Group Project-1

Introduction

The mini world describes an online store where people can review and purchase Items. Users can purchase these items via multiple methods like online wallets or online banking. The store saves previous transaction details such as address and payment method to save time for the user, Users can also track their orders after making a purchase.

Purpose

This database is used to store and manage various transactions of online store.

Users

- User to place an order
- Store management to manage various products and user's orders

Applications

- User : To store basic user details
- Address : To store the addresses of a user
- Cards Details : Store the card details of cards used by the user
- Product : Store details of the product
- Reviews : Store reviews of product by the customer
- Order : Store details regarding order

Database Requirements

Entities

1. User
    - Email
        - Char(128), Not NULL, Unique
    - Mobile Number
        - Char(128), Not NULL, Unique
    - Name
        - Composite of First name, Middle name, Last name, Not NULL
        - First name, Middle name, Last name = Char(64)
    - Premium Subscription Status
        - Boolean, Not NULL
2. Address
    - Lines
        - Composite of Address Line 1, Address Line 2, Not NULL
        - Address line 1 and Address Line 2 = Char(128)
    - City
        - Char(64), Not NULL
    - State

- - - Char(64), Not NULL
  - Zip Code
    - 100000 < Integer < 999999, Not NULL
3. Product
  - ID
    - Integer, Not NULL, Unique
  - Name
    - Char(128), Not NULL
  - Delivery availability
    - Multi valued (All cities where delivery is available), Not NULL
  - Category
    - Multi valued (All the categories product falls into), Not NULL
  - Price
    - Integer, Not NULL
  - Total Quantity
    - Integer, Not NULL
  - Rating
    - Average of all reviews
4. Review
  - Rating
    - Integer (1,5), Not NULL
  - Review
    - Char(1024)
5. Order
  - Order ID
    - Integer, Not NULL, Unique
  - Order status
    - One of [Billed, In transit, Delivered], Not NULL
  - Order Date
    - Date, Not NULL
  - Total Bill
    - Derived, Integer, Not NULL
6. Payment Method
  - Date Added
    - Date, Not NULL
  - is_default
    - Boolean, default=False, Not NULL
  - Subclasses:
    - Wallet
      - Provider (Char(32), Not NULL)
      - Auth Token (Char(128), Not NULL)
      - Balance (Integer)
    - Card Details
      - Name (Char(64), Not NULL)

- Card Number (Integer(16 digits), Not NULL)
- Expiry Date (Date, Not NULL)
- Name on card (Char(128), Not NULL)
- Billing address (Char(1024), Not NULL)

Weak Entities

`Address`, `Review`, `Payment Method` are weak entities in this database.

Relationships

1. Has addresses

   - `User` has addresses `Address`
   - Degree : 2
   - Cardinality ratio : 1:N
   - Cardinality constraint: Both `User` and `Address` have total participation

2. Can pay with

   - `User` can pay with `Payment Method`
   - Degree : 2
   - Cardinality ratio : 1:N
   - Cardinality constraint: Both `User` and `Payment Method` have total participation

3. Contains

   - `Order` contains `Product`
   - Degree : 2
   - Attributes : Quantity (Integer, Not NULL)
   - Cardinality ratio : 1:N
   - Cardinality constraint: `Order` has total participation whereas `Product` has partial participation

4. Reviewed

   - `User` rates `Product` with `Review`
   - Degree : 3
   - Cardinality ratio : 1:1:1
   - Cardinality constraint : `Review` has total participation whereas `User` and `Product` have partial participation. There is at most 1 `Review` for the pair (`User` , `Product`)

5. Placed

   - `User` placed an `Order` to an `Address` with `Payment Method`
   - Degree : 4
   - Cardinality ratio : 1:M:N:P
   - Cardinality Constraint : `Order` has total participation whereas `User`, `Address` and `Payment` have partial participation. There can be multiple `Order` per `User` having exactly one of `Address` and `Payment`

n > 3 Relationship

`Placed` is a quaternary relationship.

Functional Requirements

Modifications

1. Insert -

   - Insert User.
   - Insert Product to be sold.
   - User can insert address.
   - User can insert a payment method(entity type: payment details)
   - User can add a review.

2. Delete

   - A product can be deleted(if discontinued or out of stock).
   - User can delete a certain address.
   - User can delete a review.

3. Update

   - A product's details can be updated(price, total quantity).
   - User can change his/her review.
   - User can update address details.
   - User can update payment details.

Retrieval

1. Selection

   - Retrieve details of all the products available.
   - Retrieve details of all the users registered.

2. Projection

   - List of all orders above the value of certain amount.
   - List of all reviews for a particular product rated above a certain rating.

3. Aggregate

   - Product with maximum sales.
   - Product with highest reviews in a category.

4. Search

   - User can search for products that contain certain word.
   - User can search within their orders.

5. Analysis

   - Products in certain category with less than the average rating for that category.
   - Amount of orders placed for a particular product in a certain time period by suscribed users.