

Figure ER Diagram with appropriate relationship and cardinality for Use case provided

Using the ER diagram above, I was able to develop the SQL scripts to perform the creation of e-shop database, its tables and their relationships and later perform insertion of data of at least 5 rows in each table for answering the queries asked.

The following are the SQL statements for the following queries:

-- SQL Queries

-- 4. Provide SQL statements for the following queries:

-- a. Extract all the customers from a specific city.

SELECT \* from Customer as c

WHERE

c.city = "Lloyds";

-- b. Search for a product of a specific genre

SELECT \* from Product as p

WHERE

p.genre = 'genre-b';

-- c. Count how many customers are from a specific city.

SELECT COUNT(CustomerID)

FROM Customer

WHERE

city = "Lloyds";

-- d. Calculate the average of the unit price.

SELECT AVG(Unit\_Price)

FROM Product;

-- e. Extract all current orders

SELECT \*

FROM

Customer\_Order WHERE

Order\_Purchase\_Timestamp = "2022-12-11"

;

-- f. Extract all orders for books that has the keyword “the” in their description

SELECT \*

FROM Order\_Products as op

INNER JOIN Product ON op.ProductID=Product.ProductID

WHERE

Product.Product\_desc LIKE '%the%';

-- g. Extract all payments with credit cards for music records.

SELECT \*

FROM Order\_Products as op

INNER JOIN Product ON op.ProductID=Product.ProductID

INNER JOIN Product\_Type ON Product.Product\_TypeID=Product\_Type.Product\_TypeID

INNER JOIN Customer\_Order ON op.Customer\_OrderID=Customer\_Order.Customer\_OrderID

INNER JOIN Payment ON Customer\_Order.PaymentID=Payment.PaymentID

WHERE

Product\_Type.Product\_Type\_Name = "music records" AND

Payment.payment\_type = "Credit card"

;

-- h. Count how many employees handle music records.

SELECT COUNT(Employee.EmployeeID)

FROM Product

INNER JOIN Employee ON Product.EmployeeID=Employee.EmployeeID

INNER JOIN Product\_Type ON Product.Product\_TypeID=Product\_Type.Product\_TypeID

WHERE

Product\_Type.Product\_Type\_Name = "music records";

-- i. Count how many employees first name is John.

SELECT COUNT(EmployeeID)

FROM Employee

WHERE

Employee.First\_Name = "John";

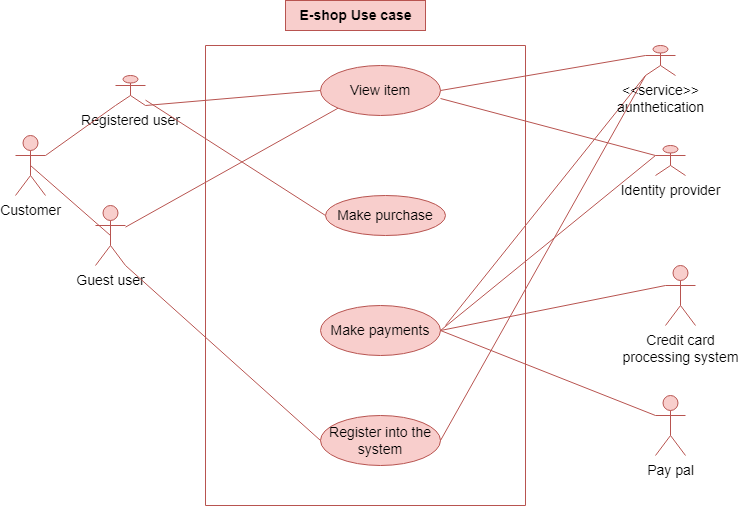
-- j. Count how many orders are in the system.

SELECT \*

FROM

Customer\_Order;

Below is the USE CASE diagram for our case:



Data governance plan from the above Use case

1. User/customer can only make payments in the e-shop if and only if is a registered user in the system otherwise, he/she will be allowed for browsing the products only.
2. Also, the payment is passed under strict authentication. The system only allows authenticated payments from registered users.