



UNIVERSITY OF GONDAR
COLLEGE OF INFORMATICS
DEPARTMENT OF COMPUTER SCIENCE

DATABASE GROUP ASSIGNMENT

GROUP MEMBERS

1. Yohannes Tadiwos (GUR/01632/14)
2. Kalab Tesafye (GUR/02985/14)
3. Mariamawit Tesfaye (GUR/01905/14)
4. Mikias Wondafrash (GUR/02771/14)
5. Saron Tewodros (GUR/00882/14)

Submitted to: Mr. Misganaw Abeje
Submission date: 04/08/2015EC

Introduction

With the rise in e-commerce popularity, the need for efficient delivery services has increased rampantly. E-commerce platforms, local businesses, industries, individuals et cetera ... are all in need of fast, reliable, affordable and efficient delivery services to ensure the seamless delivery of their goods and documents. This increased demand requires innovative solutions. However, due to a lack of sufficient platform catering to this specific need delivery service has been quite fragmented and inaccessible. To bridge this gap we plan to launch the project *DeliveryXpress: A Comprehensive Database for Efficient Delivery Services*.

The main goal of this project is to establish a platform that will cater to this specific need of the sender and receiver while ensuring the seamless and reliable delivery of their package. The platform will be designed to be easy to use and user-friendly and will allow users to choose from a range of goods from varying providers. These providers will be thoroughly vetted and their performance records will be shared with users to ensure reliability and transparency.

DeliveryXpress will offer a wide range of support tools to aid with the tracking of packages and real-time updates, delivery confirmations, and many other features. It will be built to suit the needs of all business sizes, as well as individuals. It will help them to reduce costs, streamline operations, and ultimately optimize customer satisfaction.

To ensure timely and efficient delivery our database will also offer dynamic routing, dispatching and scheduling algorithms to optimize delivery routes and ensure packages get to their desired location

quickly and securely. We will also provide enhanced communication between customers, delivery personnel and vendors, improved inventory management and reliable tracking systems. Citing the potential for exponential growth in the e-commerce delivery sector, our project aims to become a trusted partner and market leader in facilitating and delivering these services.

In summary, the project *DeliveryXpress: A Comprehensive Database for Efficient Delivery Services* will provide users with efficient and reliable delivery services, an easy-to-use interface, transparent information sharing, and many other features to meet a variety of customer needs. Its ultimate goal is to provide customers with a hassle-free and convenient experience, which can help to build customer loyalty and increase revenue for the business. The platform is the perfect solution for businesses or individuals looking to simplify their logistics needs, optimize their processes and deliver goods seamlessly to their customers.

Objectives

General Objective:

- Designing a DataBase system for a Delivery Company called DeliveryXpress

Specific Objectives:

- Identifying the business process of the company
- Identifying the building blocks of the Database
- Drawing the ER diagram
- Mapping the ER diagram to relational table
- Normalizing the database
- Implementing the database system

Methodology

Data Collection Techniques

To offer fast reliable and seamless delivery of packages, the database needs to manage different orders, GPS tracking, scanning technologies(barcodes and QR codes), customer feedback and many more.

Some of the techniques to be used for data collection include:

- A. Order Form: The most straightforward method is to have customers fill out an order form that includes all the necessary details about the order, recipient, delivery address and other pertinent details.
- B. GPS Tracking: By installing GPS on delivery vehicles and personnel we could have real-time delivery status and track any deviations from usual routes.
- C. Scanner technologies: Using barcode scanners or RFID tags to track packages at various points of transit allows for a detailed history of where each package has been when. This makes it easy to identify the time and place where a defect happened in transit.
- D. Customer Feedback: Delivery companies can solicit feedback from customers on delivery success rates, time taken, efficiency and any delays, issues or inconveniences.
- E. Data Mining: Analyzing data from various sources to identify patterns, trends, and correlations related to delivery services such as traffic volumes, weather conditions, and delivery times.
- F. Observations: Observing and recording the behaviour of delivery drivers, customers and logistic managers to identify areas for improvement and optimization.

G. Focus groups: Bring together a group of people who use or are involved with delivery services to discuss and provide feedback on the service.

Overall, using a combination of the above methods, parsing through and analyzing data can provide useful insights into operations to improve service and customer satisfaction. We may also employ other data collection techniques to ensure the collection of comprehensive and accurate data to make its data valuable for efficient delivery services.

Data Modeling Approach

Our project provides delivery services on a large scale with a focus on efficiency. This implies that we have to process a large amount of data, to handle operations such as logistic services and more. For this reason, we choose to use a Relational Database Modeling approach, because relational databases are highly structured and have the potential to handle large amounts of data efficiently. Additionally using a relational database would allow for the use of SQL, which has powerful tools for data analysis and can help optimize the delivery process.

Thus, to model a relational database for DeliveryXpress's database, we would first need to identify the entities and their attributes that would store information related to the services it delivers.

For example, we can start with the following entities:

- Customers: those who place orders
- Orders: which contain information about products ordered, pickup and delivery destination, expected delivery date, et cetera.

- Drivers: who will deliver orders.
- Vehicles: these are used by the drivers for transportation
- Warehouses: where inventory is stored.

Each of these entities has its own set of attributes that would need to be included in the database. For example:

- The customer's entity would have
 - ◆ Customer Name,
 - ◆ Email,
 - ◆ Phone Number,
 - ◆ And Delivery Address.
- Similarly, the orders entity might include attributes such as:
 - ◆ Order ID,
 - ◆ Customer ID,
 - ◆ Pickup Address,
 - ◆ Delivery Address,
 - ◆ Delivery Date,
 - ◆ Expected Delivery Time, et cetera.

After identifying entities and their attributes, we can start to establish relationships between them. Our goal here is to ensure that data is organized logically such that it can be easily accessed and updated. For example, one relationship between entities would be an order belonging to a particular customer. This would be represented by including the customer's ID attributes in the orders entity.

In terms of the physical implementation of the database, we would define tables for each of the entities and columns for their attributes. Each table would have a candidate key to identify each object in the database and we need to create a foreign key to establish relationships between entities.

(Note: An exhaustive list of all entities, attributes and relations between attributes is shown below.)

Overall a well-designed relational database would provide our project with efficient access to its delivery-related data and enable us to make more informed business decisions.

