Antonio Rodriguez, Jacob Olmos (Team Appetite)

CS 3311-010

Dr. Roya Choupani
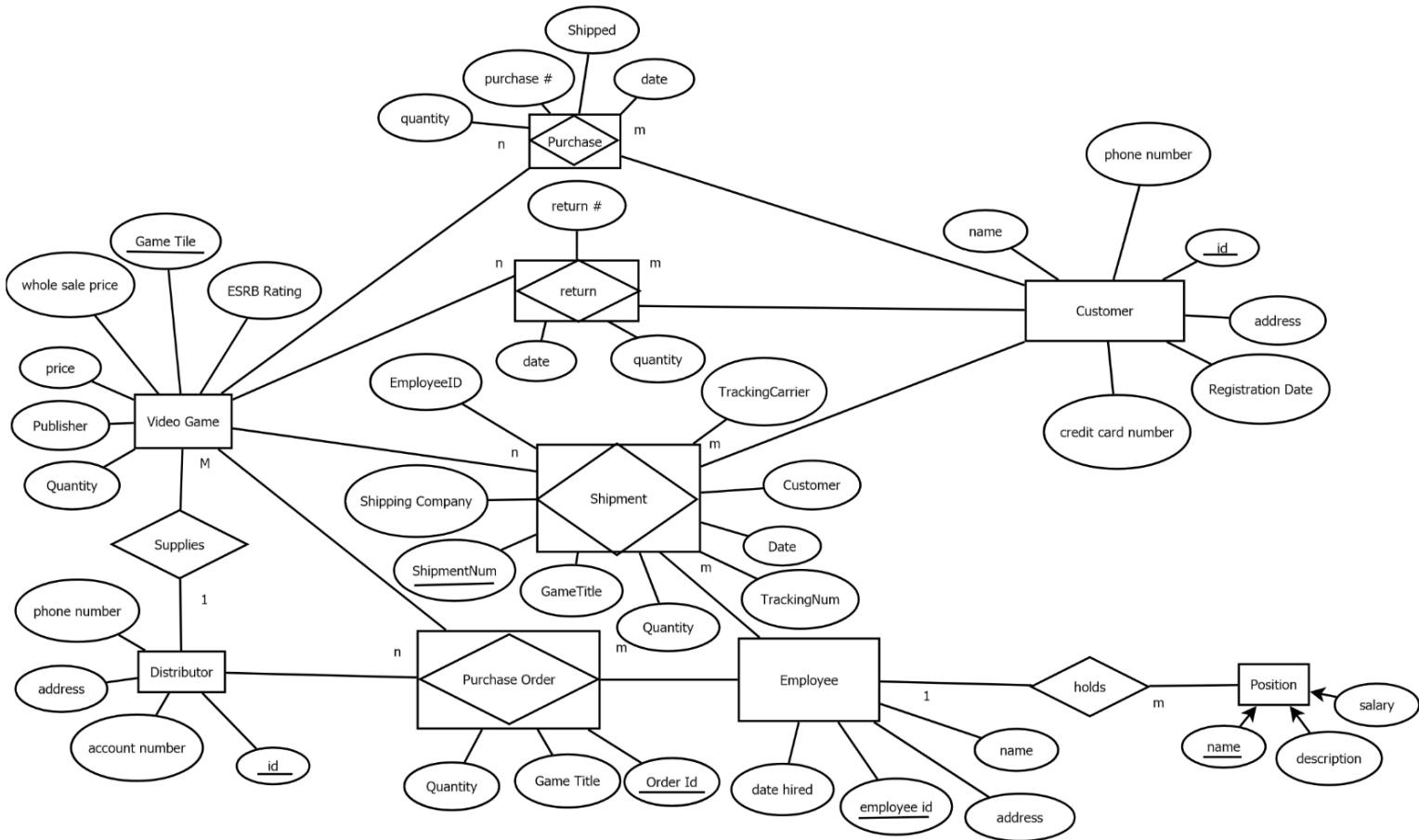
29 November 2022

Project: Game Store

Originally, our idea was to create a video game library that would assign what games people would own through a digital interface on Steam. We proceeded to expand the idea so much that it ended up becoming a digital game store focused on shipping physical games to customers.

