

6360 project phase4

- a. Problem description (Copy it from Web site).
- b. Project questions (Answer 3 questions listed in the project, justify your solution).
- c. EER diagram with all assumptions (Solution for Phase II a).
- d. Relational Schema after normalization. All relations must be in 3NF. The relational schema should include Primary key as well as foreign keys (if any) for all relations. (Solution for Phase III a).
- e. All requested SQL statements (Solution for Phase III-c, e and f).
- f. Dependency diagram (Solution for Phase III-b).

a, Project Description

Food Festive Supermarket, a grocery store in Richardson, would like one relational database to store the information about their management system to be able to carry out their work in an organized way. They have some major modules such as Person, Products and Billing. A Person must be an Employee or a Silver Customer. Details of a person such as ID, Name (First, Middle, Last), Address, Gender, Date of Birth (Must be 16 years or older), and Phone number (one person can have more than one phone number) are recorded. The Person ID should have the format "PXXX" where X is a number from 0 to 9. A Silver Customer is classified as online or non-online customer. An online customer can also be a non-online customer. The email address of the online customer is stored. Only online customers can order products online. The order details such as order number, date, amount, product details and customer details are stored. One online customer can order multiple products and a product can be ordered by multiple online customers. Employee is classified as Cashier, Floor Staff or Managers. A floor staff can be promoted to cashier and later become a manager. The start date for each designation is recorded. Each floor staff is assigned the duty of arranging products into aisles. Aisle information such as section and aisle number is recorded. The date of assignment along with aisle number is stored for each employee. One employee maybe be assigned to arrange different aisles. The information about product-aisle arrangement and the date of arrangement is also stored. Products do not have a fixed aisle and can be arranged in different aisles throughout the year. Each employee works at a store. One employee can work in multiple stores but on a given day, can work only at one store. The date and working hours of the employee are stored. Store information such as name, address and contact are stored. Each store offers 'Sale' from time to time. Details such as sale ID, description and duration is recorded. The sale IDs are not unique and cannot be used to identify a sale in the system. A bill transaction is made by a cashier who records the list of products that are purchased by a person along with the date of purchase, bill amount, store ID and payment method. The cashier details, person details, store details and product details are stored together. A Gold Customer is someone who has some extra privileges than a Silver Customer. A Gold customer can be an Employee or a Non-online Customer or both. Different vouchers are issued by the store. A non-online customer needs to buy these vouchers but vouchers are given to a Gold Customers each month free of cost. Sometimes promotional discounts are offered on the vouchers and

details such promotion ID and promotion description are recorded. The Promotional IDs are not unique and cannot be used to identify a promotion in the system. Each Gold Customer is issued a membership card. A unique membership ID is generated for each Gold Customer. This number, date of issue and other information are stored. Product details such as product ID, Quantity (0,if out of stock), description and other information are stored. Products are further classified as either perishable or non-perishable items. Date of expiry is stored for the perishable items. Various suppliers, whose information are also stored in the system supply products. One Supplier may supply more than one product. But one product is supplied by only one supplier.

b. Project questions

1. Is the ability to model superclass/subclass relationships likely to be important in a grocery system environment such as Food Festive? Why or why not?
2. Can you think of 5 more business rules (other than the one explicitly described above) that are likely to be used in a supermarket environment? Add your rules to the above requirement to be implemented.
3. Justify using a Relational **DBMS** like Oracle for this project.

Solution:

1, Yes, I believe that the ability to model superclass subclass relationships is important in this grocery system environment such as Food Festive.

A subclass is a class that derives from another class. A subclass inherits state and behavior from all of its ancestors. The term superclass refers to a class's direct ancestor as well as all of its ascendant classes. There are likely to be different types of **PERSON**, **EMPLOYEE**, **CUSTOMERS** which are best represented in the form of a subclass of their respective superclass.

2,

(1)Each **Gold Customer** can only have one active **Membership** card, and the **Membership** card must be related with **Gold Customer**.

(2)Each **Bill Transaction** can only relate to one **Person** and one **Cashier**.

(3)**Purchase Detail** is the **Bill Transaction**'s detail item, so it can only relate to one **Bill Transaction** and one **Product**.