Hardware and Software Requirements

As the scale of our project grows, the software and hardware requirements may vary. However, here is a list of the general requirements that we may need:

→ Software Requirements:

- ◆ Database management System: such as Microsoft SQL server
- ◆ Programming language: such as Java or Python
- ◆ Web Application Framework: such as Django
- ◆ Web Servers: such as Apache or Nginx
- ◆ API Development: such as Postman
- ◆ Mapping and Geolocation Services: such as Google Maps
- ◆ Security Software: such as firewalls. Antivirus, and encryption tools.

→ Hardware Requirements:

- ◆ Server: A dedicated machine or hosting platform that can support the database and web server requirements, such as AWS EC2, Google Computing Engine, or Microsoft Azure.

- ◆ **Storage:** Sufficient storage capacity to house the database, application code and static assets such as images, videos, et cetera.
- ◆ **Memory:** Adequate system memory to ensure efficient processing and handling of requests.
- ◆ **Network:** High-speed internet connectivity with low latency to support real-time delivery tracking and status updates.

The actual requirements may depend on the number of concurrent users, the amount of data being processed and stored, the expected load on the system, and other factors that may arise during the development and implementation phase of the project.

Database Designing

Business Rules

As stated above, our project aims to provide efficient delivery service. Therefore, our business rules reflect this objective and guide the development of our database.

Here are our business rules for project DeliveryXpress:

1. Customers must provide accurate and complete shipping information. These include name, address, phone number, email address et cetera.
2. The system should be able to calculate shipping costs based on the distance, weight and any special requirements such as expected early delivery or fragile items.
3. The system must ensure that all orders are assigned to the most appropriate delivery vehicle based on factors such as distance, weight, and delivery time.
4. The system must provide customers with accurate and timely information related to their orders, including tracking information, estimated delivery times and any delays that may occur during transit. It should be able to send delivery status to customers via email or text message.
5. All packages must be scanned and logged into the system upon receipt, including package source, destination, weight, and details of the contents.
6. All delivery personnel must adhere to driving regulations and safe driving practices to protect themselves, other drivers on the road and the items being transported.

7. The system should be capable of generating reports and analytics that can be used to identify areas for improvement and optimize delivery processes.
8. The system should be able to handle multiple types of products, including perishable and hazardous materials and enforce compliance with relevant shipping regulations.
9. Delivery routes must be optimized based on the most efficient delivery routes, taking into account traffic conditions and other factors that may impact delivery times.
10. Delivery schedules must be coordinated with delivery routes to ensure timely deliveries.
11. There should be appropriate security measures put in place to safeguard customer data and prevent unauthorised access to the system.
12. All transactions associated with shipping and payments must be accurately recorded and reconciled to financial records to incur transparency and accountability.
13. In case of any discrepancies or issues with the shipment, clear escalation paths must be defined to minimize delays in resolving issues.
14. Delivery confirmations must be obtained from recipients upon delivery to reduce the likelihood of fraudulent claims or disputes.
15. In the event of a delivery problem or complaint from a customer, the system must have a customer service module that allows customers to submit feedback and resolve any issues in a timely and satisfactory manner.
16. The system must generate automatic alerts to the delivery personnel and customers in case of any unforeseen delay or problem caused by traffic or weather conditions.

These business rules will help shape the design and functionality of our database. They have a significant influence on our data model and the data model captures relevant aspects of the business processes involved in delivery, including the procedures for taking orders, processing payments, tracking packages and delivering goods. They also dictate the specific requirements our system must meet to deliver an efficient delivery service.

Overall, the business rules of the DeliveryXpress project play a crucial role in shaping the functionality, design and usability of our database, ensuring that it meets the needs for a high-quality and efficient delivery service to its customers.

Building Blocks of the Database

Entities:

1. Customer: A person or organization that requests the delivery service.
2. Order: A specific delivery request made by a customer.
3. Delivery Personnel: The person assigned to deliver the order.
4. Delivery Vehicle: The vehicle (eg., truck, car, bicycle) used to transport the order.
5. Delivery Status: The status of the order. (eg., in transit, delivered)
6. Pickup Location(s): The location(s) where the order is picked up. It could be one of DeliveryXpress's warehouses or a vendor's warehouse.
7. Destination(s): The location(s) where the order is delivered.
8. Payment: The amount paid by the customer for the delivery service or the product.
9. Billing: The record of payment transactions and invoices.

10. Customer Support: The department responsible for handling customer inquiries, complaints and feedback. Manages mode of communication, customer preference, complaint status et cetera.
11. Delivery History: The recording of past deliveries and associated details such as time, date, address, delivery status, et cetera.
12. Vendor: The company that provides the products to be delivered, including its employees, operations, and logistics.
13. Service Type: The type of delivery service offered. (e.g., same-day, overnight, scheduled)
14. Inventory: Stock of products available at the warehouse.
15. Warehouse: Storage place where orders are packed and dispatched
16. Shipment: Process of packing and dispatching orders from the warehouse
17. Communication Channels: Platform used to communicate between customers, delivery persons, and customer support.
18. Packaging Details: Information like dimension weight, and type of packaging of the order.
19. Payment Provider: Third-party payment service providers.
20. Special Requirements: Any specific requirements as requested by the customers like fragile items or temperature control.

Attributes of the above entities:

1. Customer:
 - a. Customer ID (Unique Identifier)
 - b. Name (First name, Last name)
 - c. Address (Street, City, State, Zip Code)
 - d. Phone Number
 - e. Email Address

- f. Payment Information (E~Birr, Cash, Wire Transfer, et cetera.)
 - g. Purchase History (Date, Time, Items Purchased)
 - h. Delivery Address (If different from billing address, or home address)
 - i. Preferred Delivery Time
 - j. Delivery Instructions (if any)
 - k. Loyalty Program Membership
 - l. Account Creation Date
 - m. Account status (Active, Inactive, Suspended, etc.)
 - n. Password (encrypted)
 - o. Profile Picture
2. Order:
- a. Order ID (Unique Identifier)
 - b. Customer ID
 - c. Address (can be the customer's address or a different location)
 - d. Pickup Address (mostly warehouse)
 - e. Delivery Date
 - f. Pickup Date
 - g. Order Status (e.g., in transit, delivered, canceled)
 - h. Payment Method (the payment method used to pay for this specific order, e.g., cash on delivery, or bank transfer)
 - i. Payment Status (e.g., paid, pending, failed)
 - j. Total cost (including shipment and any taxes and fees)
 - k. Quantity of Items, Name of Items, And Description of Items
 - l. Special Instructions or Delivery Notes are set by the customer.
 - m. Estimated Delivery Time
 - n. Delivery Personnel.
3. Delivery Personnel:

