Courier Management System

Task 1 - Database Design

Design a SQL schema for a Courier Management System with tables for Customers, Couriers, Orders, and Parcels. Define the relationships between these tables using appropriate foreign keys.

Requirements:

• Define the Database Schema • Create SQL tables for entities such as User, Courier, Employee, Location,Payment

• Define relationships between these tables (one-to-many, many-to-many, etc.).

• Populate Sample Data • Insert sample data into the tables to simulate real-world scenarios.

User Table:

User

(UserID INT PRIMARY KEY, Name VARCHAR(255),

Email VARCHAR(255) UNIQUE,

Password VARCHAR(255),

ContactNumber VARCHAR(20), Address TEXT);

Courier Table:

Courier

(CourierID INT PRIMARY KEY,

SenderName VARCHAR(255),

SenderAddress TEXT,

ReceiverName VARCHAR(255),

ReceiverAddress TEXT,

Weight DECIMAL(5, 2),

Status VARCHAR(50),

TrackingNumber VARCHAR(20) UNIQUE,

DeliverDate DATE);

CourierServices Table:

CourierServices

(ServiceID INT PRIMARY KEY,

ServiceName VARCHAR(100), Cost DECIMAL(8, 2));

Employee Table:

(EmployeeID INT PRIMARY KEY,

Name VARCHAR(255), Email VARCHAR(255) UNIQUE, ContactNumber VARCHAR(20), Role VARCHAR(50),

Salar DECIMAL(10, 2));

Location Table:

(LocationID INT PRIMARY KEY,

LocationName VARCHAR(100),

Address TEXT);

Payment Table:

(PaymentID INT PRIMARY KEY,

CourierID INT, LocationId INT,

Amount DECIMAL(10, 2), PaymentDate DATE, FOREIGN KEY (CourierID) REFERENCES Couriers(CourierID),

FOREIGN KEY (LocationID) REFERENCES Location(LocationID));

1. Introduction

The **Courier Management System** is designed to efficiently manage user orders, courier tracking, payments, and employee records. This documentation outlines the database schema, entity relationships, and data structure necessary for the system's operation.

2. Database Entities and Their Attributes

1. User

Stores sender and receiver details.

- **UserID** (PK) – Unique user identifier.
- **Name, Email, Password, ContactNumber, Address** – User details.

2. Courier

Manages courier shipment information.

- **CourierID** (PK) – Unique courier identifier.
- **SenderName, SenderAddress, ReceiverName, ReceiverAddress** – Shipment details.
- **Weight, Status, TrackingNumber (Unique), DeliveryDate** – Package specifics.

3. Courier Services

Defines available courier service types.

- **ServiceID** (PK) – Unique service identifier.
- **ServiceName, Cost** – Service details.

4. Employee

Stores employee information.

- **EmployeeID** (PK) – Unique employee identifier.
- **Name, Email, ContactNumber, Role, Salary** – Employee details.

5. Location

Represents courier hubs and service branches.

- **LocationID** (PK) – Unique location identifier.
- **LocationName, Address** – Location details.

6. Payment

Manages payments for courier services.

- **PaymentID** (PK) – Unique payment identifier.
- **CourierID (FK), LocationID (FK), Amount, PaymentDate** – Payment details.

3. Cardinality Relationships

Users ↔ Courier → 1:N

- One **User** can send multiple **Couriers**, but each **Courier** belongs to only one **User**.

Courier ↔ Payment → 1:1

- Each **Courier** has one **Payment**, and each **Payment** is linked to only one **Courier**.

Payment ↔ Location → N:1

- Multiple **Payments** are linked to a **single Location**.

Courier ↔ Employee → N:1

- Multiple **Couriers** can be assigned to a **single Employee**.

