Designing an E-commerce Platform Database

For

Information systems and Data bases
Course

Prof. Mauro Iacono

Authors:
Mohammdhossein Darabi

The E-commerce Platform

The E-commerce Platform is an online system designed to help with buying and selling products and services. It manages many activities to ensure that customers have a smooth shopping experience. This platform manages products by giving each one a unique ID and providing details such as name, description, price, and stock quantity. Products are sorted into categories to make browsing easier for customers. Each product is linked to a supplier through an inventory system who provides the product, helping to keep inventory levels accurate. The platform also supports selling services, including subscriptions and training programs. These services are delivered over specified time periods. Each service has details such as service name, description, price, and duration. Subscriptions provide access to certain benefits or content for a set period, while training programs can include scheduled sessions.

Customer information is recorded carefully. Each customer has a unique ID with personal details like name, email, phone number, and address. Customers can be individuals or companies. For companies, the platform allows the creation of a framework contract and recording Contact_Person. This contract specifies the terms under which the company can receive goods or services, ensuring they have continuous access to needed resources and for individuals the platform store Date of Birth, Gender.

When a customer places an order, the order processing system takes care of everything. Each order gets a unique ID and includes details such as the products or services ordered, quantities, prices, and the total amount. The system tracks the status of each order from when it is placed until it is delivered.

Suppliers are important for keeping the platform's product inventory up to date. Each supplier has a profile with details like the company name, contact information, and the products they provide. The platform makes it easy to communicate with suppliers to fulfill orders and restock inventory, ensuring products are always available for customers. Customer feedback is very important. Customers can leave reviews and ratings for products and services they have bought. Each review is connected to the customer and the product or service. Reviews include a rating and a written comment, which helps other customers make decisions and helps improve the products and services offered.

Managing inventory is crucial to avoid running out of stock or having too much. The platform tracks how many units of each product are available and Product, Last Restock Date, Warehouse Location and updates the stock levels in real-time as orders are placed and fulfilled. If stock levels fall below a certain point, alerts are generated to prompt reorders from suppliers.

Shipping and delivery are also managed by the platform. It records shipping addresses for each order and offers various shipping options for customers. The system works with shipping companies to provide tracking information and estimated delivery times, ensuring that products reach customers on time.

Restructuring Specification
(Homogenous phrases)

Phrases of a general nature

The E-commerce Platform is an online system designed to facilitate buying and selling of products and services. It manages various activities to ensure a smooth shopping experience for customers.

Phrases related to the Products

This platform assigns a unique ID to each product and records details such as name, description, price, and stock quantity. Products are grouped into categories to simplify browsing for customers. Each product is linked to a supplier through an inventory system, which helps in maintaining accurate inventory levels.

Phrases related to the Services

The platform also supports the sale of services, including subscriptions and training programs. These services are delivered over specific durations. Each service includes details such as service name, description, price, and duration. Subscriptions grant access to certain benefits or content for a defined period, while training programs include scheduled sessions.

Phrases related to the Customers

Customer information is carefully recorded. Each customer has a unique ID with personal details like name, email, phone number, and address. Customers can be individuals or companies. For companies, the platform allows the creation of a framework contract specifying the terms for receiving goods or services, ensuring continuous access to necessary resources and recording information related to Contact_Person.For individuals the platform store Date of Birth, Gender.

Phrases related to the Orders

When a customer places an order, the order processing system manages all details. Each order receives a unique ID and includes information such as the products or services ordered, quantities, prices, and total amount. The system tracks the status of each order from placement to delivery.

Phrases related to the Suppliers

Suppliers are crucial for keeping the platform's product inventory updated. Each supplier has a profile with details like company name, contact information, and the products they provide. The platform facilitates communication with suppliers to fulfill orders and restock inventory, ensuring product availability for customers.

Phrases related to the Reviews

Customer feedback is essential. Customers can leave reviews and ratings for products and services they have purchased. Each review is linked to the customer and the product or service. Reviews include a rating and a written comment, helping other customers make decisions and improving the products and services offered.

Phrases related to the Inventory

Managing inventory is vital to avoid stockouts or overstocking. The platform tracks the number of units available for each product and Product, Last Restock Date, Warehouse Location and updates stock levels in real-time as orders are placed and fulfilled.

Phrases related to the Shipping

Shipping and delivery are managed by the platform. It records shipping addresses for each order and offers various shipping options. The system integrates with shipping companies to provide tracking information and estimated delivery times, ensuring timely delivery to customers.

Glossary of Terms
(Static Aspect of Description)

TERM	DESCRIPTION	SYNONYM	LINKS
Product	An item available for sale on the platform, identified by a unique ID, with details such as name, description, price, and stock quantity.	Item, Good	Item, Category, Inventory
Category	A classification for products to facilitate easier browsing for customers.	Group, Section	Product
Supplier	A provider of products, with a profile including company name, contact information, and the products they supply.	Vendor, Provider	Inventory
Service	A non-physical item for sale, including subscriptions and training programs, with details such as name, description, price, and duration.	_	Subscription, Training Program
Training Program	A service consisting of scheduled sessions for educational purposes.	_	Service
Subscription	A service providing access to certain benefits or content for a specified period.	Service, Customer	Service
Customer	An individual or company purchasing products or services, with a unique profile containing personal details like name, email, phone number, and address.	Client, Buyer	Order,Individual, Company,