- a. ID (Unique Identifier)
 - b. Name
 - c. Date of Birth
 - d. Address
 - e. Email
 - f. Phone Number
 - g. Nationality
 - h. Identification Number (Government Issued, e.g., passport, driver's license)
 - i. Date of hire
 - j. Employee status (full-time, part-time, contractor)
 - k. Delivery vehicle type (e.g., truck, car, bike)
 - l. Vehicle license plate details
 - m. Vehicle insurance details
 - n. Deriving license details
 - o. Uniform size
 - p. Shift preferences
 - q. Work experience
 - r. Salary
 - s. Bank account details
 - t. Emergency Contact information
 - u. Availability Schedule
 - v. Performance metrics (e.g., deliveries completed per day, customer satisfaction rating)
 - w. Training records
 - x. Performance review/evaluation results
 - y. Benefits (e.g., healthcare, retirement plan)
4. Delivery Vehicle:
- a. Vehicle ID (Unique identifier)
 - b. Make (manufacturer of the vehicle)
 - c. Model (specific model of the vehicle)
 - d. Year (the year the vehicle was manufactured)

- e. Type (the classification of the vehicle such as length and height)
 - f. Size (physical dimensions of the vehicle)
 - g. Capacity (amount it can carry)
 - h. Fuel type (gasoline, or diesel)
 - i. MPG (miles per gallon, indicating the vehicle's fuel efficiency)
 - j. License Plate Number
 - k. Registration Expiration Date (the date the vehicle's registration expires)
 - l. Insurance Provider (the company providing insurance coverage for the vehicle)
 - m. Insurance Expiration Date
 - n. Maintenance Schedule (frequency, last service, expected next service)
 - o. Current Location and Status (for tracking and delivery estimation)
5. Delivery Status:
- a. Order ID
 - b. Status (Pending, in transit, delivered, attempted delivery, delayed, et cetera)
 - c. Delivery Date
 - d. Estimated Delivery time
 - e. Delivery Address
 - f. Delivery Attempts
 - g. Delivery Confirmation
 - h. The delivery method
 - i. Tracking number (a unique number assigned to order shipment that enables tracking of the delivery progress)
 - j. Delivery Personnel
6. Pickup Location(s):
- a. Location ID

- b. Address
 - c. City
 - d. State/Province
 - e. Country
 - f. Postal/Zip Code
 - g. Contact Name
 - h. Contact Email
 - i. Latitude
 - j. Longitude
 - k. Pickup Hours
 - l. Pickup Notification Required (indicates whether an advance notification is required for pickups at this location)
 - m. Pickup instructions
 - n. Pickup Capacity (maximum number of pickups that should be handled at this location)
 - o. Pickup restrictions (any restrictions that should be followed when picking up from this location)
 - p. Pickup fees (if any)
 - q. Pickup Confirmation Required (Indicates whether a confirmation is required from the pickup location before the pickup can be scheduled)
7. Destination(s):
- a. Destination ID
 - b. Street Address
 - c. City
 - d. State/Province/Region
 - e. Postal/Zip code
 - f. Country
 - g. Contact name (the person who will be contacted to receive the item)
 - h. Contact Phone Number

- i. Contact Email
 - j. Delivery Instructions
 - k. Delivery Notes
 - l. Delivery Status
 - m. Delivery Date/Time
 - n. Package Information
8. Payment:
- a. Payment ID
 - b. Customer ID
 - c. Amount
 - d. Payment Type (cash, credit, or wire transfer)
 - e. Transaction Date
 - f. Payment Status (pending, completed, failed, et cetera)
 - g. Transaction ID
 - h. Payment Method (Internet Banking, Mobile Banking, cash on delivery or others)
 - i. Payment Gateway Name (if payment was made through an electronic payment gateway option)
9. Billing:
- a. Bill ID (Unique identifier for each bill)
 - b. Customer ID
 - c. Invoice Date
 - d. Due Date
 - e. Payment Method
 - f. Payment Status
 - g. Amount Due
 - h. Items Purchased
 - i. Tax
 - j. Shipping charges
 - k. Discounts
 - l. Previous Balance (any outstanding balance from previous transactions)

- m. Late Fees (if the customer fails to pay by the due date)
 - n. Payment Confirmation
 - o. Notes (about the delivery transaction, if any)
10. Customer Support:
- a. Customer ID
 - b. Customer Name
 - c. Customer Contact Number
 - d. Email Address
 - e. Mailing Address
 - f. Orders History
 - g. Order Status
 - h. Payment Information
 - i. Delivery Address
 - j. Delivery Status
 - k. Delivery Date and time
 - l. Complaint Description
 - m. Complaint Date and time
 - n. Complaint Status
 - o. Resolution Description
 - p. Resolution Date and time
 - q. Feedback on Service
 - r. Frequency of Service Requests
 - s. Mode of Communication Preference
 - t. Preferred delivery timing
11. Delivery History:
- a. Delivery ID
 - b. Sender Name
 - c. Sener Address
 - d. Sender Contact Number
 - e. Recipient Name
 - f. Recipient Address
 - g. Recipient Contact Number

- h. Package Contents
- i. Delivery Date
- j. Delivery Time
- k. Delivery Status
- l. Delivery Method
- m. Delivery Cost
- n. Delivery Personnel
- o. Tracking Information
- p. Proof of Delivery
- q. Delivery Notes
- r. Delivery feedback from customers
- s. Delivery delay details
- t. Reason for delivery failures/returns

12. Vendor:

- a. Vendor ID
- b. Vendor Name
- c. Vendor Email
- d. Vendor Phone
- e. Vendor Address
- f. Account Creation Date
- g. Business type (e.g., physical shop, online merchant, an individual et cetera.)
- h. Operation hours
- i. Bank Account Details
- j. Payment method
- k. Delivery Zone
- l. Service type (type of delivery service their product or service needs, e.g., same-day, next-day, overnight, et cetera)
- m. Brand rating (rating information about vendor history)
- n. Product Category

- o. Average delivery time (the average time it takes for the vendor to prepare and hand over a delivery package (in minutes/hours/days))

13. Service Type:

- a. Service Type ID
- b. Service Name
- c. Service Description
- d. Service Cost
- e. Delivery Area
- f. Delivery Time
- g. Delivery Method
- h. Delivery Vehicle Capacity
- i. Maximum Weight Capacity
- j. Permitted Locations
- k. Delivery Tracking
- l. Special Requirements