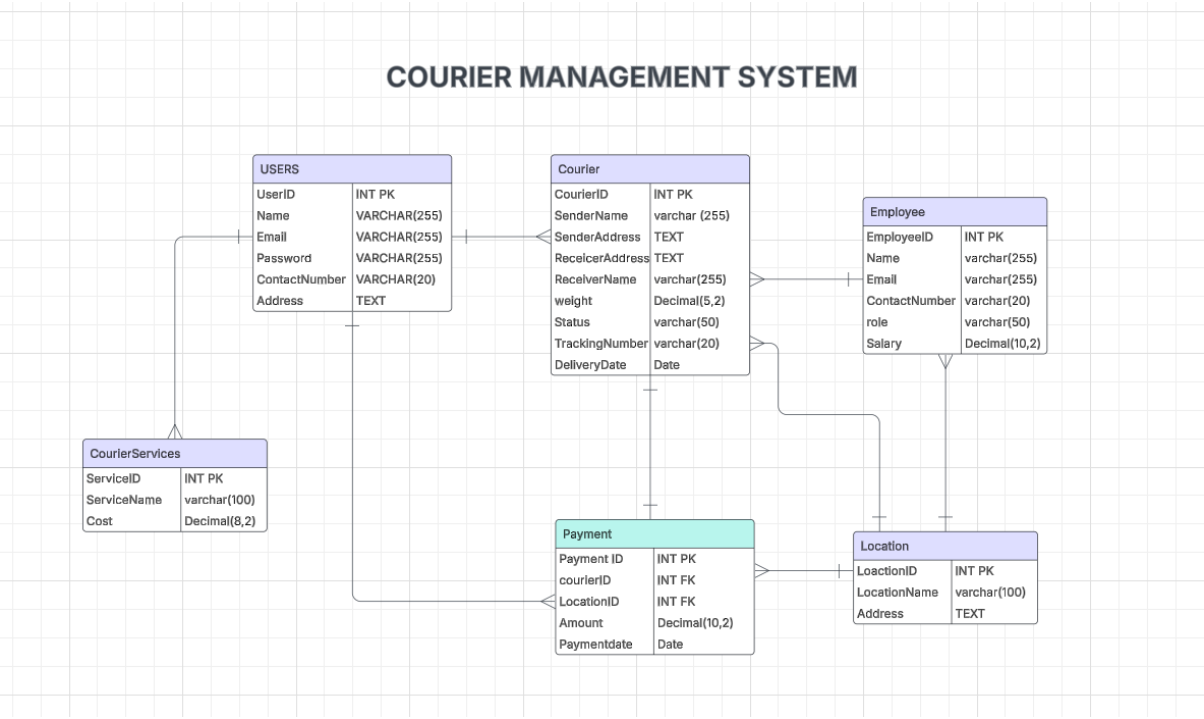
CourierServices ↔ Users → N:1

- Multiple **Users** can use a single **Courier Service**.

Employee ↔ Location → N:1

- Mutiple **Employee** can assigned for **single Location.**

4. ER Diagram



5. Steps

DATABASE CREATION

create database CMS;

User table

create table Users(UserID int primary key , Name varchar(255),Email varchar(255) unique, Password varchar(255),ContactNumber varchar(20),Address Text);

insert into Users values (1,'virat','virat@gmail.com','pass1111','9711223344','7 lion street,madurai');

insert into Users values (2,'dhoni','dhoni@gmail.com','pass2222','9766778899','17 king street,Trichy');

insert into Users values (3,'vijay','vijay@gmail.com','pass3333','9944556677','1 temple street,Trichy');

truncate table Users

select *from Users;

Results Messages

	UserID	Name	Email	Password	ContactNumber	Address
1	1	virat	virat@gmail.com	pass1111	9711223344	7 lion street,madurai
2	2	dhoni	dhoni@gmail.com	pass2222	9766778899	17 king street,Trichy
3	3	vijay	vijay@gmail.com	pass3333	9944556677	1 temple street,Trichy

Courier table

create table Courier (CourierID int primary key , SenderName varchar(255),SenderAddress text , ReceiverName varchar(255),ReceiverAddress text ,