Order	A record of a customer's purchase, with a unique ID and details such as products or services ordered, quantities, prices, and total amount.	Purchase, Transaction	Customer,Item, Shipping Company, Review
Review	Customer feedback on products or services, including a rating and written comment, linked to the customer's profile and the product or service.	Feedback, Rating	Order
Inventory	The system for tracking stock levels of products, updated in real-time as orders are placed and fulfilled.	Stock, Supply	Product, Supplier
Shipping Company	The process of delivering products to customers, including recording shipping addresses, offering shipping options, and providing tracking information.	Delivery, Shipment	Order
Framework Contract	A contract for companies specifying terms under which they receive goods or services, ensuring continuous access to needed resources.	Agreement, Deal	Customer, Service
Unique ID	A distinctive identifier assigned to products, services, orders, and customers to ensure accurate tracking and management.	Identifier, Code	Product, Service, Order, Customer,…

Operations on Data
(Dynamic Aspect of Description)

Operations Relating to Customers

- Add Customer: Create a new customer .
- Update Customer Information: Modify details such as name, email, phone number, and address.
- Delete Customer: Remove a customer.
- Manage Customer Types: Distinguish between individual and company customers.
- Link Customer to Address: Associate multiple addresses with a customer profile.

Operations Relating to Individual Customers

- Add Individual Customer: Create a new individual customer .
- Update Individual Customer Details: Modify individual-specific details such as date of birth and gender.

Operations Relating to Company Customers

- Add Company Customer: Create a new company customer profile.
- Update Company Customer Details: Modify company-specific details such as company name, contact person.
- Manage Framework Contracts: Create, update, and manage contracts with company customers.

Operations Relating to Products

- Add Product: Create a new product entry.
- Update Product Details: Modify product-specific details such as name, description, price, stock quantity.
- Delete Product: Remove a product entry.
- Link Product to Category: Assign a product to a category.
- Link Product to Supplier: Associate a product with a supplier.

Operations Relating to Categories

- Add Category: Create a new product category.
- Update Category Details: Modify category-specific details such as name and description.
- Delete Category: Remove a category.

Operations Relating to Suppliers

- Add Supplier: Create a new supplier profile.
- Update Supplier Details: Modify supplier-specific details such as company name and contact information.
- Delete Supplier: Remove a supplier profile.
- Link Supplier to Inventory: Associate products with a supplier.

Operations Relating to Subscriptions

- Add Subscription: Create a new subscription.
- Update Subscription Details: Modify subscription-specific details such as name, description, price, duration, and benefits.
- Delete Subscription: Remove a subscription.
- Link Subscription to Services: Associate services with a subscription.

Operations Relating to Training Programs

- Add Training Program: Create a new training program.
- Update Training Program Details: Modify training program-specific details such as name, description, price, duration, and schedule.
- Delete Training Program: Remove a training program.
- Link Training Program to Services: Associate services with a training program.

Operations Relating to Orders

- Add Order: Create a new order.
- Update Order Details: Modify order-specific details such as order date, total amount, order status, and delivery address.
- Delete Order: Remove an order.

- Link Order to Products and Services: Associate products and services with an order.
- Track Order Status: Monitor the status of each order from placement to delivery.

Operations Relating to Reviews

- Add Review: Create a new review.
- Update Review Details: Modify review-specific details such as rating and comment.
- Delete Review: Remove a review.

Operations Relating to Inventory

- Add Inventory Record: Create a new inventory entry.
- Update Inventory Details: Modify inventory-specific details such as product, quantity available, last restock date, and warehouse location.
- Delete Inventory Record: Remove an inventory entry.

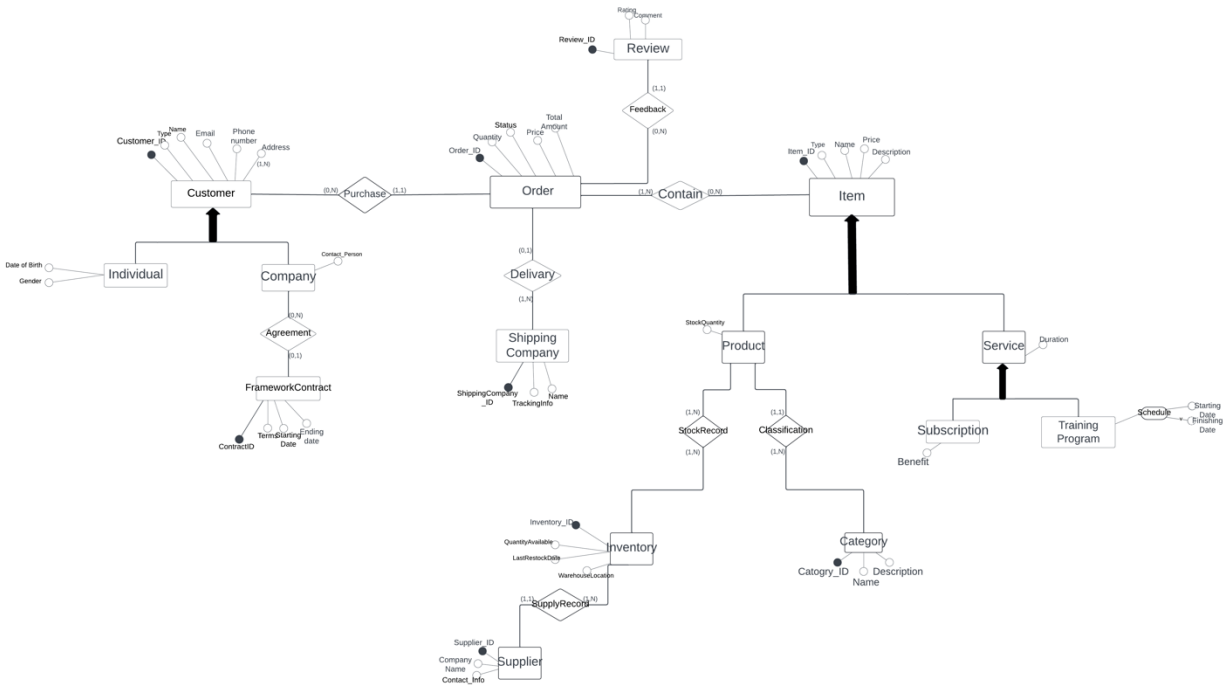
Operations Relating to Shipping Companies