14. Inventory:

- a. Product ID
- b. Product Name
- c. Manufacturer Name
- d. SKU (stock-keeping Unit)
- e. UPC (Universal Product Code)
- f. Quantity in Stock
- g. Minimum Inventory Level
- h. Maximum Inventory Level
- i. Reorder Point
- j. Location of the product in the warehouse
- k. Cost per unit
- l. Selling price per unit
- m. Date of last inventory check
- n. Date of last product delivery
- o. Date of next planned delivery

- p. Supplier ID
- q. Supplier Name
- 15. Warehouse:
 - a. Warehouse ID
 - b. Warehouse Name
 - c. Address
 - d. Contact Person
 - e. Phone Number
 - f. Email Address
 - g. Maximum Capacity (in terms of Storage)
 - h. Current Stock Level
 - i. Date of Last Inventory Check
 - j. Security Level (e.g., high, medium, low)
 - k. Temperature Control Available (Yes/No)
 - l. Humidity Control Available (Yes/No)
 - m. Regular Cleaning Schedules (frequency in days)
 - n. Pest Control Schedule (frequency in days)
 - o. Warehouse Open hours
- 16. Shipment:
 - a. Shipment ID (Unique Identifier for each shipment)
 - b. Sender Name
 - c. Sender Address
 - d. Receiver Address
 - e. Shipment Date
 - f. Expected Delivery Date (anticipated delivery date)
 - g. Actual Delivery Date (Date the shipment was delivered)
 - h. Shipment Status
 - i. Size (dimensions of the shipment package)
 - j. Shipping Method
 - k. Shipping Cost
 - l. Tracking Number
 - m. Sender Email

- n. Receiver Email
- o. Additional Charges (associated with the shipment)
- 17. Communication Channels:
 - a. Channel ID
 - b. Channel Name (e.g., Email, Phone, SMS).
 - c. Channel Type (e.g., Email, Phone, text)
 - d. Channel Status (active, inactive, completed)
 - e. Channel Address (contact information associated with the channel ID)
 - f. Customer ID
 - g. Delivery Personnel ID
 - h. Customer Support ID
 - i. Last Updated
 - j. Created At
- 18. Packaging Details:
 - a. Package ID
 - b. Sender Name
 - c. Sender Address
 - d. Sender Email
 - e. Sender Phone Number
 - f. Receiver Name
 - g. Receiver Address
 - h. Receiver Phone Number
 - i. Delivery Date
 - j. Delivery Time
 - k. Delivery Status
 - l. Estimated Time of Arrival
 - m. Weight
 - n. Dimensions (Height, Width, Length)
 - o. Mode of Transport
 - p. Shipping Method
 - q. Handling Instructions

- r. Insurance Details
- s. Customs Information (e.g., customs value, Description of contents, duty and taxes)
- t. Insurance Details (if applicable)
- u. Special Requirements

19. Payment Provider:

- a. Payment Provider ID
- b. Payment Provider Name
- c. Payment Provider Website
- d. Payment Method
- e. Payment Gateway ID
- f. API key
- g. Secret Key
- h. Payment Processing Fee
- i. Currencies Accepted
- j. Payment settlement Period
- k. Payment Confirmation Process
- l. Integration Guide and Documentation
- m. Security and Fraud Prevention Measures
- n. Customer Service Contact Information
- o. Payment Reports and Analytics

20. Special Requirements:

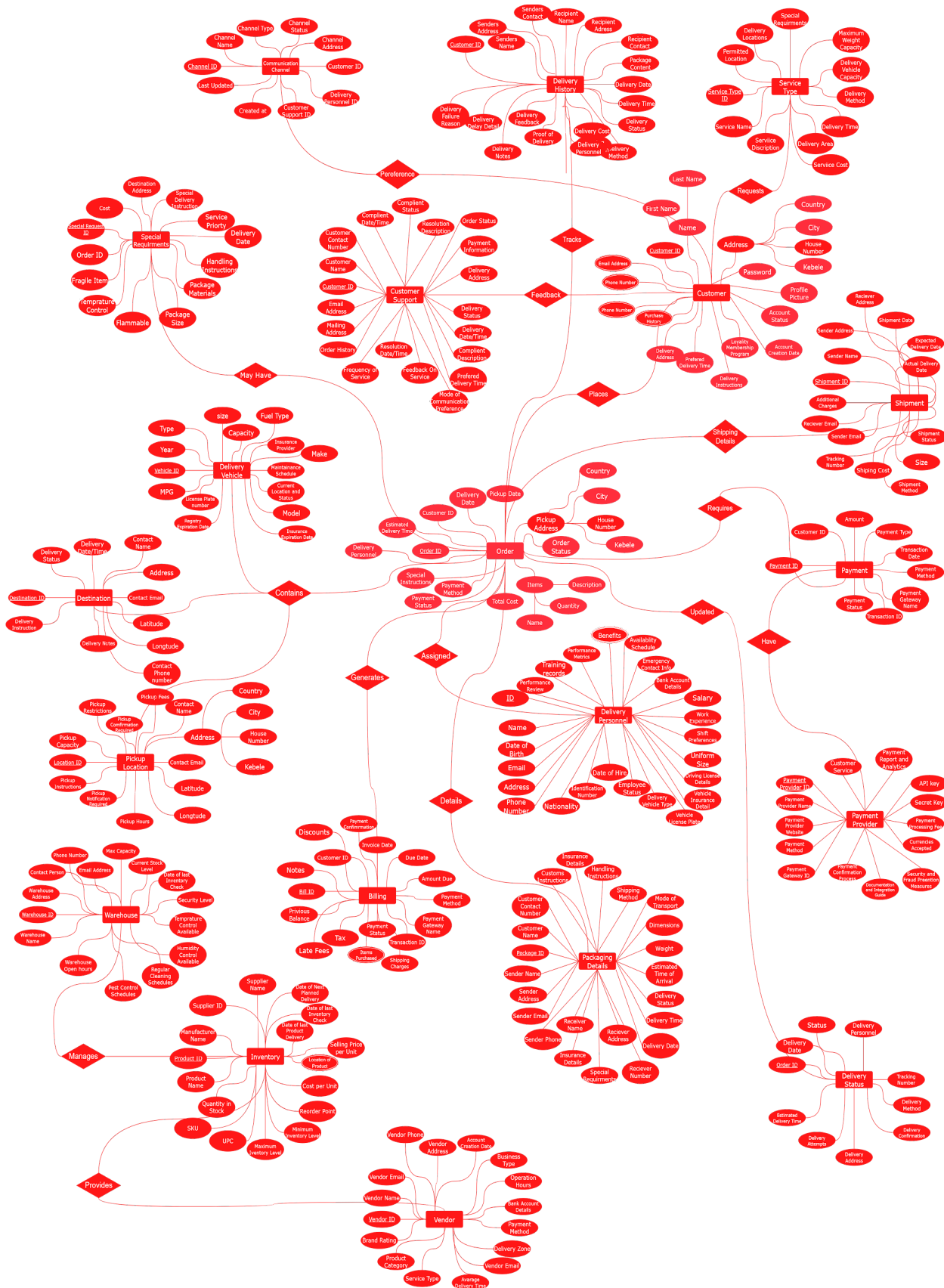
- a. Special Request ID (an identifier for the special request in the database)
- b. Order ID (to identify to which order a special requirement belongs)
- c. Fragile Item (boolean indicating if the item is fragile)
- d. Temperature Control (boolean if an item requires temperature control)
- e. Flammable (boolean, if the item is flammable)
- f. Package Size
- g. Package Materials

