PROJECT PHASE - I

Team Name : PVP

Team Members :

1) P.Lokesh (2019101062)
2) P.Varun (2019101114)
3) T.Puneeth (2019101064)

INTRODUCTION TO MINI WORLD

The goal of this project is to provide a detailed database for easy administration of a Food court at a university. The food court provides an option for the students and the staff of the university to pay all their bills of a week at once. An account for all members of the university is generated to pay weekly bills. The food court contains a variety of stalls .Details of each stall are also maintained separately.

DATABASE REQUIREMENTS

1) ENTITY TYPES :

a) Customer - Details of a customer.

b) Stall- Details of a stall in the food court.

c) Stall Maintenance - Details about maintenance of the stall.

d) Employee - Details of a employee working at the food court.

e) Dependant – Details of the visitor.

f) Bill - Details of a bill bought at a stall.

Entity Type - Customer

1) Name(VARCHAR(100),NOT NULL) : The name of the student/staff - Composite Attribute containing First name and Last name.

2) Gender(CHAR(),NOT NULL) : The gender of the customer - Simple Attribute{M,F}.

3) Customer ID(INT(10), NOT NULL) : ID of the customer - Key Attribute.

4) Role(VARCHAR(20),NOT NULL) : The role of the customer in the university - Simple Attribute.

Role has overlapping subclasses such as Students, TA’s, Professors, Administration workers.

5) Email ID(VARCHAR(50), NOT NULL) : Email ID of the customer – Simple Attribute

6) Phone Number(VARCHAR(50), NOT NULL) : Phone Number of a customer – Multi Valued Attribute (One or more phone numbers for communication)

7) Amount(INT(10), NOT NULL) – Amount to be paid – Simple Attribute.

Entity Type – Stall

1) Stall Name(VARCHAR(50), NOT NULL) : Name of the stall – Simple Attribute

2) Stall Id(INT(10), NOT NULL) : The ID of the stall – Key Attribute

3) Stall Timings(VARCHAR(500), NOT NULL) : The timings of stall – Composite Attribute (Composed of the opening time and closing time).

4)Menu(VARCHAR(5000), NOT NULL): The menu of the stall – Nested composite attribute{breakfast,lunch,dinner}.

5) Duration(INT(10), NOT NULL) : The number of hours for which the stall is open in a day – Derived Attribute(Derived from Stall timings attribute).

Entity Type – Stall Maintenance (WEAK ENTITY)

1) Stall ID(INT(10), NOT NULL) : ID of the stall for which we store details – Simple Attribute.

2) Date(VARCHAR(50), NOT NULL) : The date on which the stall maintenance details of the month are inserted – Simple Attribute.

3) Income(INT(10), NOT NULL) : Income in a month – Simple Attribute.

4) Expenditure(INT(10), NOT NULL) : Expenses in a month – Simple Attribute.

Entity Type – **Employee**

1) Name(VARCHAR(50), NOT NULL) : The name of the employee – Simple Attribute.

2) Gender(CHAR(), NOT NULL) : The gender of the employee – Simple Attribute{M,F}.

3) Employee ID(INT(10), NOT NULL) : Unique ID of the employee – Key Attribute.

4) Stall ID(INT(10), NOT NULL) : The Stall ID of the stall in which the employee works – Simple Attribute.

5) Role(VARCHAR(50), NOT NULL) - The kind of job he is assigned in the stall which he works- Simple Attribute

Contains subclasses – Chef, Server, Cleaner, Manager, Accountant.

This is a disjoint set of subclasses.

6) Salary(INT(10), NOT NULL) : Salary of the employee per month – Simple Attribute.

7) Working Days(INT(10), NOT NULL) : Number of days working in a month – Simple Attribute.

Entity Type – **Dependent** (WEAK ENTITY)

1) ID(INT(10), NOT NULL) : Customer ID of the member of University who accompanies the dependent – Simple Attribute.

2) Dependent Name(VARCHAR(50), NOT NULL) : Name of the Dependent – Simple Attribute.

3) Dependent Gender(CHAR(), NOT NULL) : Gender of the dependent – Simple Attribute.

Entity Type – Bill

1) Customer ID(INT(10), NOT NULL) : The Customer Id of the person who is paying the bill – Simple Attribute.

2) Name(VARCHAR(50), NOT NULL) : The name of person who bought the bill(Can be the dependent name also) – Simple Attribute.

3) Stall ID(INT(10), NOT NULL) : From which stall the bill is bought – Simple Attribute.

4) Amount(INT(10), NOT NULL) : The amount on the bill – Simple Attribute.