- Add Shipping Company: Create a new shipping company profile.
- Update Shipping Company Details: Modify shipping company-specific details such as name, contact information, and tracking information.
- Delete Shipping Company: Remove a shipping company profile.
- Link Shipping Company to Orders: Associate a shipping company with orders.

Operations Relating to Framework Contracts

- Add Framework Contract: Create a new framework contract.
- Update Framework Contract Details: Modify contract-specific details such as terms, start date, and end date.
- Delete Framework Contract: Remove a framework contract.

Entity Relationship Diagram



Preliminary Reasoning – Entities & Relationships

ITEM: The Item entity is designed to represent both products and services available for sale on the platform represented by a generalizatiom, ensuring a unified approach to managing different types of offerings. This entity is identified by a unique Item_ID, facilitating easy tracking and management.Additionally, connecting items to categories helps organize the offerings, making browsing easier for customers. The relationship with suppliers ensures that inventory levels are maintained accurately, with each item linked to its provider.

PRODUCT: The Product entity, identified by a unique Product_ID, includes details such as name, description, price, and stock quantity. By linking products to categories and suppliers, the platform can manage the product offerings effectively. The relationship with inventory ensures accurate stock levels and prompt restocking when necessary.

SERVICE: The Service entity, identified by a unique Service_ID, includes details such as name, description, price, duration, and schedule. This entity allows the platform to offer various services, including subscriptions and training programs. The relationship with subscriptions ensures that customers have access to services for a specified period, while the training program entity manages scheduled sessions.

SUBSCRIPTION: The Subscription entity manages the access to services over a specified period. Identified by a unique Subscription_ID, this entity includes details such as name, description, price, and duration. By linking subscriptions to services, the platform ensures that customers receive the benefits they are entitled to during the subscription period.

TRAINING PROGRAM: The Training Program entity manages scheduled sessions for training services. Identified by a unique Program_ID, this entity includes details such as name, description, price, duration, and schedule. By linking training programs to services, the platform ensures that customers have access to structured training sessions.

CUSTOMER: The Customer entity include both individual and company customers, represented by a generalizatiom, providing a comprehensive approach to managing customer data. This entity is identified by a unique Customer_ID, and through generalization, it handles specific attributes for individuals and companies (like CompanyName and ContactPerson). This approach allows the platform to cater to different customer types while maintaining a consistent structure. The relationships established with the Order entity ensure that customer activities, such as placing orders and leaving feedback, are accurately recorded and managed.

ORDER: The Order entity captures all purchase transactions made by customers, providing a detailed record of each order. Identified by a unique Order_ID, this entity includes necessary details such as the order date, total amount, and order status. By linking orders to customers through a one-to-many relationship, the platform can track all orders placed by a customer.But we can eliminat total amount as an attribute cause it can be obtained by multiplying quantity by price in each order .

SUPPLIER: The Supplier entity is essential for maintaining accurate inventory levels and ensuring that products are available for customers. Identified by a unique Supplier_ID, this entity includes details about the suppliers, such as company name and contact information. By linking suppliers to products through Inventory Entity, the platform can manage products each supplier provides. This connection helps in restocking inventory and maintaining product availability, ensuring a smooth supply chain operation.

REVIEW: The Review entity captures customer feedback on products and services that they ordered, which is vital for improving offerings and aiding other customers in making informed decisions. Identified by a unique Review_ID. This ensures that reviews are tied to specific orders, providing context for the feedback.

INVENTORY: The Inventory entity plays a critical role in managing stock levels and ensuring product availability as an intermediate entity between Supplier and Product. Identified by a unique Inventory_ID, this entity tracks the quantity of each product available, the last restock date, and the warehouse location. By linking inventory records to products and suppliers, the platform can efficiently manage stock levels, prompt reorders when necessary, and maintain accurate records of product availability. This structure helps prevent stockouts.

SHIPPING COMPANY: The Shipping Company entity manages delivering products to customers. Identified by a unique ShippingCompany_ID, this entity includes details such as the company name and tracking information. By linking shipping companies to orders, the platform ensures that each order is associated with the appropriate shipping method and tracking details are provided.

FRAMEWORK CONTRACT: The Framework Contract entity is designed to manage the terms of agreements with company customers, ensuring continuous access to necessary resources. Identified by a unique Contract_ID, this entity includes details such as contract terms, start date, and end date. By linking framework contracts to company customers, the platform can handle long-term agreements and provide consistent service to companies. This structure supports business relationships.

CATEGORY: The Category entity organizes products into different groups, making it easier for customers to browse offerings. Identified by a unique Category_ID, this entity includes details such as name and description. By linking products to categories, the platform ensures a structured and user-friendly browsing experience.