(4)Each **Shift Duty** is the same as above which can only relate to one **Store** and one **Employee**

(5)**Store** can offer 0~N **Sale**

3, see **grocery_store.sql**, **Views.sql**, **queries.sql** in the zip.

Grocery_store.sql create table to create database, tables and all other structures.

Views.sql to find Top Gold Customer, Popular Product, Top Store, Potential Gold Customers,Top Supplier

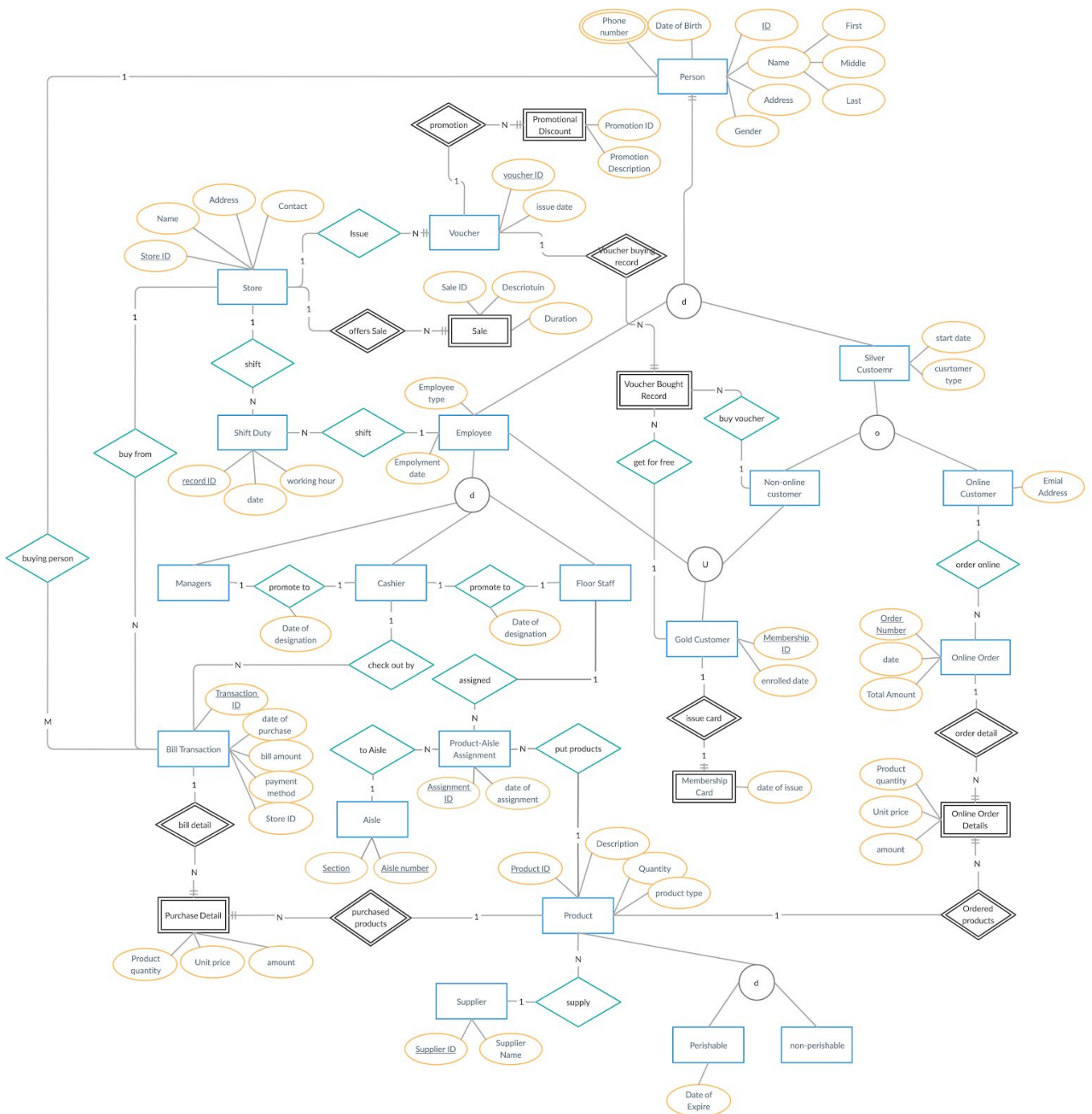
Queries.sql to find some special data what we want.