5) Date and Time(VARCHAR(50), NOT NULL) : The date and time on which the bill is bought – Composite Attribute (Composed of date and time)

6) Order Number(VARCHAR(50), NOT NULL) : The combination of Stall ID and Date and time is a Composite Key Attribute.

2) RELATIONSHIP TYPES :

1) **Eats_in** : Customer Entity type is in Many to Many Relationship with Stall Entity type. A customer can eat in several stalls and a stall can serve several customers.

Can be read as Customer X Eats_in Stall Y.

Cardinality ratio – M:N.

Participation constraints – Both the customer and stall entity types are in partial participation .

2) **IsRelatedTo** : Customer entity type is in Many to One Relationship with Dependent Entity Type. Each customer can have many dependents, but a dependent is related to only one customer.

Can be read as Customer X IsRelatedTo Dependent Y.

Cardinality Ratio – 1:N.

Participation constraints – Customer is in partial participation and Dependents is in Total participation.

3) **WorksFor** : Employee entity type has One to Many Relationship with Stall entity type as the number of employees working in a particular stall can be greater than 1, but an Employee works in at most one stall.

Can be read as Employee X works for Stall Y.

Cardinality Ratio – N:1

Participation constraints – Total participation of both the entity types.

4) **MaintainsRecordsof** : StallMaintenance entity type is in one to many relationship with Stall entity type. A Stall entity is maintained by several Stall Maintenance entities, but a stall maintenance entity keeps record of only one stall entity.

Can be read as StallMaintenace X MaintainsRecordsof Stall Y.

Cardinality Ratio - N:1

Participation Constraints – Total participation of both the entity types.

5) **ExpenditureCalculation** : Expenditure of a Stall maintenance entity is maintained by adding the salary of the employees and the bills produced in a particular stall.

Cardinality Ratio –

Stall maintenance – One

Employees – Many

Bills – Many

Stall - One

Participation constraints –

Stall maintenance – Total

Employees – Total

Bills – Partial

Stall – Total

6) **Manages** : Employee entity type(Manager) is in many to one relationship with Employee entity type. A Manager manages salary of many employees , an employee's salary is managed by only one manager.

A Manager manages the salary of employees in the role of an employee.

Can be read as Employee X Manages salary of Employee Y.

Cardinality Ratio – 1:N

Participation constraints – The First Employee Entity type(Manager) is in partial participation. The Second Employee Entity type is in total participation.

- Bill is the entity with a composite key attribute(Order number)
- Stall maintenance and Dependants are two weak entity types.
- ExpenditureCalculation is a quaternary relationship type.

- There are two subclasses. One in role attribute of Customer and another in role attribute of Employee.
- Composite attributes – Menu, Stall timings in Stall entity type.
- Multi-valued attribute – Phone number in customer entity type.
- Derived attribute – Duration in Stall entity type.
- Manages is the relationship type with same participating entity type in distinct roles.
- WorksFor is a relationship with a cardinality constraints with max participation of Employee entity being 1.

FUNCTIONAL REQUIREMENTS

(a)

1) **Selection** : If a customer ID is selected , all his data will retrieved.

2) **Projection** : If the amount to be paid by a customer is more than Rs.500 then he has to pay his previous bills to make a new bill. Projection query is used to find the customers who has the amount attribute more than 500.

3) **Aggregate** : We use Sum and Difference Aggregate functions to update the Amount Attribute.

4) **Search** : The name attribute in the customer entity type uses this operation to search various customers.

(b)

Analysis Reports :

1) Total profit or loss can be calculated for each stall based on total payment received from the customers

and total salaries given to the employees with expenditure given per month.

2) Total money spent by a customer can be calculated from his weekly bills.

3) Average number of orders per day in stall can be calculated.

(c)

1) **LOADING OR INSERTION OF DATA**

i) When a person enrolled in the university wants to register his/her data in the food court, his/her data is inserted into the customer entity type.

ii) When a new dependent enters the university, his/her will be inserted into Dependent's Entity Type.

iii) A new stall can be added to the list of stalls if required.

iv) Respective integrity constraints will be checked while insertion.

2) **MODIFICATION AND UPDATION OF DATA**

i) Once a customer paid his weekly bills, the amount attribute is set to 0. If he pays only partial amount of the bill , then the amount attribute will be updated accordingly.

ii) If an employee gets a hike , his salary will be updated.

3) **DELETION OF DATA**

i) After a student graduates his/her data from the database will be deleted.

ii) If an employee resigns, his/her information can be removed.

THE END