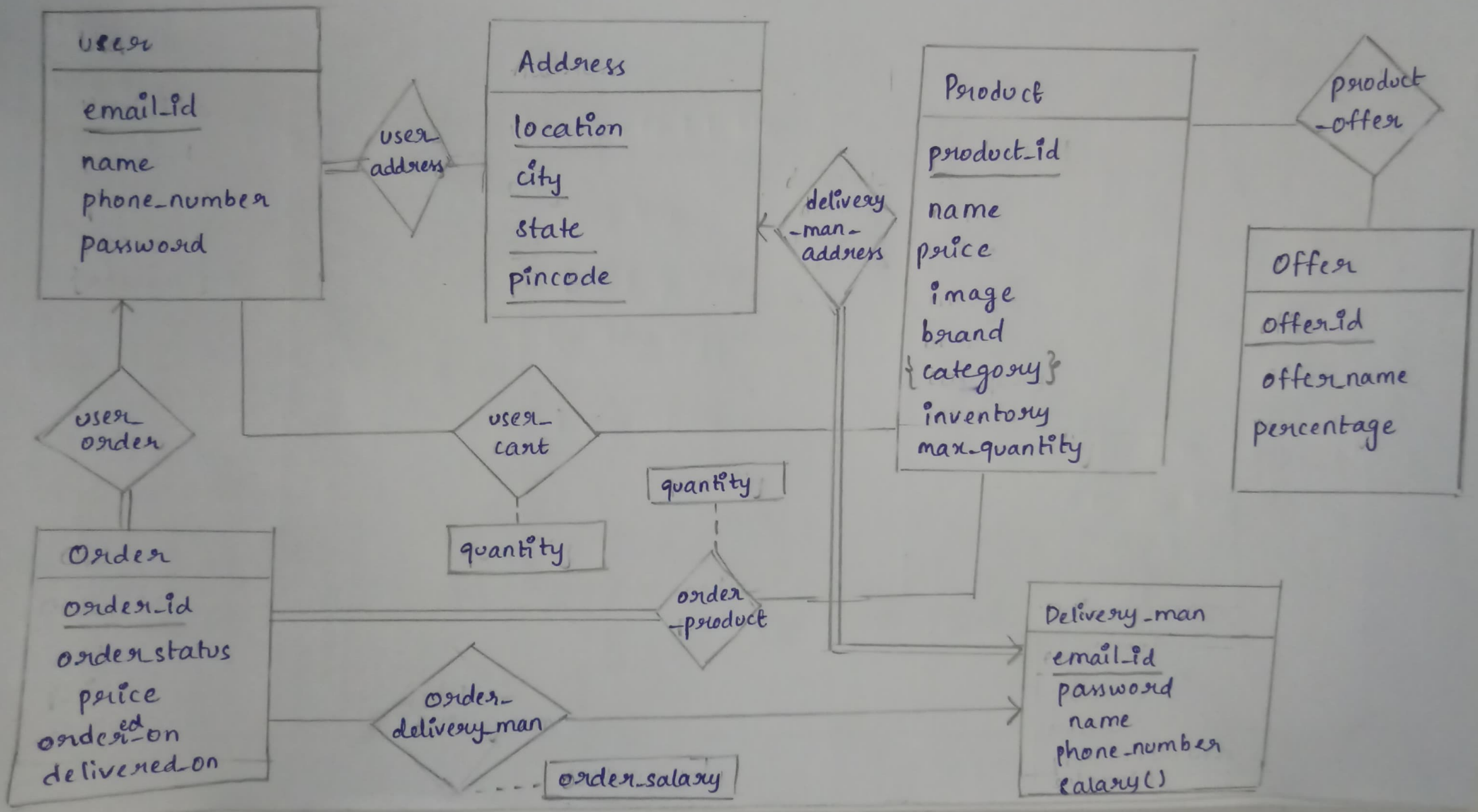
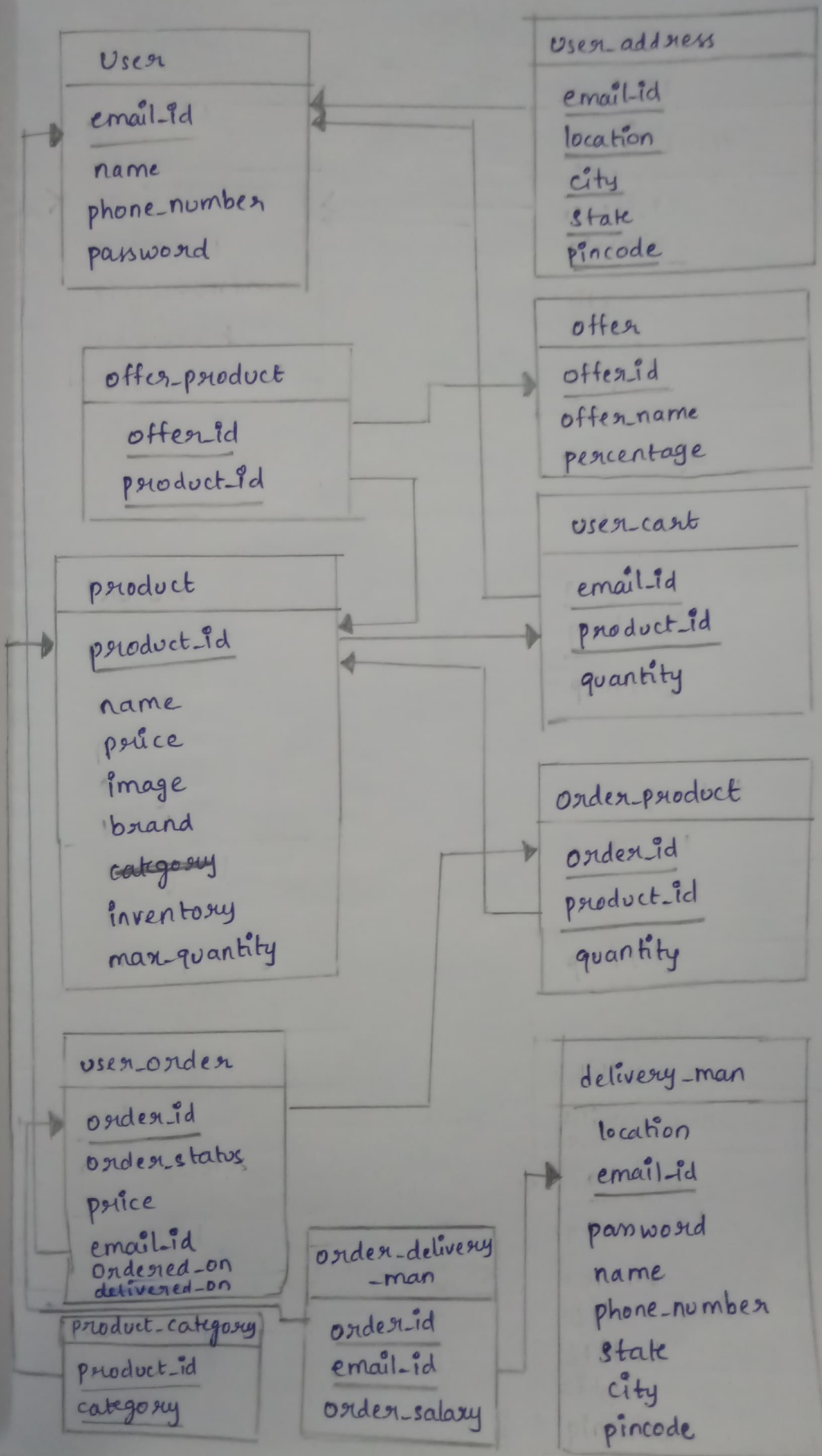
ONLINE GROCERY STORE E-R DIAGRAM (GROUP 22)