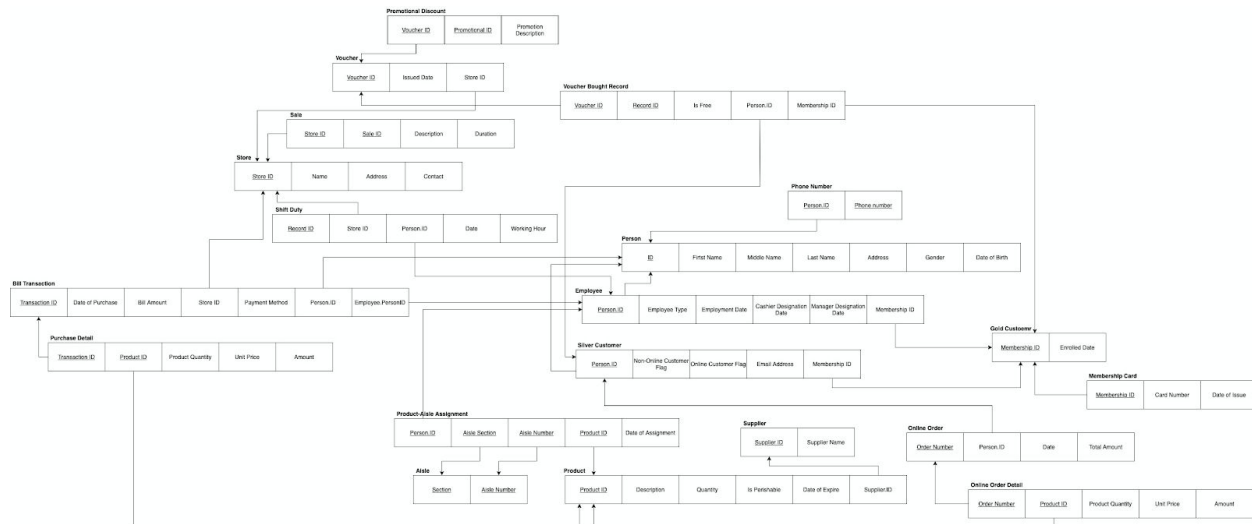
After using **MYSQL** i feel **MYSQL** suitable for many character:

it have High Performance and Scalability to meet the demands of exponentially growing data loads and users and self-healing Replication Clusters to improve scalability, performance and availability.

c. EER diagram with all assumptions

EER Diagram





Entity used in EER diagram:

Person, Employee, Managers, Cashier, Floor Staff, Silver Customer, Non-online Customer, Online Customer, Gold Customer, Membership Card, Store, Voucher, Voucher Bought Record, Promotional Discount, Sale, Shift Duty, Bill Transaction, Purchase Detail, Product, Supplier, Perishable, Non-Perishable, Product-Aisle Assignment, Aisle, Online Order, Online Order Detail.

Entity Detail Explanation:

1. **Person** can be specified into **Employee** and **Silver Customer**, and they are distinct.
2. **Employee** can be specified into **Managers**, **Cashier**, **Floor Staff**, and they are distinct.
3. **Silver Customer** can be specified into **Non-Online Customer** and **Online Customer**, and they are overlapped.
4. **Employee** and **Non-Online Customer** can be union to **Gold Customer**.
5. **Floor Staff** can be promoted to **Cashier**, **Cashier** can be promoted to **Managers**.
6. **Product-Aisle Assignment** is used to record the tasks assigned to **Floor Staff**, so it is related with **Floor Staff**, **Aisle** and **Product**.
7. **Bill Transaction** is made by **Cashier** to check the **Person** buying activity, so it is related with **Cashier**, **Person**.
8. **Purchase Detail** is used to record the item of each purchase (**Bill Transaction**), so it is a weak entity related with **Bill Transaction** and **Product**.
9. **Online Customer** can order online, so we use **Online Order** to record the order information, and also the **Online Order Details** to record the items of

