PROBLEM STATEMENT: PHASE 2

Restaurant Management System

We have chosen this topic because one of us run a restaurant in India. Having seen and being part of its operations we feel that this would be an ideal topic since we understand the business well. The project will be an all in one restaurant software which contains the food and beverages offered, list of employees, list of customers. The software can be used to monitor which table is being addressed by which waiter, list of food and beverages ordered by each table etc. It should be able to generate a cost total of each table at the time of billing. It should be available for employees, customers and the management (directors, shareholders, investors). We are also considering of adding table reservation, home delivery by assigning an area code for ach delivery boy and in the future adding order online feature too.

The rest of the article describes the functionality and the data it would hold when it is completed.

Data description: (brief description, might be subjected to modification)

Item (food and beverage):

ItemId: unique id for each item on the menu.

ItemName: name of the item on the menu.

ItemType: type of the item on the menu, for eg. Starters, curries, breads, beverages etc.

ItemCost: price of the item on the menu.

ItemCat: category of the item on the menu, for eg. Veg or non-veg.

IngName: Contains the names of the core ingredients used in the dish.

Employee (super class):

EmpId: unique id for every employee of the restaurant.

EmpName: first name of the employee.

Address: address of the employee.

PNumber: phone number of the employee.

Salary: salary of the employee.

Leave: Holds the start and end date of leave. This provides the length of the employee’s absence.

Waiters (predicate defined sub class):

This sub classed entity holds data of all the employees who works specifically inside the restaurant. They do not deal with the deliveries at all.

Delivery - Boys (predicate defined sub class):

This sub classed entity holds data of all the employees who work specifically on delivery

DelArea: Area name for delivery.

Chefs (predicate defined sub class)

Chef will share the same attributes as the employee, it is a predicate defined sub class (designation).

Customer:

CId: unique id for every customer.

CustName: first name of the customer.

Address: address of the customer.

PNumber: phone number of the customer.

Birthday (DOB): birthday of the customer. This field is used in order to offer some special discounts to the customer.

Anniversary: anniversary of the customer. This field is used in order to offer some special discounts to the customer.

Dine-In customer(predicate defined sub class):

Customers who orders items at the restaurant.

Door Delivery customer (predicate defined sub class):

Customers who want items to be delivered.

Inventory:

IngId: Unique ID for each ingredient.

IngredientName: Name of the ingredient.

Quantity: Quantity of the ingredient.

Supplier:

SupplierId: Supplier identity.

SupplierName: Name of the supplier.

SupplierContact: Contact details of the supplier.

SupplierCost: Total cost of the items supplied by the supplier.

Table:

TableNo. : The table number for waiters reference.

Capacity: Number of people the table can accommodate.

EmpID: The ID of the employee who is assigned to the table.

Entity Description:

1. Item: This entity is a model of the physical menu offered at the restaurant, its primary key is ItemID. We have identified this as an entity because it has an independent existence and has characteristics that define it.

2. Customer: This entity models the customers visiting the restaurant. CustId is used to identify a customer, it is used as a primary key.
   CustName: Customer Name is stored as a composite attribute containing first name and last name. The customer super class entity has two predicate sub classes dine-in customers and Door Delivery customers.

3. Employee: This entity models the employees of the restaurant. Its primary key is EmpId.
   EmpName: Name of the employee is stored as a composite attribute containing First name and last name.

Leave: It is a multivalued composite attribute which will indicate the leave duration for an employee.

4. Inventory: This entity models the data about the ingredients used for the items.IngId is used as primary key to identify the ingredient.

5. Supplier: This entity models suppliers for the various ingredients used in the restaurant. SupplierId is used as primary key.

6. Table: This entity models the tables used in the restaurant. TableNo is used as a primary key to identify a table.

DESIGN DESCRIPTION.

Books: This is a relationship that relates customer and table entity (i.e) Customer **BOOKS** a table. It has book as multivalued composite attribute consisting of date and time. A customer may book one or more tables or may not book a table on a specific day at different times. A table may be booked by only one customer at a time. We are assuming that there are twenty tables so max value is 20.

Orders: This relationship relates the customer and the item entity (i.e) Customer **ORDERS** an item. It has orderdate and ordertype as attribute. A customer may order one or more items and an item may or may not be ordered by a customer. The assumption made is that the customer makes only one delivery order a day.

Manages: This relates employee and inventory entities (i.e) Employee **MANAGES** the Inventory. The manager manages the inventory. The assumption made is that our model has one inventory managed by one manager.

Supervision: This is a recursive relationship .Employee **SUPERVISES** other Employees. The assumption made is one employee who is a supervisor supervises all the other employees.

Negotiation: This is a relationship between employee and supplier.(i.e) Employee **NEGOTIATES** with Suppliers. The assumption made is that the manager may negotiate with one or many suppliers.

Cooks: This is a relationship between employee (chef) and items (i.e) Chef **COOKS** an item. The assumption made is that an item may be cooked by one or many chefs at different times and a chef may cook many items.

Delivers: This is a relationship between employee (delivery boy) and door delivery customer (i.e) Delivery Boy delivers to the Door delivery customer. The delivery is made to a particular customer on a particular day, the customer is identified using the customer ID. The assumption made is that one delivery boy is assigned to one area and he can deliver to many customers in that area. The customers in one area will get deliveries from only one delivery boy.

Serves: This is a relationship between employee (waiter) and customer (i.e) Waiter **SERVES** customer. The assumption made is that one waiter can serve one or more customers and a customer has to be served by only one waiter.

Generate bill: This is a ternary relationship between employee, item and customer. The bill is generated for each customer for an order made in a day, assuming that a customer can make one or more dine-in orders a day and only one delivery order a day.

Supplies: This is a relationship between supplier and inventory (i.e) supplier **SUPPLIES** to the inventory. It has a multivalued composite attribute which specifies the order date and the delivery date. The assumption made is suppliers can supply many ingredients and one ingredient can be supplied by many suppliers.

BUSNIESS GOALS:

Based on the data inserted we will be able to find out the following:

1. The most ordered dish : This report can be generated from the data captured.
2. The least ordered dish : This report can be generated by using the data capture.
3. Based on whether the most ordered item we can identify the best chef : This report can be generated using the data in the item, employee (chef) and the bill generated.
4. We can identify the most crowded day of the month : This report can be generated by using the number of bills generated on a day.
5. We can identify the most profitable day of the month : This report can be generated by summing up all the bill amounts generated for each day of the month.
6. We can also identify the area placing the most number of home delivery orders and concentrate on improving faster food delivery in these areas.

7. Based on the birthdays and anniversary dates collected from the customer table we can give them special discounts on these days and attract more customers.
8. Based on the total monthly income made and the salary given to each employee, cost of supplies we can generate profit reports.
9. Important decisions of either hiring or firing employees can be taken based on the profit reports.
10. Based on the Supplier data, we can identify frequency of late delivery.
11. Based on the least ordered dish identified, we can decrease the quantity of ingredients ordered.
12. Based on the most ordered dish identified, we can increase the quantity of ingredients ordered.
13. We can identify the most frequent customer.
14. Based on the frequency of delays we can terminate the contract.
15. Based on the leave duration of the employees we can assign tasks to other employees.