SCHEMA DIAGRAM (GROUP 22)



GROUP NUMBER : 22
GROUP MEMBERS :

1. Vaishnavi Munghate 18CS30045
2. Aayushi Vidyanta 18CS30003
3. Medidoddi Vahini 18CS10032

Primary keys and foreign keys are mentioned in the database schema

Data Types

1. User
    a. email_id - nvarchar(255), not null
    b. password - varchar(50), not null
       Constraint : check(length(password) >= 8)
    c. phone_number - varchar(10), null
       Constraint : check(length(phone_number) = 10 or phone_number is null)
    d. name - varchar(50), not null
2. User_address
    a. email_id - nvarchar(255), not null
    b. location - varchar(200), not null
    c. city - varchar(50), not null
    d. state - varchar(50), not null
    e. pincode - varchar(6), not null
       Constraint : check(length(pincode) = 6)
3. Product
    a. product_id - varchar(10), not null
       Constraint : check(length(product_id) = 10)
    b. name - varchar(50), not null
    c. price - numeric(8, 2), not null
       Constraint: check(price > 0)
    d. image - image, null
    e. brand - varchar(30), not null
    f. inventory - int, not null
       Constraint : check(inventory >= 0)
    g. max_quantity - int, not null
       Constraint : check(max_quantity > 0)
4. Product_category
    a. product_id - varchar(10), not null
       Constraint : check(length(product_id) = 10)
    b. category - varchar(30), not null
5. Offer
    a. offer_id - varchar(5), not null
       Constraint : check(length(offer_id) = 5)

b. offer_name - varchar(50), not null
c. percentage - decimal(5, 2), null
Constraint : check((0 <= percentage and percentage <= 100) or percentage is null)

6. Offer_product
   a. offer_id - varchar(5), not null
      Constraint : check(length(offer_id) = 5)
   b. product_id - varchar(10), not null
      Constraint : check(length(product_id) = 10)
7. User_cart
   a. email_id - nvarchar(255), not null
   b. product_id - varchar(10), not null
      Constraint : check(length(product_id) = 10)
   c. quantity - int, not null
      Constraint : check quantity >= min(p.inventory, p.max_quantity)
      from Product p
      where p.product_id = product_id
      (will be checked using triggers)
8. User_order
   a. order_id - varchar(20), not null
      Constraint : check(length(order_id) = 20)
   b. order_status - enum(“CONFIRMED”, “CANCELLED”, “DELIVERING”, “DELIVERED”), not null
   c. price - numeric(8, 2), not null
      Constraint : update price = select sum(p.price*op.quantity)
      from Product p, Order_product op
      where order_id = op.order_id and p.product_id = op.product_id;
      (will be automatically updated using triggers)
   d. email_id - nvarchar(255), not null
   e. ordered_on - TIMESTAMP using DEFAULT CURRENT_TIMESTAMP and ON UPDATE CURRENT_TIMESTAMP, not null
   f. delivered_on - TIMESTAMP, null
9. Order_product
   a. order_id - varchar(20), not null
      Constraint : check(length(order_id) = 20)
   b. product_id - varchar(10), not null
      Constraint : check(length(product_id) = 10)
   c. quantity - int, not null
      Constraint : check quantity >= min(p.inventory, p.max_quantity)
      from Product p
      where p.product_id = product_id
      (will be checked using triggers)

10. Delivery_man
    a. email_id - nvarchar(255), not null
    b. password - varchar(50), not null
       Constraint : check(length(password) >= 8)
    c. phone_number - varchar(10), not null
       Constraint : check(length(phone_number) = 10)
    d. name - varchar(50), not null
    e. location - varchar(200), not null
    f. city - varchar(50), not null
    g. state - varchar(50), not null
    h. pincode - varchar(6), not null
       Constraint : check(length(pincode) = 6)
11. Order_delivery_man
    a. order_id - varchar(20), not null
       Constraint : check(length(order_id) = 20)
    b. email_id - nvarchar(255), not null
    c. order_salary - numeric(8, 2), not null
       Constraint: check(order_salary > 0)

Assumptions

1. User can cancel only when its status is “CONFIRMED”, not later
2. User can pay for the order only using cash on delivery
3. Users can have multiple addresses while a delivery man can have only one address. Also, the relationship between User - Order and Delivery_man - order is different. Hence, two separate tables for them are required
4. Delivery_man’s monthly salary is calculated by adding the order_salary of all the orders delivered by him/her in that month

*****