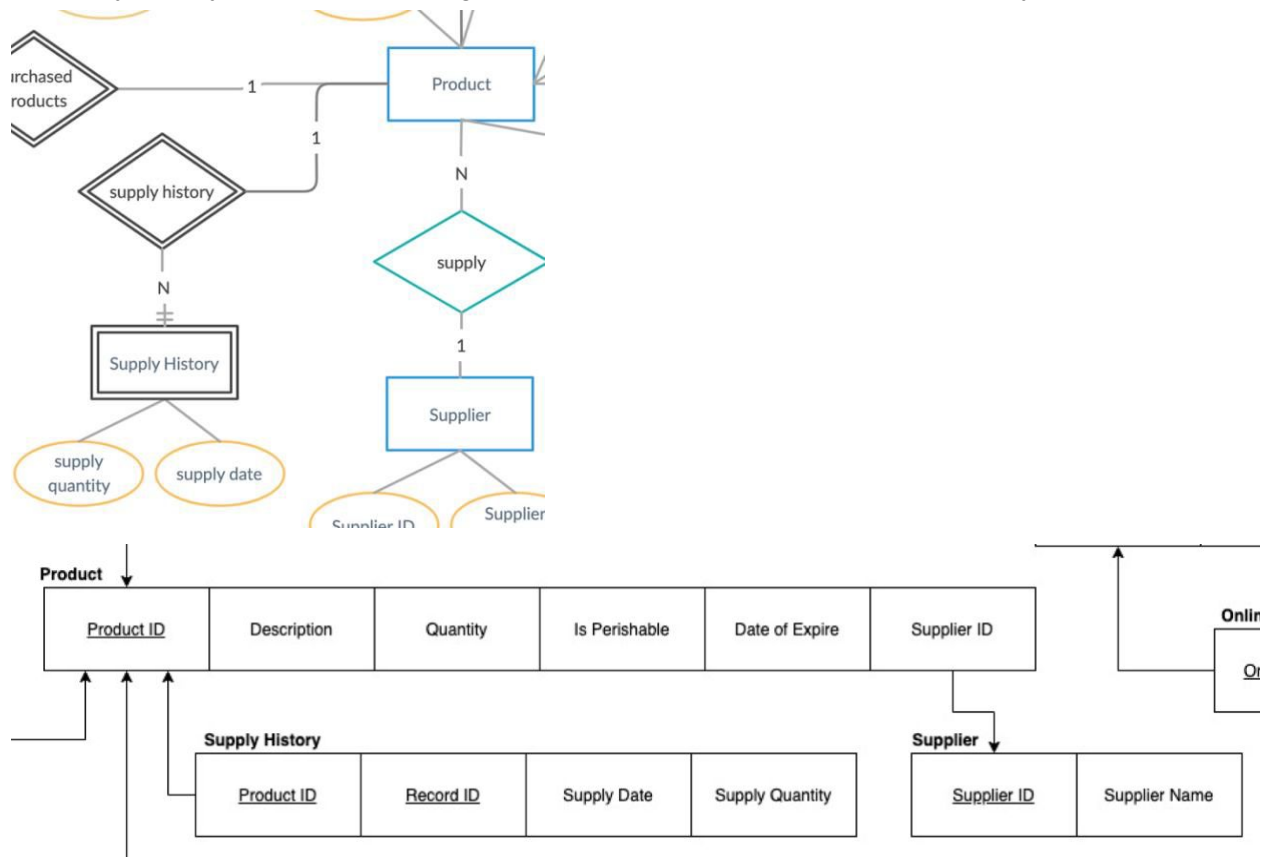
- each order. So the **Online Order** is related to **Online Customer** and **Online Order Detail**, **Online Order Detail** is also related with **Product**.
10. **Product** are supplied by **Supplier**.
 11. **Product** can be specified into **Perishable** and **Non-Perishable**.
 12. **Employee** can work for **Stores**, so we can use **Shift Duty** to record the working activities, i.e. record which employee can work for which Store at what date. With the same **Shift Duty** entity, as the same meaning, we can also know who works for the specific Store at that date.
 13. **Store** can issue **Voucher** which may contains **Promotional Discount**.
 14. **Non-Online Customer** can buy **Voucher**, and **Gold Customer** can get **Voucher** for free, so we can use **Voucher Bought Record** to record the information we have got the **Voucher**.
 15. **Store** can also offer **Sale**.