We focused on aspects of what video games would be and how we'd acquire them through an online business focused on shipping.

We used the industry standard for video game costs, which is usually 60 dollars, using GameStop, Walmart, and Amazon as references. Most retailers buy games at wholesale for 50 dollars a piece, with 10 dollars for profit.

Shipped

purchase #

date

quantity

n Purchase m

return #

Game Tile

whole sale price

ESRB Rating

n return m

phone number

name

id

price

Publisher

Quantity

Video Game

EmployeeID

date quantity

TrackingCarrier

Customer

address

Registration Date

credit card number

M

Shipping Company

n Shipment m

Customer

Supplies

ShipmentNum

GameTitle

Date

TrackingNum

phone number

1

Quantity

m

address

Distributor

n Purchase Order m

Employee

1 holds m

Position

salary

account number

id

Quantity

Game Title

Order Id

date hired

employee id

name

name

description

name

address

Data Library

CustomerID
RegistrationDate
PhoneNum
CreditCardNum
Name
Address
phoneNumber
AccountNum
DistributorID
PositionName
Salary
Description
GameTitle
Platform
Price
Publisher
ESRBRating
Quantity
WholeSalePrice
Genre
DateHired
EmployeeID
CustomerOrderDate
PurchaseNum
Shipped
ShipmentNum
TrackingNum
TrackingCarrier
shipmentDate
ReturnNum
ReturnDate

Entity-Relationships

Entities:
Distributor
Video Game
Purchase
Customer
Shipment
Employee
Purchase Order
Position
Return

Relationships:
Distributor supplies **Video Game**.
Customer uses **Purchase** to buy a game from **Video Game.**
Video Game being bought become a **Purchase Order.**
Purchase Order gets processed by **Employee** and sends its to **Shipment**.
If **Customer** is unhappy with the game, he can send it to **Return** for a refund.

All Table Attributes/Created Tables

CREATE TABLE Distributor (
      address   VARCHAR(150)NOT NULL,
      phoneNumber DECIMAL(11),
      AccountNum   DECIMAL(9),
      DistributorID DECIMAL(9),
            Primary Key(DistributorID));

CREATE TABLE Customer (
      CustomerID   Decimal(5),
      RegistrationDate   date,
      PhoneNum   CHAR(10),
      CreditCardNum   Decimal(16),
      Name   VARCHAR(150)NOT NULL,
      Address   VARCHAR(150)NOT NULL,
            PRIMARY KEY(CustomerID));

CREATE TABLE EmployeePosition (
      PositionName Char(50),
      Salary   Decimal(5),
      Description   Char(30),
            Primary Key(PositionName));

CREATE TABLE VideoGame (
      GameTitle   VARCHAR(50)NOT NULL,
      Platform   Char(50),
      Price   Decimal(5,2),
      Publisher   VARCHAR(100)NOT NULL,
      DistributorID Decimal(9),
      ESRBRating   Char(5),
      Quantity   DECIMAL(5),
      WholeSalePrice   DECIMAL(9),
      Genre   CHAR(50)
            Primary Key(GameTitle),
            FOREIGN KEY(DistributorID) REFERENCES Distributor(DistributorID));

