. It envisions a meticulously crafted database that captures the essence of Sanji's restaurant on an enchanting island. The primary objective of this database is to facilitate data retrieval, ensure data integrity and security, data analysis, maintenance, and application integration. Users of the database include Sanji, restaurant staff, customers, managers, and potential investors/partners. The database stores information related to employees, customers, vendors, orders, dishes, ingredients, inventory, bills, tables, restock orders, feedback, and reservations. It maintains strong entity types, such as employees, customers, vendors, orders, dishes, inventory, bills, tables, restock orders, and feedback. Additionally, it defines various binary and degree > 2 relationship types to link these entities. Functional requirements encompass data modification and retrieval tasks. Users can insert new customer and order details, add dishes to the menu, include new vendors, and recruit employees. Deletion involves removing dishes from the menu, and updates allow for changes in dish/ingredient prices, vendor payment statuses, and employee salaries. Retrievals include data selection, analysis, and aggregation to generate insights, such as calculating average salaries and tracking financial performance.
Thus, "Sanji's Baratie" database aims to provide a comprehensive and efficient platform for managing a restaurant's operations, from customer interactions to menu items, inventory, and financial transactions, while embracing the spirit of the "One Piece" universe.