Relationship and Assumption:

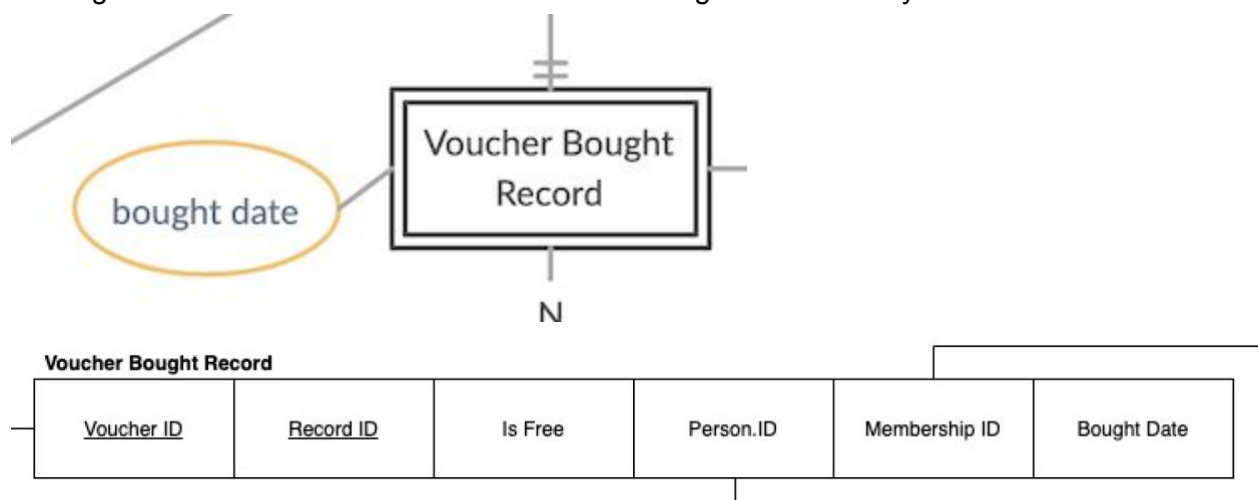
1. **Promotional Discount** is a weak entity related with **Voucher**, i.e. a **Voucher** may not have **Promotional Discount**, but a **Promotional Discount** cannot exist alone and must related with **Voucher**. (same assumption for all weak entities and will not specify below).
2. **Store** can issue 0~N **Vouchers** and **Vouchers** can contain 0~N **Promotional Discount**.
3. **Store** can offer 0~N **Sale**.
4. Each **Voucher Bought Record** is a record item which can only contain one **Voucher** and one **Gold Customer** or **Non-Online Customer**.
5. Each **Shift Duty** is the same as above which can only relate to one **Store** and one **Employee**.
6. Each **Bill Transaction** can only relate to one **Person** and one **Cashier**.
7. **Purchase Detail** is the **Bill Transaction**'s detail item, so it can only relate to one **Bill Transaction** and one **Product**.
8. Each **Product-Aisle Assignment** is the record item, which can only relate to one **Floor Staff**, one **Aisle** and one **Product**.
9. Each **Gold Customer** can only have one active **Membership card**, and the **Membership card** must be related with **Gold Customer**.
10. Each **Online Order Detail** is the record item, which can only relate with one **Online Order** and one **Product**. But generally, each **Online Order** can have many **Products**, and each **Product** can be ordered in many **Online Orders**. And the **Online Order**'s Total Amount is a derived attribute which is the sum of all order details' amount. **Purchase Detail** is quite similar.

d. Relational Schema after normalization. All relations must be in 3NF. The relational schema should include Primary key as well as foreign keys (if any) for all relations