CREATE TABLE Employee (
      DateHired   date,

```
        EmployeeID    Decimal(9),
        Name              VARCHAR(100)NOT NULL,
        Address                VARCHAR(100)NOT NULL,
        PositionName  Char(50),
              Primary KEY(EmployeeID),
              FOREIGN KEY(PositionName) REFERENCES
EmployeePosition(PositionName));

CREATE TABLE PurchaseOrderCustomer (
        CustomerOrderDate              Date,
        PurchaseNum                    DECIMAL(9),
        Quantity                            DECIMAL(9),
        GameTitle                      VARCHAR(50)NOT NULL,
        Platform                       Char(50),
        Shipped                             char(20),
        CustomerID                     Decimal(5),
              Primary KEY(PurchaseNum),
              FOREIGN KEY(GameTitle) REFERENCES VideoGame(GameTitle),
              FOREIGN KEY(CustomerID) REFERENCES Customer(CustomerID));

Create Table Shipment (
        GameTitle     VARCHAR(50)NOT NULL,
        Platform      Char(50),
        EmployeeID    Decimal(9),
        ShipmentNum Decimal(20),
        TrackingNum  Decimal(20),
        TrackingCarrier       char(50),
        CustomerID    Decimal(5),
        Quantity              Decimal(9),
        shipmentDate  date,
              PRIMARY KEY(ShipmentNum),
              Foreign Key(EmployeeID) REFERENCES Employee(EmployeeID),
              Foreign KEY(CustomerID) REFERENCES Customer(CustomerID));

Create Table ReturnItem (
        ReturnNum     Decimal(5),
        Quantity      Decimal(5),
        GameTitle     VARCHAR(50)NOT NULL,
        Platform      Char(50),
        CustomerID    Decimal(5),
```

ReturnDate   Date,
          Primary Key(ReturnNum),
          FOREIGN KEY(CustomerID) REFERENCES Customer(CustomerID),
FOREIGN KEY(GameTitle) REFERENCES VideoGame(GameTitle));

Queries Used

Total Income
Select Sum(VideoGame.Price*PurchaseOrder.Quantity) AS TotalIncome From VideoGame, PurchaseOrder
   where VideoGame.GameTitle = PurchaseOrder.GameTitle;

Game Expenses
Select Sum(VideoGame.WholeSalePrice) AS GameExpenses From Videogame;

Total Employee Salary
Select SUM(EmployeePosition.Salary) AS TotalEmpSalary FROM Employee, EmployeePosition
   where Employee.PositionName = EmployeePosition.PositionName;

Inventory Report
Select VideoGame.GameTitle, VideoGame.Platform, VideoGame.Publisher, VideoGame.Quantity,
      VideoGame.Price, VideoGame.WholeSalePrice FROM VideoGame;

Shipment Report
Select PurchaseOrderCustomer.GameTitle, PurchaseOrderCustomer.Platform, PurchaseOrderCustomer.Shipped
   FROM PurchaseOrderCustomer, VideoGame
   where VideoGame.GameTitle = PurchaseOrderCustomer.GameTitle;

Shipment History
Select Customer.Name, Shipment.GameTitle, Shipment.Platform, PurchaseOrderCustomer.CustomerOrderDate, Shipment.shipmentDate,
   Shipment.ShipmentNum, Shipment.TrackingNum, Shipment.TrackingCarrier From Customer, Shipment, PurchaseOrderCustomer
   where Customer.CustomerID = Shipment.CustomerID AND PurchaseOrderCustomer.Quantity = Shipment.Quantity;

Purchase Order Report
Select PurchaseOrderCustomer.GameTitle, PurchaseOrderCustomer.Platform,
   PurchaseOrderCustomer.Quantity, PurchaseOrderCustomer.Shipped,
   PurchaseOrderCustomer.CustomerID, PurchaseOrderCustomer.CustomerOrderDate FROM
   PurchaseOrderCustomer where PurchaseOrderCustomer.CustomerOrderDate = '2022-11-26';

Genre Selection
SELECT * FROM videogame
WHERE LOCATE("Action", Genre);

Publisher Selection
SELECT * FROM videogame WHERE Publisher = "Nintendo";

Discounting Price
Update videogame
Set
Price =
Case
WHEN publisher = "Nintendo" then Price-((Price*10) /100)
end;

Alphabetical Order
Select GameTitle, Platform, Price, Publisher, DistributorID, ESRBRating, Quantity,
WholeSalePrice, Genre
FROM videogame
ORDER BY GameTitle ASC;

Lowest to Highest Price Order
Select GameTitle, Platform, Price, Publisher, DistributorID, ESRBRating, Quantity,
WholeSalePrice, Genre
FROM videogame
ORDER BY Price ASC;