Weight decimal(5,2),Status varchar(50),TrackingNumber varchar(20) unique, DeliveryDate date);

insert into Courier values (001, 'virat','7 lion street,madurai','Anu','12 mainroad,Chennai',2.5,'delivered','TRK001','2025-01-01');

insert into Courier values (002, 'dhoni','17 king street,Trichy','sakshi','2 vikasa road,vellore',3.5,'delivered','TRK002','2025-03-07');

insert into Courier values (003, 'vijay','1 temple street,Trichy','mithra','4 mainroad,cuddalore',2.5,'transported','TRK003','2025-01-06');

select *from Courier

Results Messages

	CourierID	SenderName	SenderAddress	ReceiverName	ReceiverAddress	Weight	Status	TrackingNumber	DeliveryDate
1	1	virat	7 lion street,madurai	Anu	12 mainroad,Chennai	2.50	delivered	TRK001	2025-01-01
2	2	dhoni	17 king street,Trichy	sakshi	2 vikasa road,vellore	3.50	delivered	TRK002	2025-03-07
3	3	vijay	1 temple street,Trichy	mithra	4 mainroad,cuddalore	2.50	transported	TRK003	2025-01-06

courier services

create table CourierServices (ServiceID int primary key , ServiceName varchar(100),Cost decimal(8,2));

insert into CourierServices values (1,'local',100.00);

insert into CourierServices values (2,'local',60.00);

insert into CourierServices values (3,'local',95.00);

select *from CourierServices;

Results Messages

	ServiceID	ServiceName	Cost
1	1	local	100.00
2	2	local	60.00
3	3	local	95.00

Employee table

create table EmployeeTable (EmployeeID int primary key , Name varchar(255),Email varchar(255) unique,ContactNumber varchar(20),Role varchar(50), Salary decimal(10,2));

insert into EmployeeTable values (111 , 'rajesh','rajesh@gmail.com','9133553377','delivery agent',3000.00);

insert into EmployeeTable values (112 , 'kumar','kumar@gmail.com','9136653377','support',2000.00);

insert into EmployeeTable values (113 , 'rahul','rahul@gmail.com','9133559787','delivery agent',3000.00);

select *from EmployeeTable;

Results Messages

	EmployeeID	Name	Email	ContactNumber	Role	Salary
1	111	rajesh	rajesh@gmail.com	9133553377	delivery agent	3000.00
2	112	kumar	kumar@gmail.com	9136653377	support	2000.00
3	113	rahul	rahul@gmail.com	9133559787	delivery agent	3000.00

Location table

create table Location(LocationID int primary key, LocationName varchar(100), Address Text);

insert into Location values(1,'Madurai','7 lion street,madurai');

insert into Location values(2,'Trichy','17 king street,Trichy');

insert into Location values(3,'vellore','2 vikasa road,vellore');

select *from Location ;

Results Messages

	LocationID	LocationName	Address
1	1	Madurai	7 lion street,madurai
2	2	Trichy	17 king street,Trichy
3	3	vellore	2 vikasa road,vellore

payment table

create table Payment(PaymentID int primary key , CourierID int ,LocationID int , Amount decimal(10,2),PaymentDate date, constraint FK_CID foreign key (CourierID) references Courier (CourierID)on delete cascade ,constraint FK_LID foreign key (LocationID) references Location(LocationID)on delete set null);

insert into Payment values(1, 001,1, 100.00, '2025-01-01');

insert into Payment values(2, 002,2,200.00, '2025-03-07');

select *from Payment;

Results Messages

	PaymentID	CourierID	LocationID	Amount	PaymentDate
1	1	1	1	100.00	2025-01-01
2	2	2	2	200.00	2025-03-07

Conclusion:

The **Courier Management System** database schema ensures efficient tracking, payment processing, and employee management through well-structured relationships. It maintains **data integrity, minimizes redundancy, and enhances operational efficiency** with clearly defined entity dependencies