1. Supply history and corresponding attributes are added to record more supply information.



2. Bought date is additional attribute for Voucher Bought Record entity



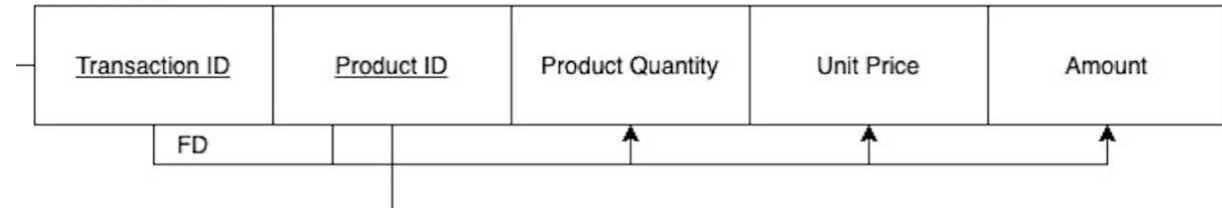
3. Normalization tables to 3NF:

All the tables in the relation diagram are already in 3NF, and the dependency diagram is attached in zip folder.

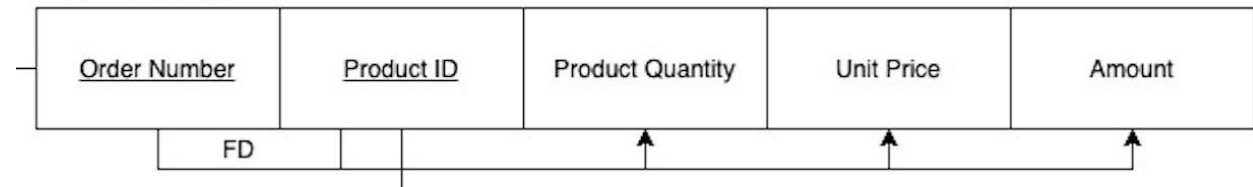
Special case:

In Online_order_detail and Purchase_detail tables, the “Amount” attribute is a derived attribute calculated from Product_quantity*Unit_price, so it will not be decomposed from the tables.