Preliminary Reasoning – Cardinalities

1. Customer to Order

- Customer (0, N)
  - Minimum Cardinality: 0 - A customer can exist without placing any orders.
  - Maximum Cardinality: N - A customer can place multiple orders over time.
- Order (1, 1)
  - Minimum Cardinality: 1 - An order must be placed by one customer.
  - Maximum Cardinality: 1 - An order can be placed by only one customer.

2. Order to Review

- Order (0, N)
  - Minimum Cardinality: 0 – An order might not have any review.
  - Maximum Cardinality: N – An order can have multiple reviews for different products or service.
- Review (1, 1)
  - Minimum Cardinality: 1 - A review must be associated with one order Maximum Cardinality: 1 - A review can be linked to only one order .

3. Company to Framework Contract

- Company (0, N)
  - Minimum Cardinality: 0 - A company can have no framework contract.
  - Maximum Cardinality: N - A company can have multiple contracts over time.
- Framework Contract (0, 1)
  - Minimum Cardinality: 0 - A contract can exist without being linked to a company (e.g., drafts or templates).
  - Maximum Cardinality: 1 - A contract can not be linked to multiple companies .

4. Order to Shipping Company

- Order (0, 1)
  - Minimum Cardinality: 1 - An order can have no shipping companies at first.
  - Maximum Cardinality: 1 - An order can be handled by only one shipping company.
- Shipping Company (1, N)
  - Minimum Cardinality: 1 - A shipping company must handle at least one order.
  - Maximum Cardinality: N - A shipping company can handle multiple orders.

5. Product to Inventory

- Product (1, N)
  - Minimum Cardinality: 1 - Each product must have an inventory record.
  - Maximum Cardinality: N - A product can have multiple inventory records over time.
- Inventory (1, N)
  - Minimum Cardinality: 1 - An inventory record must be associated with a product.
  - Maximum Cardinality: N - An inventory record can include multiple products

6. Inventory to Supplier

- Inventory (1, N)
  - Minimum Cardinality: 1 - Each inventory record must be associated with one supplier.
  - Maximum Cardinality: N - Each inventory record can be linked to multiple suppliers over time.
- Supplier (1, 1)
  - Minimum Cardinality: 1 - Each supplier must supply at least one inventory record.
  - Maximum Cardinality: N - Each supplier can supply multiple inventory records.

7. Product to Category

- Product (1, 1)
  - Minimum Cardinality: 1 - Each product must belong to one category at least.
  - Maximum Cardinality: 1 - A product can belong to multiple categories.
- Category (1, N)
  - Minimum Cardinality: 1 - Each category must include at least one product.
  - Maximum Cardinality: N - Each category can include multiple products.

Data Dictionary - Entities

ENTITY	DESCRIPTION	ATTRIBUTES	IDENTIFIER
Product	An item available for purchase on the platform.	Item_ID, Type,Name, Description, Price, StockQuantity,	Item_ID,Type
Category	A classification for products.	Category_ID, Name, Description	Category_ID
Supplier	A provider of products.	Supplier_ID, CompanyName, ContactInfo	Supplier_ID
Service	Non-physical items like subscriptions or training programs.	Item_ID, Type ,Name, Description, Price, Duration, Schedule, ServiceType	Item_ID,Type
Subscription	Access to benefits or content for a period.	Item_ID, Type ,Name, Description, Price, Duration, Benefits	Item_ID,Type
TrainingProgram	A scheduled training session.	TrainingID, Name, Description, Price, Duration, Schedule	TrainingID
Customer	A person or company purchasing products or services.	Customer_ID,Name , Email, PhoneNumber, Address, Type	Customer_ID, Type
Order	A record of a customer's purchase.	Order_ID,Quantity, Price ,TotalAmount	Order_ID

Inventory	Tracks product stock levels.	Inventory_ID, QuantityAvailable, LastRestockDate	Inventory_ID
Shipping	Process of delivering products.	ShippingCompany_ID, Name, TrackingInfo	ShippingCompany_ID
FrameworkContract	Terms for supplying goods/services to companies.	Contract_ID, Terms, StartDate, EndDate	Contract_ID
Review	Feedback on products or services.	Review_ID,Rating, Comment	Review_ID