- h. Handling Instructions
- i. Delivery Date
- j. Destination address
- k. Special Delivery Instructions
- l. Service Priority
- m. Cost

Relationship between entities:

1. A customer places zero to many Orders, and each order is placed by one customer.
2. Each order has one or multiple Delivery Personnel assigned, and each Delivery Personnel can be assigned to one or several orders.
3. An order has one Delivery Vehicle assigned, and each Delivery vehicle can be assigned to one or many orders.
4. The Delivery Status of an order can be updated over time, and each order can have one current Delivery Status at any given time.
5. An Order can contain one or more pickup locations and each pickup location can be tied to one or many orders.
6. An Order can have one or multiple Destinations, and each Destination can be for one or more Orders.
7. Each Order requires a Payment, and every Payment is associated with one Order.
8. An Order generates one or multiple Billings, and every Billing has one Order associated with it.
9. Customers may contact Customer support for Assistance, and each customer can perform zero to many Customer Support interactions over time

10. Delivery History tracks the path of an order from pickup to destination, and each Order generates one Delivery History entry.
11. Each vendor associated with the company has one or many Service Types they offer, and one or many vendors offer each Service Type.
12. Products are tracked in Inventory, whereby inventory has one or multiple items, and each item can be tracked in one or many Inventories.
13. The Warehouses manage inventory levels, and each Warehouse manages one or several Inventories.
14. Shipment refers to the shipping details of an order, and each shipment is received by one Order
15. Communications Channels denote the ways of communication utilised by customers such as social media, phone or email, and each Order is assigned one or several Communications Channels.
16. Packaging Details signify the packaging of an order, and every Order correlates with one packaging Detail.
17. Payment Provider details are entered for each Payment, and each payment can have one Payment Provider associated with it.
18. Special Requirements denote specific customer requests related to their order, and each Order may have zero to many Special Requirements.
19. Special Requirements denote specific customer requests related to their order, and each Order may have zero to many Special Requirements.



Customer	
CustomerID	INT
FirstName	VARCHAR(50)
LastName	VARCHAR(50)
Address	VARCHAR(100)
City	VARCHAR(50)
State	VARCHAR(50)
ZipCode	VARCHAR(10)
PhoneNumber	VARCHAR(20)
EmailAddress	VARCHAR(50)
PaymentInformation	VARCHAR(50)
PurchaseHistory	VARCHAR(100)
DeliveryAddress	VARCHAR(100)
PreferredDeliveryTime	VARCHAR(50)
DeliveryInstructions	VARCHAR(100)
LoyaltyProgramMembership	VARCHAR(50)
AccountCreationDate	DATE
AccountStatus	VARCHAR(20)
Password	VARCHAR(100)
ProfilePicture	BLOB

Delivery Status	
OrderID	INT
Status	VARCHAR(20)
DeliveryDate	DATE
EstimatedDeliveryTime	VARCHAR(50)
DeliveryAddress	VARCHAR(100)
DeliveryAttempts	INT
DeliveryConfirmation	VARCHAR(50)
DeliveryMethod	VARCHAR(50)
TrackingNumber	VARCHAR(50)
DeliveryPersonnel	VARCHAR(50)

Billing	
BillID	INT
CustomerID	INT
InvoiceDate	DATE
DueDate	DATE
PaymentMethod	VARCHAR(50)
PaymentStatus	VARCHAR(20)
AmountDue	DECIMAL(10,2)
Purchased	VARCHAR(100)
Tax	DECIMAL(10,2)
ShippingCharges	DECIMAL(10,2)
Discounts	DECIMAL(10,2)
PreviousBalance	DECIMAL(10,2)
LateFees	DECIMAL(10,2)
PaymentConfirmation	VARCHAR(50)
Notes	VARCHAR(100)

Wishlist	
WishlistID	INT
CustomerID	INT
ProductID	INT
AddedDateAndTime	DATETIME

Product Category Mapping	
ProductID	INT
CategoryID	INT

Order	
OrderID	INT
CustomerID	INT
Address	VARCHAR(100)
PickupAddress	VARCHAR(100)
DeliveryDate	DATE
PickupDate	DATE
OrderStatus	VARCHAR(20)
PaymentMethod	VARCHAR(50)
PaymentStatus	VARCHAR(20)
TotalCost	DECIMAL(10,2)
QuantityOfItems	INT
NameOfItems	VARCHAR(100)
DescriptionOfItems	VARCHAR(200)
SpecialInstructions	VARCHAR(100)
EstimatedDeliveryTime	VARCHAR(50)
DeliveryPersonnel	VARCHAR(50)

Pickup Location	
LocationID	INT
Address	VARCHAR(100)
City	VARCHAR(50)
StateOrProvince	VARCHAR(50)
Country	VARCHAR(50)
PostalOrZipCode	VARCHAR(10)
ContactName	VARCHAR(50)
ContactEmail	VARCHAR(50)
Latitude	DECIMAL(8,6)
Longitude	DECIMAL(9,6)
PickupHours	VARCHAR(50)
PickupNotificationRequired	BOOLEAN
PickupInstructions	VARCHAR(100)
PickupCapacity	INT
PickupRestrictions	VARCHAR(100)
PickupFees	DECIMAL(10,2)
PickupConfirmationRequired	BOOLEAN