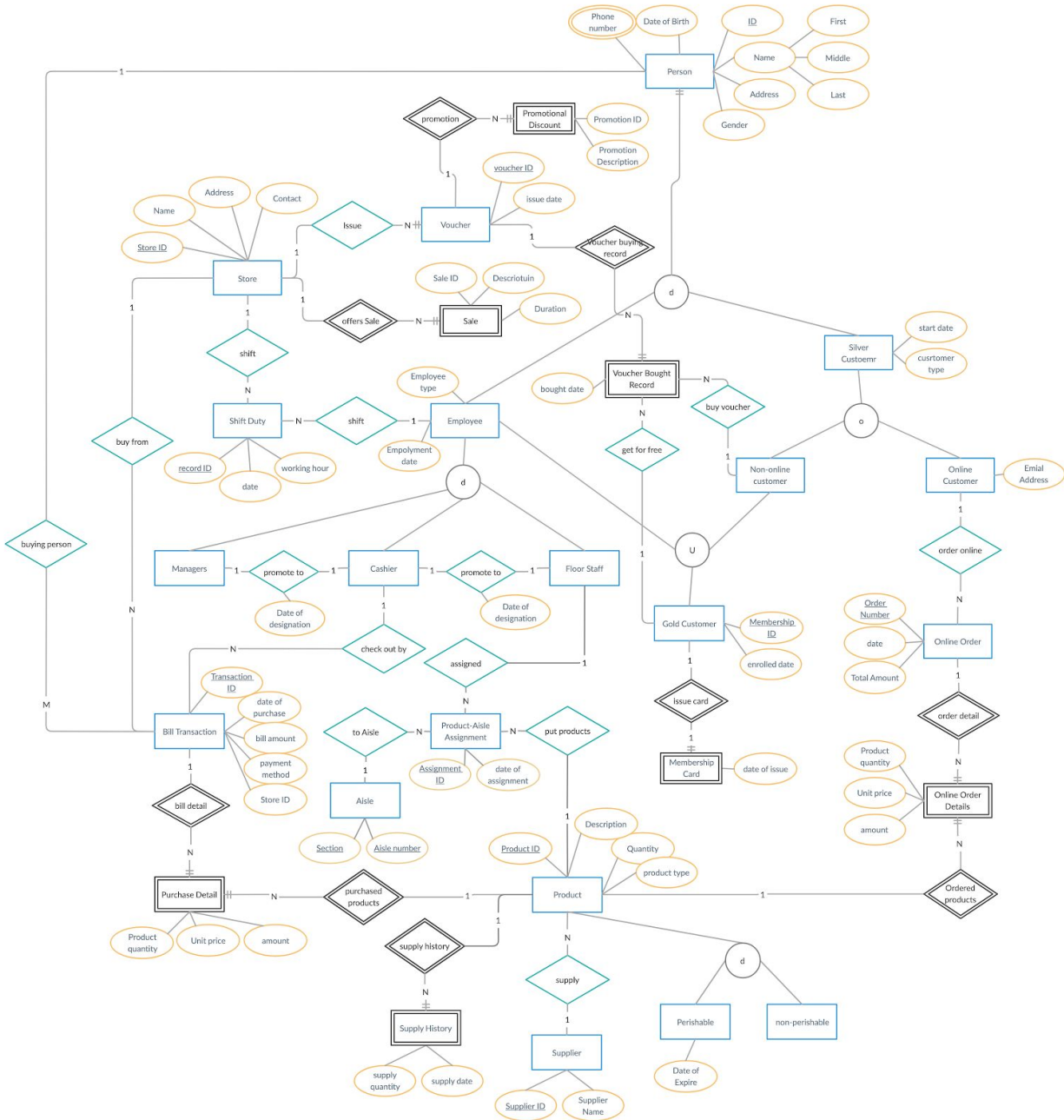
Purchase Detail



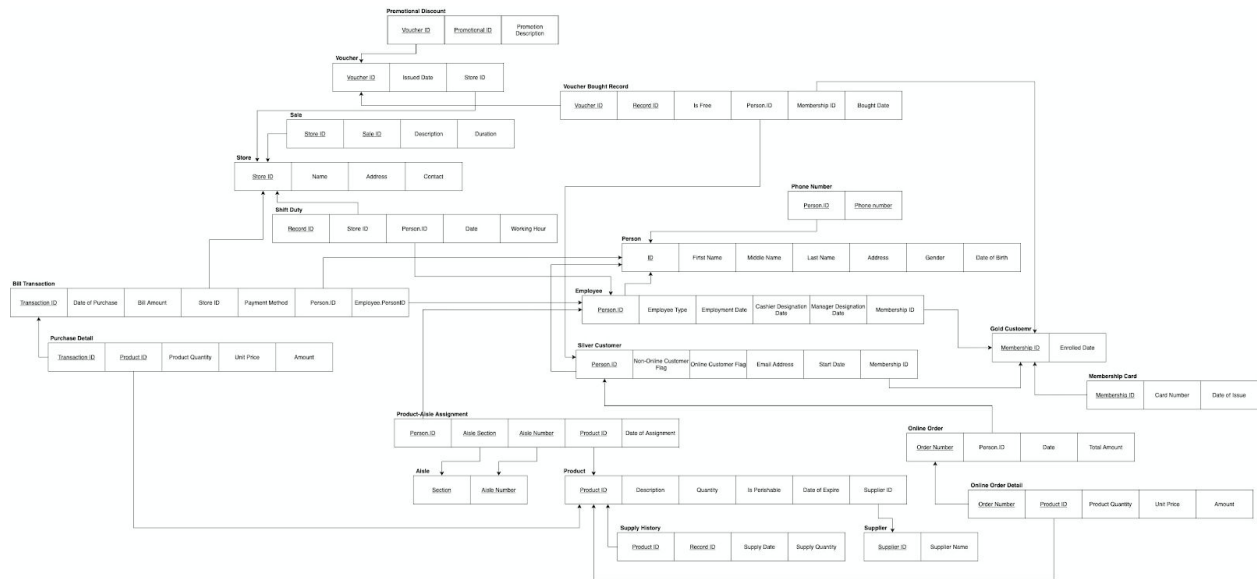
Online Order Detail



Modified EER Diagram



Modified Relation Diagram



e. All requested SQL statements

In the attachment **grocery_store.sql**, **Views.sql**, **queries.sql**

f. Dependency diagram

Dependency Diagram