Data Dictionary - Relationships

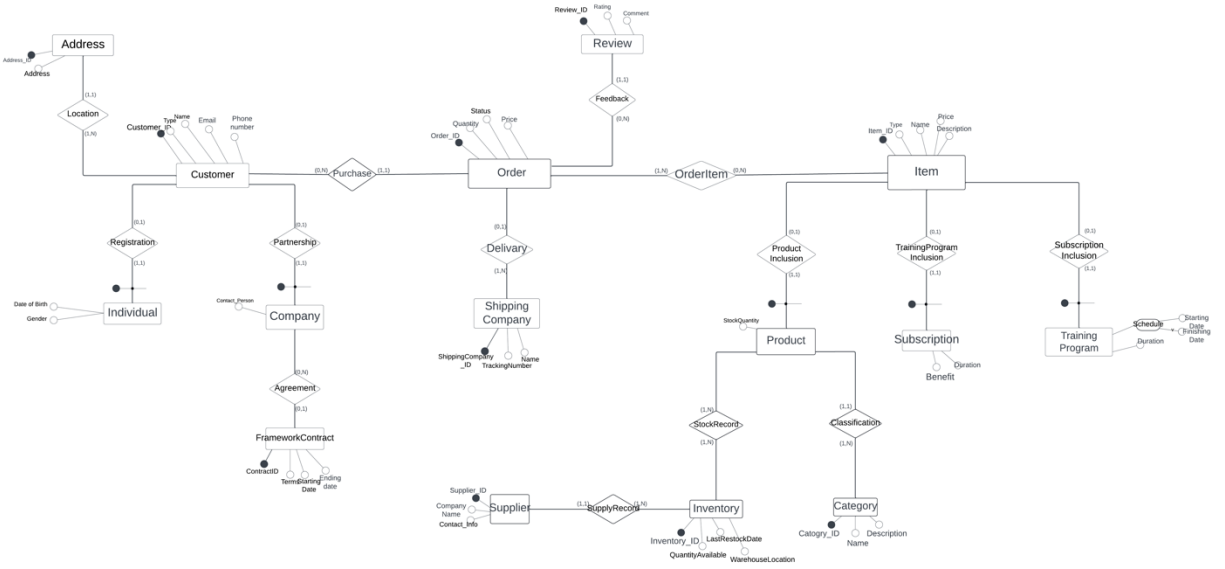
RELATIONSHIP	DESCRIPTION	ENTITIES INVOLVED
Classification	Links products to their categories.	Product-Category
StockRecord	A classification for products.	Product-Inventory
Purchase	Links orders to customers.	Order-Customer
OrderItem	Order Contains Items	Order-Item
FeedBack	Links reviews to orders.	Review-Order
StockRecord	Links suppliers to inventory records .	Inventory-Product
Delivary	A record of a customer's purchase.	Shipping-Order
Agreement	Links framework contracts to companies.	FrameworkContract-Company

Business Rules

- Products on the platform must be linked to at least one supplier, ensuring that the source of the products is clearly identified and inventory can be managed accurately.
- orders must specify a Shipping Company and delivary address, facilitating accurate and timely delivery of products to customers. This ensures that the order processing system can manage all aspects of the order from placement to delivery.
- Framework contracts must link to company customers to manage long-term agreements effectively. This ensures that the contractual obligations are clear and that companies receive consistent service, supporting sustained business relationships.
- Digital services must be accessed within their subscription period.
- Each customer profile can include multiple addresses for different purposes but the customer must finalize one as the primary delivery address
- Orders cannot be placed unless a primary delivery address is specified in the customer profile.
- Real-time updates to stock levels are essential as orders are placed and fulfilled, ensuring the platform reflects current availability and helps manage inventory effectively.
- All attributes refering to Description for some involved entites must be single-line sentences and should not include multiple clauses or high complexity.
- Generate alerts when stock levels fall below a certain threshold to prompt reorders from suppliers.
- Orders must include tracking information provided by shipping companies.
- Customers must receive updates on the status of their orders.
- Each review must be linked to the customer and the product or service being reviewed.
- Reviews must include a rating and a written comment.
- The system must support the association of suppliers with products to maintain accurate inventory levels.
- The platform must facilitate communication with suppliers to fulfill orders and restock inventory.
- Ensure timely delivery to customers by working with shipping companies and providing tracking information.

Translation Into Relational Model

1. Restructuring ER diagram



Preliminary Reasoning – Restructuring ER

- **Address Attribute as a Multivalued Attribute**

The attribute Address for customers is multivalued because customers can have multiple addresses. Representing Address as a separate entity allows us to efficiently manage multiple addresses per customer. This approach ensures data normalization, reduces redundancy, and maintains data integrity by linking addresses to customers through a one-to-many relationship.

- **Removing Generalization of Customer**

Removing the generalization of the Customer entity while keeping separate entities for Individual and Company reduces access complexity. Our operations mainly refer to the general Customer entity, and having a direct Customer entity minimizes the need for multiple accesses. Retaining the Individual and Company entities allows us to cater to operations that need to access specific attributes unique to these entities, such as Date of Birth and Gender for individuals, and Company Name and Contact Person for companies. This approach optimizes data retrieval and management, ensuring both efficiency and flexibility in handling different customer types.

- **Term attribute of Framework Contract Entity**

The Term attribute in the Framework Contract is retained as a single attribute due to limited information in the database description and infrequent operations based on user requirements. This simplifies the schema, ensuring efficient data management. The Term attribute will only accept brief, single-line descriptions, and this must be specified in the business rules to maintain data consistency and manageability.

- **TotalAmount attribute of Order Entity**

The TotalAmount attribute in the Order entity has been eliminated to avoid redundancy, as it can be calculated by multiplying the Price and Quantity attributes already present in the Order entity. This ensures data consistency and reduces unnecessary storage, simplifying the schema while maintaining the necessary information.

- **Removing Generalization of Service**

The generalization of the Service entity has been removed, We chose to substitute the generalization with relationships. User requirements specified operations that only referred to occurrences of the service entities rather than service itself, such as subscriptions or training programs, rather than the service itself. and therefore make distinctions between child and parent entities.

- **Removing Generalization of Item Entity**

The generalization of the Item entity, which included Product, TrainingProgram, and Subscription, has been removed. We chose to substitute the generalization with relationships, which transforms the generalization into multiple one-to-one relationships linking the parent entity (Item) with the child entities (Product, TrainingProgram, and Subscription). This approach ensures that operations can refer to specific types of items, reducing complexity and maintaining clarity in the data model. Additionally, this structure allows each occurrence of an Item to be linked only to one specific type of entity, preventing redundancy and maintaining data integrity.