Customer Support	
CustomerID	INT
CustomerName	VARCHAR(50)
CustomerContactNumber	VARCHAR(20)
EmailAddress	VARCHAR(50)
MailingAddress	VARCHAR(100)
OrdersHistory	VARCHAR(100)
OrderStatus	VARCHAR(20)
PaymentInformation	VARCHAR(50)
DeliveryAddress	VARCHAR(100)
DeliveryStatus	VARCHAR(20)
DeliveryDateAndTime	DATETIME
ComplaintDescription	VARCHAR(100)
ComplaintDateAndTime	DATETIME
ComplaintStatus	VARCHAR(20)
ResolutionDescription	VARCHAR(100)
ResolutionDateAndTime	DATETIME
FeedbackOnService	VARCHAR(100)
FrequencyOfServiceRequests	VARCHAR(50)
ModeOfCommunicationPreference	VARCHAR(50)
PreferredDeliveryTiming	VARCHAR(50)

Coupons	
CouponID	INT
CouponCode	VARCHAR(50)
DiscountPercentage	DECIMAL(5,2)
ExpirationDate	DATE

Product Reviews	
ReviewID	INT
ProductID	INT
CustomerID	INT
ReviewTitle	VARCHAR(100)
ReviewDescription	VARCHAR(500)
Rating	INT
ReviewDateAndTime	DATETIME

Delivery Personnel	
ID	INT
Name	VARCHAR(50)
DateOfBirth	DATE
Address	VARCHAR(100)
Email	VARCHAR(50)
PhoneNumber	VARCHAR(20)
Nationality	VARCHAR(50)
IdentificationNumber	VARCHAR(50)
DateOfHire	DATE
EmployeeStatus	VARCHAR(20)
DeliveryVehicleType	VARCHAR(50)
VehicleLicensePlateDetails	VARCHAR(50)
VehicleInsuranceDetails	VARCHAR(100)
DrivingLicenseDetails	VARCHAR(100)
UniformSize	VARCHAR(20)
ShiftPreferences	VARCHAR(50)
WorkExperience	VARCHAR(100)
Salary	DECIMAL(10,2)
BankAccountDetails	VARCHAR(100)
EmergencyContactInformation	VARCHAR(100)
AvailabilitySchedule	VARCHAR(100)
PerformanceMetrics	VARCHAR(100)
TrainingRecords	VARCHAR(100)
PerformanceReviewEvaluationResults	VARCHAR(100)
Benefits	VARCHAR(100)

Destination	
DestinationID	INT
StreetAddress	VARCHAR(100)
City	VARCHAR(50)
StateOrProvinceOrRegion	VARCHAR(50)
PostalOrZipCode	VARCHAR(10)
Country	VARCHAR(50)
ContactName	VARCHAR(50)
ContactPhoneNumber	VARCHAR(20)
ContactEmail	VARCHAR(50)
DeliveryInstructions	VARCHAR(100)
DeliveryNotes	VARCHAR(100)
DeliveryStatus	VARCHAR(20)
DeliveryDateAndTime	DATETIME
PackageInformation	VARCHAR(100)

Delivery History	
DeliveryID	INT
SenderName	VARCHAR(50)
SenderAddress	VARCHAR(100)
SenderContactNumber	VARCHAR(20)
RecipientName	VARCHAR(50)
RecipientAddress	VARCHAR(100)
RecipientContactNumber	VARCHAR(20)
PackageContents	VARCHAR(100)
PackageWeight	DECIMAL(10,2)
DeliveryDateAndTime	DATETIME
DeliveryStatus	VARCHAR(20)
DeliveryPersonName	VARCHAR(50)
DeliveryPersonContactNumber	VARCHAR(20)

Refunds And Returns	
ReturnID	INT
CustomerID	INT
ProductID	INT
OrderID	INT
ReturnReason	VARCHAR(100)
ReturnDateAndTime	DATETIME
RefundAmount	DECIMAL(10,2)
RefundMode	VARCHAR(50)
RefundStatus	VARCHAR(20)
RefundDateAndTime	DATETIME

Product Specifications	
ProductID	INT
SpecificationName	VARCHAR(50)
SpecificationValue	VARCHAR(50)

Delivery Vehicle	
VehicleID	INT
Make	VARCHAR(50)
Model	VARCHAR(50)
Year	INT
Type	VARCHAR(50)
Size	VARCHAR(50)
Capacity	INT
FuelType	VARCHAR(50)
MPG	DECIMAL(4,2)
LicensePlateNumber	VARCHAR(20)
RegistrationExpirationDate	DATE
InsuranceProvider	VARCHAR(50)
InsuranceExpirationDate	DATE
MaintenanceSchedule	VARCHAR(100)
CurrentLocationAndStatus	VARCHAR(100)

Payment	
PaymentID	INT
CustomerID	INT
Amount	DECIMAL(10,2)
PaymentType	VARCHAR(20)
TransactionDate	DATE
PaymentStatus	VARCHAR(20)
TransactionID	VARCHAR(50)
PaymentMethod	VARCHAR(50)
PaymentGatewayName	VARCHAR(50)

Reviews And Ratings	
ReviewID	INT
CustomerID	INT
ProductID	INT
ReviewTitle	VARCHAR(100)
ReviewDescription	VARCHAR(500)
Rating	INT
ReviewDateAndTime	DATETIME

Product Categories	
CategoryID	INT
CategoryName	VARCHAR(50)
CategoryDescription	VARCHAR(100)

Product Attributes	
ProductID	INT
AttributeName	VARCHAR(50)
AttributeValue	VARCHAR(50)

Normalization

Database Normalization is a technique of organizing the data in the database. Normalization is a systematic approach of decomposing tables to eliminate data redundancy(repetition) and undesirable characteristics like Insertion, Update and Deletion Anomalies. It is a multi-step process that puts data into tabular form, removing duplicated data from the relation tables.

Normalization is used for mainly two purposes,

- Eliminating redundant(useless) data.
- Ensuring data dependencies make sense i.e data is logically stored.

If a table is not properly normalized and have data redundancy then it will not only eat up extra memory space but will also make it difficult to handle and update the database, without facing data loss. Insertion, Updation and Deletion Anomalies are very frequent if database is not normalized.

FIRST NORMAL FORM

For a table to be in the First Normal Form, it should follow the following 4 rules:

1. It should only have single(atomic) valued attributes/columns.

Each column of your table should be single valued which means they should not contain multiple values.

2. Values stored in a column should be of the same domain.

This is more of a "Common Sense" rule. In each column the values stored must be of the same kind or type.

For example: If you have a column **dob** to save date of births of a set of people, then you cannot or you must not save 'names' of some of them in that column along with 'date of birth' of others in that column. It should hold only 'date of birth' for all the records/rows.