- **StockQuantity attribute of Product Entity**

The StockQuantity attribute in the Product entity has been eliminated to avoid redundancy, as it can be obtain by the related attribute in Inventory entity. This ensures data consistency and reduces unnecessary storage, simplifying the schema while maintaining the necessary information.

- **Benefit attribute of Subscription Entity**

The Benefit attribute in the Subscription entity presents ambiguity and should ideally be represented as a separate entity to capture the details of this concept clearly. However, due to the lack of specific information in the database description about how the benefits of subscriptions are structured, we have retained it as a single attribute. This approach keeps the schema simple but acknowledges that further analysis and requirements gathering are necessary to properly define and represent subscription benefits. This simplification is temporary and highlights the need for future updates based on more detailed user requirements.

Preliminary Reasoning – Cardinalities

1. Customer to Address

- Customer (1, N)
  - Minimum Cardinality: 1 - A customer must have at least one address for order delivery.
  - Maximum Cardinality: N - A customer can have multiple addresses (e.g., home, work).
- Address (0, 1)
  - Minimum Cardinality: 0 - An address can exist without being linked to a customer (e.g., addresses stored for future use).
  - Maximum Cardinality: 1 - An address typically belongs to one customer.

2. Customer to Individual

- Customer (0, 1)
  - Minimum Cardinality: 0 - Not all customers are individuals.
  - Maximum Cardinality: 1 - A customer can be linked to only one individual.
- Individual (1, 1)
  - Minimum Cardinality: 1 - An individual must be a customer.
  - Maximum Cardinality: 1 - An individual can only be one customer.

3. Customer to Company

- Customer (0, 1)
  - Minimum Cardinality: 0 - Not all customers are companies.
  - Maximum Cardinality: 1 - A customer can be linked to only one company.
- Company (1, 1)
  - Minimum Cardinality: 1 - A company must be a customer.
  - Maximum Cardinality: 1 - A company can only be one customer.

4. Item to Product

- Item (1, 1)
  - Minimum Cardinality: 1 - A product is always an item.
  - Maximum Cardinality: 1 - An item can be a product.
- Product (0, 1)
  - Minimum Cardinality: 0 - Not all items are products.
  - Maximum Cardinality: 1 - A product can be an item.

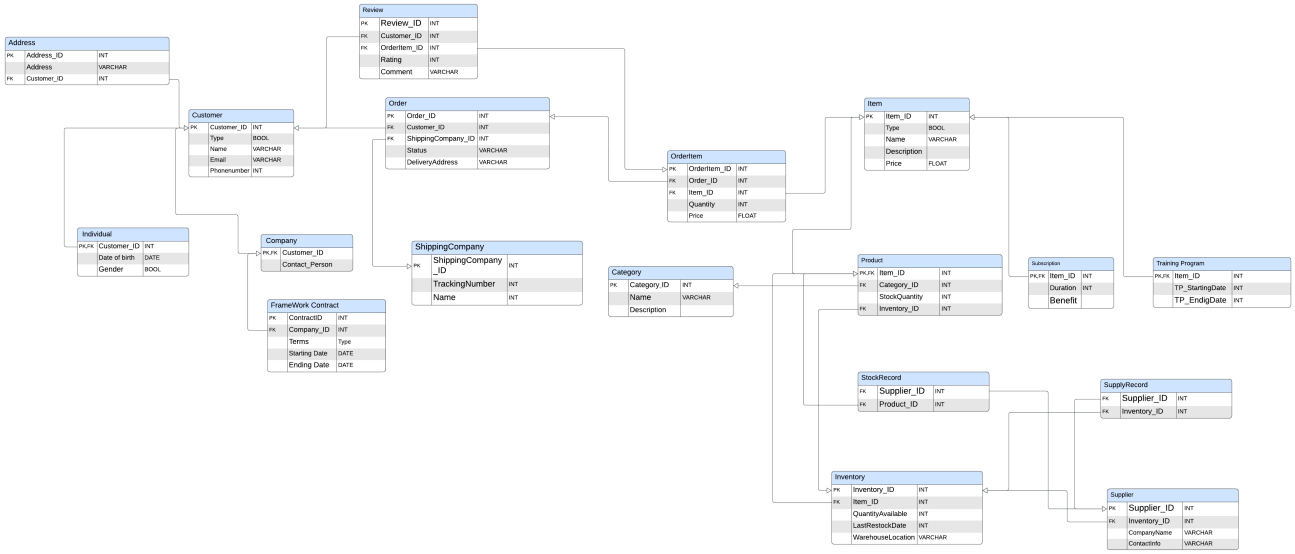
5. Item to Subscription

- Item (1, 1)
  - Minimum Cardinality: 1 - A subscription is always an item.
  - Maximum Cardinality: 1 - An item can be a subscription.
- Subscription (0, 1)
  - Minimum Cardinality: 0 - Not all items are subscriptions.
  - Maximum Cardinality: 1 - A subscription can be an item.

6. Item to TrainingProgram

- Item (1, 1)
  - Minimum Cardinality: 1 - A training program is always an item.
  - Maximum Cardinality: 1 - An item can be a training program.
- TrainingProgram (0, 1)
  - Minimum Cardinality: 0 - Not all items are training programs.
  - Maximum Cardinality: 1 - A training program can be an item.