3. All the columns in a table should have unique names.

This rule expects that each column in a table should have a unique name. This is to avoid confusion at the time of retrieving data or performing any other operation on the stored data.

If one or more columns have same name, then the DBMS system will be left confused.

4. And the order in which data is stored, does not matter.

SECOND NORMAL FORM

For a table to be in the Second Normal Form, it must satisfy two conditions:

1. The table should be in the First Normal Form.
2. There should be no Partial Dependency.

-Partial Dependency exists, when for a composite primary key, any attribute in the table depends only on a part of the primary key and not on the complete primary key.

-To remove Partial dependency, we can divide the table, remove the attribute which is causing partial dependency, and move it to some other table where it fits in well.

THIRD NORMAL FORM

Third Normal Form is an upgrade to Second Normal Form. When a table is in the Second Normal Form and has no transitive dependency, then it is in the Third Normal Form.

For a table to be in the third normal form,

1. It should be in the Second Normal form.
2. And it should not have Transitive Dependency.

The advantage of removing transitive dependency is,

- Amount of data duplication is reduced.
- Data integrity achieved.

BOYCE CODD NORMAL FORM

Boyce-Codd Normal Form or BCNF is an extension to the third normal form, and is also known as 3.5 Normal Form.

For a table to satisfy the Boyce-Codd Normal Form, it should satisfy the following two conditions:

1. It should be in the Third Normal Form.
2. And, for any dependency $A \rightarrow B$, A should be a super key.

it means, that for a dependency $A \rightarrow B$, A cannot be a non-prime attribute, if B is a prime attribute.

FOURTH NORMAL FORM

Fourth Normal Form comes into the picture when Multi-valued Dependency occurs in any relation.

For a table to satisfy the Fourth Normal Form, it should satisfy the following two conditions:

1. It should be in the Boyce-Codd Normal Form.
2. And, the table should not have any Multi-valued Dependency.

A table is said to have a multi-valued dependency, if the following conditions are true,

1. For a dependency $A \twoheadrightarrow B$, if for a single value of A, multiple values of B exist, then the table may have a multi-valued dependency.
2. Also, a table should have at least 3 columns for it to have a multi-valued dependency.
3. And, for a relation $R(A, B, C)$, if there is a multi-valued dependency between, A and B, then B and C should be independent of each other.

If all these conditions are true for any relation(table), it is said to have a multi-valued dependency.

Database Designing

1. Customer:

```
CREATE DATABASE DeliveryXperss;  
USE DeliveryXpress;  
  
CREATE TABLE Customer (  
  CustomerID INT PRIMARY KEY,  
  FirstName VARCHAR(50),  
  LastName VARCHAR(50),  
  Address VARCHAR(100),  
  City VARCHAR(50),  
  State VARCHAR(50),  
  ZipCode VARCHAR(10),  
  PhoneNumber VARCHAR(20),  
  EmailAddress VARCHAR(50),  
  PaymentInformation VARCHAR(50),  
  PurchaseHistory VARCHAR(100),  
  DeliveryAddress VARCHAR(100),  
  PreferredDeliveryTime VARCHAR(50),  
  DeliveryInstructions VARCHAR(100),  
  LoyaltyProgramMembership VARCHAR(50),  
  AccountCreationDate DATE,  
  AccountStatus VARCHAR(20),  
  Password VARCHAR(100),  
  ProfilePicture BLOB  
);
```

2. Order:

```
er:  
CREATE TABLE Order (  
  OrderID INT PRIMARY KEY,  
  CustomerID INT,  
  Address VARCHAR(100),
```

```
PickupAddress VARCHAR(100),
DeliveryDate DATE,
PickupDate DATE,
OrderStatus VARCHAR(20),
PaymentMethod VARCHAR(50),
PaymentStatus VARCHAR(20),
TotalCost DECIMAL(10,2),
QuantityOfItems INT,
NameOfItems VARCHAR(100),
DescriptionOfItems VARCHAR(200),
SpecialInstructions VARCHAR(100),
EstimatedDeliveryTime VARCHAR(50),
DeliveryPersonnel VARCHAR(50
```

3. Delivery Personnel:

```
CREATE TABLE DeliveryPersonnel (
ID INT PRIMARY KEY,
Name VARCHAR(50),
DateOfBirth DATE,
Address VARCHAR(100),
Email VARCHAR(50),
PhoneNumber VARCHAR(20),
Nationality VARCHAR(50),
IdentificationNumber VARCHAR(50),
DateOfHire DATE,
EmployeeStatus VARCHAR(20),
DeliveryVehicleType VARCHAR(50),
VehicleLicensePlateDetails VARCHAR(50),
VehicleInsuranceDetails VARCHAR(100),
DerivingLicenseDetails VARCHAR(100),
UniformSize VARCHAR(20),
ShiftPreferences VARCHAR(50),
WorkExperience VARCHAR(100),
Salary DECIMAL(10,2),
BankAccountDetails VARCHAR(100),
```

```
EmergencyContactInformation VARCHAR(100),
AvailabilitySchedule VARCHAR(100),
PerformanceMetrics VARCHAR(100),
TrainingRecords VARCHAR(100),
PerformanceReviewEvaluationResults VARCHAR(100),
Benefits VARCHAR(100)
);
```

4. Delivery Vehicle:

```
CREATE TABLE DeliveryVehicle (
VehicleID INT PRIMARY KEY,
Make VARCHAR(50),
Model VARCHAR(50),
Year INT,
Type VARCHAR(50),
Size VARCHAR(50),
Capacity INT,
FuelType VARCHAR(50),
MPG DECIMAL(4,2),
LicensePlateNumber VARCHAR(20),
RegistrationExpirationDate DATE,
InsuranceProvider VARCHAR(50),
InsuranceExpirationDate DATE,
MaintenanceSchedule VARCHAR(100),
CurrentLocationAndStatus VARCHAR(100)
);
```

5. Delivery Status:

```
CREATE TABLE DeliveryStatus (
OrderID INT PRIMARY KEY,
Status VARCHAR(20),
DeliveryDate DATE,
EstimatedDeliveryTime VARCHAR(50),
DeliveryAddress VARCHAR(100),
DeliveryAttempts INT,
```

```
DeliveryConfirmation VARCHAR(50),  
DeliveryMethod VARCHAR(50),  
TrackingNumber VARCHAR(50),  
DeliveryPersonnel VARCHAR(50)  
);
```