2. Translation into Logical Model



Preliminary Reasoning – Normalization

1. Address: This table stores multiple addresses for each customer, linked via Customer_ID.

- 1NF: Address is in 1NF as each attribute (Address_ID, Address) is atomic and each tuple is unique.
- 2NF: Since the primary key Address_ID is a single attribute, there are no partial dependencies (all non-key attributes depend on the entire primary key).
- 3NF: There are no transitive dependencies as there are no non-key attributes that depend on other non-key attributes.

2. Customer: This entity stores general information about all customers, differentiating them by type (individual or company) using the CustomerType attribute.

- 1NF: Customer is in 1NF as each attribute (Customer_ID, Name, Email, PhoneNumber, CustomerType) is atomic and each tuple is unique.
- 2NF: Customer_ID is the single primary key, ensuring no partial dependencies (all non-key attributes depend on the entire primary key).
- 3NF: No transitive dependencies exist as all non-key attributes depend solely on the primary key Customer_ID.

3. Individual: This table stores specific information for individual customers, linked to the Customer table via Customer_ID.

- 1NF: Each attribute (Customer_ID, DateOfBirth, Gender) is atomic and each tuple is unique.
- 2NF: The primary key Customer_ID ensures no partial dependencies as all non-key attributes depend on the entire primary key.
- 3NF: No transitive dependencies as all non-key attributes depend solely on the primary key Customer_ID.

4. Company: This table stores specific information for company customers, linked to the Customer table via Customer_ID

- 1NF: Each attribute (Customer_ID, CompanyName, ContactPerson) is atomic and each tuple is unique.
- 2NF: The primary key Customer_ID ensures no partial dependencies as all non-key attributes depend on the entire primary key.
- 3NF: No transitive dependencies as all non-key attributes depend solely on the primary key Customer_ID.

5. FrameworkContract: This table stores framework contracts for company customers, linked via Customer_ID.

- 1NF: Each attribute (Contract_ID, Customer_ID, Terms, StartDate, EndDate) is atomic and each tuple is unique.
- 2NF: The primary key Contract_ID ensures no partial dependencies as all non-key attributes depend on the entire primary key.
- 3NF: No transitive dependencies as all non-key attributes depend solely on the primary key Contract_ID.

6. Order: This table stores information about orders placed by customers, linked to customers via Customer_ID.

- 1NF: Each attribute (Order_ID, Customer_ID, Status, ShippingCompany_ID) is atomic and each tuple is unique.
- 2NF: The primary key Order_ID ensures no partial dependencies as all non-key attributes depend on the entire primary key.
- 3NF: No transitive dependencies as all non-key attributes depend solely on the primary key Order_ID.

7. OrderItem: This table represents the many-to-many relationship between orders and items, storing details about the items in each order.

- 1NF: Each attribute (OrderItem_ID, Order_ID, Item_ID, Quantity, Price) is atomic and each tuple is unique.
- 2NF: The primary key OrderItem_ID ensures no partial dependencies as all non-key attributes depend on the entire primary key.
- 3NF: No transitive dependencies as all non-key attributes depend solely on the primary key OrderItem_ID.

8. ShippingCompany: This table stores information about shipping companies used to deliver orders.

- 1NF: Each attribute (ShippingCompany_ID, Name, TrackingNumber) is atomic and each tuple is unique.
- 2NF: The primary key ShippingCompany_ID ensures no partial dependencies as all non-key attributes depend on the entire primary key.
- 3NF: No transitive dependencies as all non-key attributes depend solely on the primary key ShippingCompany_ID.

9. Item: This table stores information about all items (both products and services) available on the platform.

- 1NF: Each attribute (Item_ID, Name, Description, Price) is atomic and each tuple is unique.
- 2NF: The primary key Item_ID ensures no partial dependencies as all non-key attributes depend on the entire primary key.
- 3NF: No transitive dependencies as all non-key attributes depend solely on the primary key Item_ID.

10. Product: This table stores specific information for products, linked to the Item table via Item_ID.

- 1NF: Each attribute (Item_ID, StockQuantity) is atomic and each tuple is unique.
- 2NF: The primary key Item_ID ensures no partial dependencies as all non-key attributes depend on the entire primary key.
- 3NF: No transitive dependencies as all non-key attributes depend solely on the primary key Item_ID.

11. Subscription: This table stores information about subscriptions, linked to the Item table via Item_ID.

- 1NF: Each attribute (Item_ID, Benefit) is atomic and each tuple is unique.
- 2NF: The primary key Item_ID ensures no partial dependencies as all non-key attributes depend on the entire primary key.
- 3NF: No transitive dependencies as all non-key attributes depend solely on the primary key Item_ID.

12. TrainingProgram: This table stores information about training programs, linked to the Item table via Item_ID.

- 1NF: Each attribute (Item_ID, Schedule) is atomic and each tuple is unique.
- 2NF: The primary key Item_ID ensures no partial dependencies as all non-key attributes depend on the entire primary key.
- 3NF: No transitive dependencies as all non-key attributes depend solely on the primary key Item_ID.

13. Inventory: This table stores inventory information for products, linked to products via Product_ID

- 1NF: Each attribute (Inventory_ID, Product_ID, QuantityAvailable, LastRestockDate, WarehouseLocation) is atomic and each tuple is unique.
- 2NF: The primary key Inventory_ID ensures no partial dependencies as all non-key attributes depend on the entire primary key.

- 3NF: No transitive dependencies as all non-key attributes depend solely on the primary key Inventory_ID.