6. Pickup Location(s):

```
CREATE TABLE PickupLocation (  
LocationID INT PRIMARY KEY,  
Address VARCHAR(100),  
City VARCHAR(50),  
StateOrProvince VARCHAR(50),  
Country VARCHAR(50),  
PostalOrZipCode VARCHAR(10),  
ContactName VARCHAR(50),  
ContactEmail VARCHAR(50),  
Latitude DECIMAL(8,6),  
Longitude DECIMAL(9,6),  
PickupHours VARCHAR(50),  
PickupNotificationRequired BOOLEAN,  
PickupInstructions VARCHAR(100),  
PickupCapacity INT,  
PickupRestrictions VARCHAR(100),  
PickupFees DECIMAL(10,2),  
PickupConfirmationRequired BOOLEAN  
);
```

7. Destination(s):

```
CREATE TABLE Destination (  
DestinationID INT PRIMARY KEY,  
StreetAddress VARCHAR(100),  
City VARCHAR(50),  
StateOrProvinceOrRegion VARCHAR(50),  
PostalOrZipCode VARCHAR(10),  
Country VARCHAR(50),
```

```
ContactName VARCHAR(50),  
ContactPhoneNumber VARCHAR(20),  
ContactEmail VARCHAR(50),  
DeliveryInstructions VARCHAR(100),  
DeliveryNotes VARCHAR(100),  
DeliveryStatus VARCHAR(20),  
DeliveryDateAndTime DATETIME,  
PackageInformation VARCHAR(100)  
);
```

8. Payment:

```
CREATE TABLE Payment (  
PaymentID INT PRIMARY KEY,  
CustomerID INT,  
Amount DECIMAL(10,2),  
PaymentType VARCHAR(20),  
TransactionDate DATE,  
PaymentStatus VARCHAR(20),  
TransactionID VARCHAR(50),  
PaymentMethod VARCHAR(50),  
PaymentGatewayName VARCHAR(50)  
);
```

9. Billing:

```
CREATE TABLE Billing (  
BillID INT PRIMARY KEY,  
CustomerID INT,  
InvoiceDate DATE,  
DueDate DATE,  
PaymentMethod VARCHAR(50),  
PaymentStatus VARCHAR(20),  
AmountDue DECIMAL(10,2),  
Items  
Purchased VARCHAR(100),  
Tax DECIMAL(10,2),
```

```
ShippingCharges DECIMAL(10,2),
Discounts DECIMAL(10,2),
PreviousBalance DECIMAL(10,2),
LateFees DECIMAL(10,2),
PaymentConfirmation VARCHAR(50),
Notes VARCHAR(100)
);
```

10. Customer Support:

```
CREATE TABLE CustomerSupport (
CustomerID INT,
CustomerName VARCHAR(50),
CustomerContactNumber VARCHAR(20),
EmailAddress VARCHAR(50),
MailingAddress VARCHAR(100),
OrdersHistory VARCHAR(100),
OrderStatus VARCHAR(20),
PaymentInformation VARCHAR(50),
DeliveryAddress VARCHAR(100),
DeliveryStatus VARCHAR(20),
DeliveryDateAndTime DATETIME,
ComplaintDescription VARCHAR(100),
ComplaintDateAndTime DATETIME,
ComplaintStatus VARCHAR(20),
ResolutionDescription VARCHAR(100),
ResolutionDateAndTime DATETIME,
FeedbackOnService VARCHAR(100),
FrequencyOfServiceRequests VARCHAR(50),
ModeOfCommunicationPreference VARCHAR(50),
PreferredDeliveryTiming VARCHAR(50)
);
```

11. Delivery History:

```
CREATE TABLE DeliveryHistory (
DeliveryID INT PRIMARY KEY,
```

```
SenderName VARCHAR(50),
SenderAddress VARCHAR(100),
SenderContactNumber VARCHAR(20),
RecipientName VARCHAR(50),
RecipientAddress VARCHAR(100),
RecipientContactNumber VARCHAR(20),
PackageContents VARCHAR(100),
PackageWeight DECIMAL(10,2),
DeliveryDateAndTime DATETIME,
DeliveryStatus VARCHAR(20),
DeliveryPersonName VARCHAR(50),
DeliveryPersonContactNumber VARCHAR(20)
);
```

12. Reviews and Ratings:

```
CREATE TABLE ReviewsAndRatings (
ReviewID INT PRIMARY KEY,
CustomerID INT,
ProductID INT,
ReviewTitle VARCHAR(100),
ReviewDescription VARCHAR(500),
Rating INT,
ReviewDateAndTime DATETIME
);
```

13. Wishlist:

```
CREATE TABLE Wishlist (
WishlistID INT PRIMARY KEY,
CustomerID INT,
ProductID INT,
AddedDateAndTime DATETIME
);
```

14. Coupons:


```
CREATE TABLE Coupons (  
  CouponID INT PRIMARY KEY,  
  CouponCode VARCHAR(50),  
  DiscountPercentage DECIMAL(5,2),  
  ExpirationDate DATE  
);
```

15. Refunds and Returns:

```
CREATE TABLE RefundsAndReturns (  
  ReturnID INT PRIMARY KEY,  
  CustomerID INT,  
  ProductID INT,  
  OrderID INT,  
  ReturnReason VARCHAR(100),  
  ReturnDateAndTime DATETIME,  
  RefundAmount DECIMAL(10,2),  
  RefundMode VARCHAR(50),  
  RefundStatus VARCHAR(20),  
  RefundDateAndTime DATETIME  
);
```

16. Product Categories:

```
CREATE TABLE ProductCategories (  
  CategoryID INT PRIMARY KEY,  
  CategoryName VARCHAR(50),  
  CategoryDescription VARCHAR(100)  
);
```

17. Product Category Mapping:

```
CREATE TABLE ProductCategoryMapping (  
  ProductID INT,  
  CategoryID INT
```

```
);
```

18. Product Reviews:

```
CREATE TABLE ProductReviews (  
  ReviewID INT PRIMARY KEY,  
  ProductID INT,  
  CustomerID INT,  
  ReviewTitle VARCHAR(100),  
  ReviewDescription VARCHAR(500),  
  Rating INT,  
  ReviewDateAndTime DATETIME  
);
```

19. Product Specifications:

```
CREATE TABLE ProductSpecifications (  
  ProductID INT,  
  SpecificationName VARCHAR(50),  
  SpecificationValue VARCHAR(50)  
);
```

20. Product Attributes:

```
CREATE TABLE ProductAttributes (  
  ProductID INT,  
  AttributeName VARCHAR(50),  
  AttributeValue VARCHAR(50)  
);
```