14. Supplier: This table stores information about suppliers providing products

- 1NF: Each attribute (Supplier_ID, CompanyName, ContactInfo) is atomic and each tuple is unique.
- 2NF: The primary key Supplier_ID ensures no partial dependencies as all non-key attributes depend on the entire primary key.
- 3NF: No transitive dependencies as all non-key attributes depend solely on the primary key Supplier_ID.

15. SupplyRecord: This table captures the relationship between suppliers and inventory records.

- 1NF: Each tuple representing the relationship between Inventory_ID and Supplier_ID is atomic.
- 2NF: The composite primary key (Inventory_ID, Supplier_ID) ensures all non-key attributes depend on the entire key.
- 3NF: No transitive dependencies exist as no non-key attributes depend on other non-key attributes.

16. Review: This table stores reviews linked to orders, capturing customer feedback on items ordered.

- 1NF: Each attribute (Review_ID, Order_ID, Rating, Comment) is atomic and each tuple is unique.
- 2NF: The primary key Review_ID ensures no partial dependencies as all non-key attributes depend on the entire primary key.
- 3NF: No transitive dependencies as all non-key attributes depend solely on the primary key Review_ID.

17. Category: This table stores product categories, used to organize items.

- 1NF: Each attribute (Category_ID, Name, Description) is atomic and each tuple is unique.
- 2NF: The primary key Category_ID ensures no partial dependencies as all non-key attributes depend on the entire primary key.
- 3NF: No transitive dependencies as all non-key attributes depend solely on the primary key Category_ID.

SQL Queris

Customer-Related Queries

1. Retrieve All Customer Details

```
SELECT *
FROM Customer;
```

2. Retrieve All Individual Customers

```
SELECT c.Customer_ID, c.Name, c.Email, c.Phonenumber,
i.Date_of_birth, i.Gender
FROM Customer c
JOIN Individual i ON c.Customer_ID = i.Customer_ID;
```

3. Retrieve All Company Customers

```
SELECT c.Customer_ID, c.Name, c.Email, c.Phonenumber,
co.Contact_Person
FROM Customer c
JOIN Company co ON c.Customer_ID = co.Customer_ID;
```

4. Retrieve All Addresses for a Customer

```
SELECT a.Address
FROM Address a
WHERE a.Customer_ID = 1;
```

Order-Related Queries

1. List All Orders

```
SELECT *
FROM Order;
```

2. List All Orders for a Specific Customer

```
SELECT o.Order_ID, o.Status, o.DeliveryAddress
FROM Order o
WHERE o.Customer_ID = 1;
```

3. Retrieve Order Items for a Specific Order

```
SELECT oi.OrderItem_ID, i.Name, oi.Quantity, oi.Price
FROM OrderItem oi
JOIN Item i ON oi.Item_ID = i.Item_ID
WHERE oi.Order_ID = 1;
```

4. Calculate Total Amount for a Specific Order

```
SELECT SUM(oi.Quantity * oi.Price) AS TotalAmount
FROM OrderItem oi
WHERE oi.Order_ID = 1;
```

Product and Inventory Queries

1. Retrieve All Products

```
SELECT i.Item_ID, i.Name, i.Description, i.Price, p.StockQuantity
FROM Item i
JOIN Product p ON i.Item_ID = p.Item_ID;
```

2. Check Inventory Levels

```
SELECT i.Name, inv.QuantityAvailable
FROM Item i
JOIN Inventory inv ON i.Item_ID = inv.Item_ID;
```

3. List Products by Category

```
SELECT i.Item_ID, i.Name, i.Description, i.Price
FROM Item i
JOIN Product p ON i.Item_ID = p.Item_ID
WHERE p.Category_ID = 1;
```

4. Find Products by Supplier

```
SELECT i.Item_ID, i.Name, i.Description, i.Price
FROM Item i
JOIN Product p ON i.Item_ID = p.Item_ID
JOIN StockRecord sr ON p.Item_ID = sr.Product_ID
WHERE sr.Supplier_ID = 1;
```

Review Queries

1. Retrieve All Reviews

```
SELECT *
FROM Review;
```

2. List Reviews for a Specific Product

```
SELECT r.Review_ID, r.Rating, r.Comment, c.Name AS CustomerName
FROM Review r
JOIN Customer c ON r.Customer_ID = c.Customer_ID
JOIN OrderItem oi ON r.OrderItem_ID = oi.OrderItem_ID
JOIN Item i ON oi.Item_ID = i.Item_ID
WHERE i.Item_ID = 1;
```

3. Average Rating for a Specific Product

```
SELECT AVG(r.Rating) AS AverageRating
FROM Review r
JOIN OrderItem oi ON r.OrderItem_ID = oi.OrderItem_ID
WHERE oi.Item_ID = 1;
```

Subscription and Training Program Queries

1. Retrieve All Subscriptions

```
SELECT i.Item_ID, i.Name, s.Duration, s.Benefit
FROM Item i
JOIN Subscription s ON i.Item_ID = s.Item_ID
```

2. Retrieve All Training Programs**

```
SELECT i.Item_ID, i.Name, tp.TP_StartingDate, tp.TP_EndingDate
FROM Item i
JOIN TrainingProgram tp ON i.Item_ID = tp.Item_ID;
```

Framework Contract Queries

1. List All Framework Contracts

```
SELECT *
FROM FrameworkContract;
```

2. Retrieve Contracts for a Specific Company

```
SELECT fc.ContractID, fc.Terms, fc.Starting_Date, fc.Ending_Date
FROM FrameworkContract fc
WHERE fc.Company_ID = 1;
```